Semantic-based Service Analysis and Optimization

A thesis submitted to the

University of Dublin, Trinity College,

for the degree of

Doctor of Philosophy

Liam Fallon

Knowledge and Data Engineering Group,

Department of Computer Science,

Trinity College, University of Dublin,

Ireland

2013

Volume I

Thesis Proper

Declaration

I, the undersigned, declare that this work has not previously been submitted as an exercise for a degree at this or any other University, and that, unless otherwise stated, it is entirely my own work.

_____

Liam Fallon

March 2013

Permission to Lend or Copy

I, the undersigned, agree that the Trinity College Library may lend or copy this thesis upon request.

_____

Liam Fallon

March 2013

Acknowledgements

Firstly, I would like to thank my supervisor, Prof. Declan O'Sullivan, for his guidance, encouragement, and support over the course of the last number of years. His advice has always been insightful and has been invaluable for keeping my focus on the core of the research. I would also like to thank Dr. John Kenney, Dr. Sidath Handurukande, and Dr. Sven van der Meer at the Network Management Lab in Ericsson Ireland for their advice and suggestions during the course of this research, and also David Cleary for encouraging me to undertake it. I would also like to thank all the members of the KDEG group at TCD for their advice and help, and for always making me feel welcome.

I would like to thank my colleagues in Ericsson Ireland for their support and understanding during the course of this research. I would particularly like to thank my managers Jimmy O'Meara and Gabriel Hogan, who always supported and encouraged me, and aided me in balancing the work for this this research with my other activities. I would also like to thank the staff of the Prototyping and Customer Trials group at Ericsson Ireland for their support and understanding, particularly Sajeevan Achuthan, Damien Brennan and Mark McFadden.

I would like to thank Ericsson Ireland for funding this research through its excellent Scholar programme, and Michael Gallagher, Tony Devlin, and Matt Hamilton for their continued support for the duration of this research.

I would like to thank my family, particularly Aoife, Eoghan, Kenae, and Elias for their love and understanding, and for putting up with my many physical and virtual absences over this period. I would also like to thank my parents for instilling the value of learning in me and for their continuing encouragement.

Finally, I would like to dedicate this thesis to Anne, without whose love, support, advice, patience, and serenity I could never have undertaken this research.

Abstract

The need to autonomically optimize end user service experience in near real time has been identified in the literature in recent years. Service management systems that monitor end user service session context are deployed but approaches that estimate end user service experience from session context do not analyse the compliance of that experience with user expectations. Approaches that plan and execute actions to optimize end user service session delivery are not applicable to arbitrary service sessions; they work with specific service types and delivery mechanisms or do not consider end user service experience when making optimization decisions. Another barrier to autonomic end user service management optimization is the lack of a holistic model for the domain.

This thesis proposes **the Aesop approach**, an approach that addresses **semantic-based autonomic optimization of end user service delivery**. This approach has a knowledge base at its core and proposes the **EUSAO** ontology. This ontology, designed to semantically model the end user service management domain, enables partitioning of knowledge that varies over time for efficient access. The **Aesop Engine** is designed to execute an iteration of an autonomic loop (Monitor, Analyse, Plan, Execute) in near real time. It runs semantic algorithms designed to use queries and rules on subsets of the partitioned EUSAO-based knowledge in order to monitor end user sessions, to analyse their compliance with expectations, to plan optimizations, and to execute those optimizations as throttling actions on the service delivery network. The semantic-based algorithms that are proposed are efficient because they operate on small partitioned subsets of the knowledge base. The Aesop approach allows arbitrary end user service types and network domains to be added by specializing the EUSAO ontology for that domain and adding domain specific semantic mappings, queries and rules. A case study has demonstrated that the approach is applicable in the Mobile Broadband access domain.

A prototype implementation of the complete Aesop approach was evaluated on a full purpose built Home Area Network test bed, on which execution of end user service sessions was automated and controlled. In the evaluation, when the measured compliance of a set of end user service sessions with expectations when Aesop optimization was active was compared with the compliance of an identical set of sessions when Aesop optimization was inactive, a significant improvement was observed on the compliance levels of high priority sessions in all experimental scenarios, with compliance levels more than doubled in some cases.

The key contribution of the Aesop approach is that it enables user centric service delivery management. When used with other service specific and network centric approaches, it shows promise as a method of optimizing an arbitrary running set of end user service sessions so that they best meet the expectations of the individual users running those service sessions. In addition, this research has addressed key concerns that have inhibited the widespread adoption of semantic techniques by showing that, with appropriate design and implementation, semantic techniques can be efficient and can perform well, and are deployable in full autonomic systems that execute in near real time.

Table of Contents

Declaration i
Permission to Lend or Copy ii
Acknowledgements iii
Abstract iv
Table of Contents v
Table of Figures viii
Table of Tables x
Abbreviations xii
Chapter 1 Introduction 1
1.1 Motivation 1
1.2 Research Question and Objectives 2
1.3 Research Process and Approach 4
1.3.1 Research Artefacts 6
1.3.2 The Research Phases 6
1.3.3 Evaluation 8
1.4 Contributions 10
1.5 Thesis Overview 13
Chapter 2 Background and Related Work 15
2.1 Introduction 15
2.2 Background 15
2.3 Related Work 19
2.3.1 End User Service Delivery Monitoring, Analysis, and Optimization 19
2.3.2 Related Models, Vocabularies, and Ontologies 29
2.3.3 Semantic Techniques Applicable to End User Service Management 34
2.3.4 Semantics in Autonomic Network Management 37
2.4 Summary 39
Chapter 3 The EUSAO Ontology 42
3.1 Introduction 42
3.2 Requirements 42
3.3 Design 43
3.4 Implementation 52
3.5 Using the Ontology 54
3.6 Summary 56
Chapter 4 Aesop 57
4.1 Introduction 57
4.2 Requirements 58
4.3 Design 59
4.3.1 Semantic Components 61
4.3.2 The Knowledge Base 62
4.3.3 The Knowledge Bus 64
4.3.4 The Aesop Engine 66
4.4 Implementation 76
4.4.1 Semantic Components 77
4.4.2 The Knowledge Base 78
4.4.3 The Knowledge Bus 79
4.4.4 The Aesop Engine 79
4.5 Summary 86
Chapter 5 The SECCO Framework 88
5.1 Introduction 88
5.2 Requirements 89
5.3 Design 90
5.3.1 The *SECCOReporter* 91
5.3.2 The *SECCOClient* 93
5.4 Implementation 95
5.4.1 The *SECCOReporter* 96

5.4.2 The *SECCOClient* 98
5.5 Summary 101
Chapter 6 Experimentation 102
6.1 Introduction 102
6.2 The Experimental Framework 104
6.2.1 Domain 105
6.2.2 End User Services 106
6.2.3 Scenarios 107
6.2.4 Simulation Environment 111
6.2.5 Test Bed Environment 113
6.2.6 Experimental Timings and Runs 115
6.2.7 Experimental Methodology 116
6.3 Experiment 1: Semantic Monitoring of Terminal Reports 119
6.3.1 Hypothesis 119
6.3.2 Experimental Setup and Execution 119
6.3.3 Experimental Results 120
6.3.4 Analysis of Results 121
6.3.5 Summary 126
6.4 Experiment 2: Semantic Analysis of Service Experience 128
6.4.1 Hypothesis 128
6.4.2 Experimental Setup and Execution 129
6.4.3 Experimental Results 129
6.4.4 Analysis of Results 131
6.4.5 Summary 140
6.5 Experiment 3: Semantic Optimization of Services towards Expectations 141
6.5.1 Hypothesis 142
6.5.2 Experimental Setup and Execution 143
6.5.3 Experimental Results 143
6.5.4 Analysis of Results 145
6.5.5 Summary 163
6.6 Experience with Semantic Technologies Used 166
6.6.1 Protégé 166
6.6.2 Jena 167
6.6.3 The Pellet Reasoner 167
6.6.4 SPARQL 168
6.6.5 SWRL Rules 168
6.7 Conclusions from Experimentation 168
Chapter 7 Mobile Broadband Case Study 170
7.1 Introduction 170
7.2 Background 171
7.3 Scenario 173
7.4 Suggested Placement of Aesop in the 3GPP Network Architecture 173
7.5 Assessment of Applicability of Aesop Architecture and Structure 175
7.5.1 Applying the EUSAO Ontology and the Knowledge Base 176
7.5.2 Applying Semantic Knowledge Monitoring 177
7.5.3 Applying Semantic Analysis 178
7.5.4 Applying Semantic Optimization Planning and Execution 178
7.5.5 Deploying Aesop in a Full Network 179
7.5.6 Optimizing End User Service Delivery in a Full Network 180
7.5.7 Synopsis of Enhancements and Extensions to Aesop 182
7.6 Summary 183
Chapter 8 Conclusions and Future Work 185
8.1 Overview 185
8.2 Research Question and Objectives 185
8.3 Contributions 192
8.4 Future Work 194
8.5 Final Remarks 195
References 196
Appendix A End User Service Delivery on DSL and Mobile Broadband Access Networks A-1
A.1 Introduction A-1
A.2 End User Services on the Internet A-1

A.3 Impact of Network Context on Services A-2
A.3.1 Video on Demand A-2
A.3.2 Web Browsing A-4
A.3.3 File Transfer A-5
A.4 Service Delivery Context in DSL-Connected HANs A-5
A.5 Service Delivery Context in Mobile Broadband Access Networks A-8
Appendix B The EUSAO Ontology B-1
Appendix C SPARQL Queries and SWRL Rules C-1
C.1 SWRL Rules for Knowledge Loading C-1
C.2 SPARQL Queries for Semantic Analysis C-1
C.3 SPARQL Queries for Optimization Planning and Execution C-5
Appendix D Aesop Implementation Details D-1
D.1 The Knowledge Bus Interface D-1
D.2 Static Global Knowledge Configuration in Protégé D-2
D.3 SAWSDL Annotations D-3
D.4 XSL Transformations for Lifting Terminal Report Elements D-4
Appendix E Experimental Equipment E-1
Appendix F Test Bed Photographs F-1
Appendix G Third Party Software G-1
Appendix H Normal Quantile Plots for Experimental Metrics H-1
Appendix I Experimental Scenario Specifications I-1
I.1 Scenario 1: Degradation in Video on Demand Session I-1
I.2 Scenario 2: Congestion on Broadband Access Network Connection I-1
I.3 Scenario 3: Transient Impairment on Broadband Access Network Connection I-3
Appendix J Experimental Run Logs J-1
J.1 Experiment 1 ..... J-1
J.2 Experiment 2 ..... J-2
J.3 Experiment 3 ..... J-4
Appendix K Samples of Collected Results K-1
K.1 Samples of Simulation Results K-1
K.2 Samples of Test Bed Results K-2
Appendix L Examples of Analysis Scripts and Queries L-1
Appendix M Parameters Collected in Experiments M-1
M.1 Experiments in Simulated Environment M-1
M.2 Experiments in Test Bed Environment M-2
Appendix N Experimental Results N-1
N.1 Results for Experiment 1 ..... N-1
N.2 Results for Experiment 2 ..... N-3
N.2.1 Results on Session Users and Priorities N-3
N.2.2 Results on Accuracy of Aesop N-4
N.2.3 Statistical Analysis of Aesop Performance Parameters N-5
N.2.4 Statistical Analysis of Session Compliance by Scenario and Type N-13
N.3 Results for Experiment 3 ..... N-21
N.3.1 Optimization Effectiveness and Statistical Analysis of Session Compliance N-21
N.3.2 Throttling Actions and Consistency of Aesop Optimization N-31
N.3.3 Results on Session Users and Priorities N-36
N.3.4 Results on Accuracy of Aesop N-37
N.3.5 Statistical Analysis of Aesop Performance Parameters N-39

Table of Figures

Figure 1-1: Research Phases 5
Figure 1-2: Organization of Research 6
Figure 3-1: Core Concepts in the EUSAO Ontology 46
Figure 3-2: Core Global Concepts in the EUSAO Ontology 48
Figure 3-3: Snapshot Concepts and Referenced Core Global Concepts in the EUSAO Ontology 50
Figure 3-4: Sub-Concepts Representing Service Type Concepts 51
Figure 3-5: The *SSM_Terminal* Concept Specification in Protégé 53
Figure 3-6: The *ssm_communicatesUsing* Reference in Protégé 54
Figure 3-7: The *ssm_terminalModel* Attribute in Protégé 54
Figure 4-1: The Aesop System 60
Figure 4-2: UML Class Diagram of Aesop Software Components 61
Figure 4-3: Knowledge Monitoring 67
Figure 4-4: The Aesop Knowledge Loading Algorithm UML Activity Diagram 69
Figure 4-5: The Aesop Semantic Analysis Algorithm UML Activity Diagram 71
Figure 4-6: The Aesop Optimization Planning and Execution Algorithm UML Activity Diagram 73
Figure 4-7: UML Class Diagram of a Prototype Implementation of Aesop 76
Figure 4-8: Semantic Analysis Implementation 82
Figure 5-1: Conceptual Overview of SECCO 91
Figure 5-2: SECCO UML Deployment Diagram 91
Figure 5-3: *SECCOReporter* UML Class Diagram 92
Figure 5-4: *connection* Package UML Class Diagram 92
Figure 5-5: *service* Package UML Class Diagram 93
Figure 5-6: *SECCOClient* Class Diagram 94
Figure 5-7: SECCO Instance UML Deployment Diagram 96
Figure 5-8: Firefox Deployment for Web Browsing end User Session Recording 101
Figure 6-1: Facets of the Experimental Framework 104
Figure 6-2: Timeline of Scenario 1 109
Figure 6-3: Timeline of Scenario 2 110
Figure 6-4: Timeline of Scenario 3 111
Figure 6-5: Approximated Relationship between Packet Loss and MOS 112
Figure 6-6: Simulation Environment 113
Figure 6-7: Test Bed for Experimentation 113
Figure 6-8: Logical View of Test Bed with Running Sessions and Terminal Reporting 114
Figure 6-9: Relationship between Time Parameters in Experiment 1 120
Figure 6-10: Statistical Analysis of Results, 6 Terminals Reporting 121
Figure 6-11: Elapsed time per Run, 10 Runs 122
Figure 6-12: Average Generate to Store Time: 10 Runs 123
Figure 6-13: Average Processing Time, 10 Runs 123
Figure 6-14: Average Intercept Time, 10 Runs 124
Figure 6-15: Spread of Processing Time Data, All Runs 125
Figure 6-16: Spread of Intercept Time Data, All Runs 126
Figure 6-17: Processing Time Data Points (300 Terminals Reporting), All Runs 127
Figure 6-18: Intercept Time Data Points (300 Terminals Reporting), All Runs 127
Figure 6-19: Monitoring Times Average and Standard Deviation 133
Figure 6-20: Analysis Times Average and Standard Deviation 134
Figure 6-21: CPU Usage Average and Standard Deviation 135
Figure 6-22: Memory Usage Average and Standard Deviation 136
Figure 6-23 Average Session Compliance, Scenario 2 137
Figure 6-24: Average Session Compliance, Scenario 3 139
Figure 6-25: Effectiveness of Aesop Optimization, Scenarios 2 and 3 147
Figure 6-26: Average Session Compliance before and after Optimization, Scenario 2 148
Figure 6-27: Average Session Compliance before and after Optimization, Scenario 3 150
Figure 6-28: Throttling on Scenario 2 Sessions 152
Figure 6-29: Throttling on Scenario 3 Sessions 153

Figure 6-30: Optimization Times Average and Standard Deviation 157
Figure 6-31: Monitoring Times Average and Standard Deviation 159
Figure 6-32: Analysis Times Average and Standard Deviation 160
Figure 6-33: Average Times for Experiment 3 Runs 161
Figure 6-34: CPU Usage Average and Standard Deviation 162
Figure 6-35: Memory Usage Average and Standard Deviation 163
Figure 7-1: Base Stations and Possible Cells in Athlone, Ireland 172
Figure 7-2: Aesop End User Service Delivery Optimization for Mobile Broadband 174
Figure 7-3: An Aesop Instance Managing Multiple Cell Wireless Networks 176
Figure 7-4: TM Forum SID *NetworkComposite* and *SubNetwork* Concepts in UML 177
Figure 7-5: Aesop Instance Managing Regions of a Mobile Broadband Access Network 179

Figure A-1: Comparison of Fixed and Mobile Broadband Performance, from [OFCOM, 2011] A-9
Figure A-2: Comparison of Download Speeds as Signal Strength Varies, from [OFCOM, 2011] A-10
Figure A-3: Comparison of Download Speeds for Case Studies, from [OFCOM, 2011] A-11
Figure B-1: The EUSAO Ontology B-1
Figure B-2: The EUSAO Ontology in Protégé showing Concepts as Classes B-2
Figure B-3: The EUSAO Ontology in Protégé showing References as Object Properties B-2
Figure B-4: The EUSAO Ontology in Protégé showing Attributes as Data Properties B-3
Figure D-1: Service Expectation and Service Priority Configuration D-2
Figure F-1: The Experimental Test Bed F-1
Figure F-2: The Server, Traffic Shaper, and Home Gateway F-1
Figure F-3: Home Gateway Monitor F-2
Figure F-4: Close up of Home Gateway Monitor F-2
Figure F-5: Laptops and HAN Ethernet Switch F-3
Figure F-6: Laptop 1 running a Workplace Web Session F-3
Figure F-7: Laptop 2 running a File Transfer Session F-4
Figure F-8: Laptop 3 running a Social Networking Web Session F-4
Figure F-9: Laptop 4 running a Video on Demand Session F-5
Figure F-10: Close Up of Degraded Video Session running on Laptop 4 ..... F-5
Figure H-1: Normal Quantile Plots for Experimental Metrics H-1
Figure K-1: Screenshot of the Aesop client at the end of Experiment 3 Scenario 3 Run 2 ..... K-6

Table of Tables

Table 1-1: Aesop Evaluation Properties Assessed during Experimentation 9
Table 2-1: Summary of Service Expectation Metrics 23
Table 3-1: The Dynamicity and Temporality Properties of Concepts in the EUSAO ontology 44
Table 3-2: Concept Categories as used by Concepts in the EUSAO Ontology 45
Table 3-3: Ontologies, Vocabularies, and Models Used and Referenced 47
Table 3-4: Snapshots Recording Knowledge with a Time Dimension 49
Table 4-1: Static Global Knowledge Configured in Aesop 65
Table 4-2: An Annotated Element Lookup Reference (AELR) entry 81
Table 6-1: Synopsis of Experimentation 103
Table 6-2: Emulated Connection Characteristics, Congestion Experiments 105
Table 6-3: Emulated Connection Characteristics, Network Impairment Experiments 106
Table 6-4: Service Expectation Parameters per Service 107
Table 6-5: Service Priorities and Throttling for User Service Sessions 108
Table 6-6: Metrics Examined using Statistical Analysis 116
Table 6-7: Lowest and Highest Processing Times 125
Table 6-8: Aesop Monitoring Accuracy 131
Table 6-9: Aesop Analysis Accuracy 132
Table 6-10: Aesop Optimization Average Effectiveness 146
Table 6-11: Throttling Spreads for Scenario 2 153
Table 6-12: Throttling Spreads for Scenario 3 153
Table 6-13: Aesop Monitoring Accuracy 154
Table 6-14: Aesop Analysis Accuracy 155
Table 6-15: Aesop Optimization Accuracy 156
Table 7-1: Aesop Deployment in a Mobile Broadband Access Network with 50,000 Cells 180

Table A-1: UK DSL Advertised 20Mbit/s DSL Residential Broadband by ISP, November 2011 ..... A-6
Table A-2: UK Ad. 20Mbit/s DSL Residential Broadband Download Spread, November 2011 ..... A-6
Table A-3: Mobile Broadband Characteristics Measured by OFCOM A-10
Table D-1: Methods on the Knowledge Bus Interface D-1
Table D-2: Transformations Used to Lift Terminal Report Elements D-4
Table E-1: Experimental Equipment E-1
Table G-1: Third Party Software Used G-1
Table I-1: Scenario 1 Experimental Specification I-1
Table I-2: Scenario 2 Experimental Specification I-2
Table I-3: Scenario 3 Experimental Specification I-3
Table J-1: Experiment 1 Scenario 1 Run Log J-1
Table J-2: Experiment 2 Scenario 2 Run Log J-2
Table J-3: Experiment 2 Scenario 3 Run Log J-3
Table J-4: Experiment 3 Scenario 2 Run Log J-4
Table J-5: Experiment 3 Scenario 3 Run Log J-5
Table M-1: Parameters Collected in Experiments in Simulated Environment M-1
Table M-2: Parameters Collected on Sessions M-2
Table M-3: Parameters Collected on Session Metrics M-3
Table M-4: Parameters Collected on Throttling Actions M-4
Table M-5: Parameters Collected on Connections M-5
Table M-6: Other Parameters Collected M-5
Table N-1: Terminal and Terminal Report Counts for tests of Experiment 1 ..... N-1
Table N-2: Statistical Analysis of Results for Experiment 1 ..... N-2
Table N-3: Scenario 2 Sessions N-3
Table N-4: Scenario 3 Sessions N-3
Table N-5: Aesop Monitoring Accuracy N-4
Table N-6: Aesop Analysis Accuracy N-4
Table N-7: Statistical Analysis of Scenario 2 Monitoring Times N-5
Table N-8: Statistical Analysis of Scenario 2 Analysis Times N-6
Table N-9: Statistical Analysis of Scenario 2 CPU Usage N-7

Table N-10: Statistical Analysis of Scenario 2 Memory Usage N-8
Table N-11: Statistical Analysis of Scenario 3 Monitoring Times N-9
Table N-12: Statistical Analysis of Scenario 3 Analysis Times N-10
Table N-13: Statistical Analysis of Scenario 3 CPU Usage N-11
Table N-14: Statistical Analysis of Scenario 3 Memory Usage N-12
Table N-15: Statistical Analysis of Scenario 2 File Transfer Session Metrics N-13
Table N-16: Statistical Analysis of Scenario 2 Video on Demand Session Metrics N-14
Table N-17: Statistical Analysis of Scenario 2 Social Networking Web Session Metrics N-15
Table N-18: Statistical Analysis of Scenario 2 Work Web Session Metrics N-16
Table N-19: Statistical Analysis of Scenario 3 Sam Video on Demand Session Metrics N-17
Table N-20: Statistical Analysis of Scenario 3 Cathy Video on Demand Session Metrics N-18
Table N-21: Statistical Analysis of Scenario 3 Social Networking Web Session Metrics N-19
Table N-22: Statistical Analysis of Scenario 3 Work Web Session Metrics N-20
Table N-23: Optimization Effectiveness, Scenario 2 ..... N-21
Table N-24: Optimization Effectiveness, Scenario 3 ..... N-22
Table N-25: Statistical Analysis of Scenario 2 File Transfer Session Metrics N-23
Table N-26: Statistical Analysis of Scenario 2 Video on Demand Session Metrics N-24
Table N-27: Statistical Analysis of Scenario 2 Social Networking Web Session Metrics N-25
Table N-28: Statistical Analysis of Scenario 2 Work Web Session Metrics N-26
Table N-29: Statistical Analysis of Scenario 3 Priority 2 Video on Demand Session Metrics N-27
Table N-30: Statistical Analysis of Scenario 3 Priority 5 Video on Demand Session Metrics N-28
Table N-31: Statistical Analysis of Scenario 3 Social Networking Web Session Metrics N-29
Table N-32: Statistical Analysis of Scenario 3 Work Web Session Metrics N-30
Table N-33: Throttling in Scenario 2 Sessions N-31
Table N-34: Throttling in Scenario 3 Sessions N-33
Table N-35: Throttling Spreads during Scenario 2 Session Starts, Minute 0-22 ..... N-35
Table N-36: Throttling Spreads during Scenario 2 Session Terminations, Minute 44-60 ..... N-35
Table N-37: Throttling Spreads during Scenario 3 Impairment Increase, Minute 12-30 ..... N-35
Table N-38: Throttling Spreads during Scenario 3 Impairment Decrease, Minute 30-60 ..... N-35
Table N-39: Scenario 2 Sessions N-36
Table N-40: Scenario 3 Sessions N-36
Table N-41: Aesop Monitoring Accuracy N-37
Table N-42: Aesop Analysis Accuracy N-37
Table N-43: Aesop Optimization Accuracy N-38
Table N-44: Statistical Analysis of Scenario 2 Monitoring Times N-39
Table N-45: Statistical Analysis of Scenario 2 Analysis Times N-40
Table N-46: Statistical Analysis of Scenario 2 Optimization Times N-41
Table N-47: Statistical Analysis of Scenario 2 CPU Usage N-42
Table N-48: Statistical Analysis of Scenario 2 Memory Usage N-43
Table N-49: Statistical Analysis of Scenario 3 Monitoring Times N-44
Table N-50: Statistical Analysis of Scenario 3 Analysis Times N-45
Table N-51: Statistical Analysis of Scenario 2 Optimization Times N-46
Table N-52: Statistical Analysis of Scenario 3 CPU Usage N-47
Table N-53: Statistical Analysis of Scenario 3 Memory Usage N-48

Abbreviations

ACS.....Auto Configuration Server
Aesop .....An approach for optimization of end user service delivery
API.....Application Programming Interface
CIM.....Common Information Model
CLI .....Command Line Interface
DEN-ng.....Directory Enabled Networking, next generation
DMTF .....Distributed Management Task Force
DSL.....Digital Subscriber Line
EUSAO .....End User Service Analysis and Optimization ontology
FOAF.....Friend Of A Friend
FOCALE.....The Foundation Observation Comparison Action Learn rEason architecture
GPRS.....General Packet Radio Service
GPS.....Global Positioning System
GSQR .....Generic Service Quality Reporting
HAN .....Home Area Network
IMEI .....International Mobile Station Equipment Identity
IP.....Internet Protocol
IPTV.....Internet Protocol TeleVision
IQR .....Interquartile Range
IRP.....Integration Reference Point
ISP .....Internet Service Provider
JSON.....JavaScript Object Notation
JVM.....Java Virtual Machine
KPI.....Key Performance Indicator
KQI .....Key Quality Indicator
LTE.....Long term Evolution
MANET .....Mobile Ad-hoc NETwork
MAPE .....Monitor, Analyse, Plan, Execute
MIB .....Management Information Base
NETCONF .....NETwork CONFiguration protocol
NGMN .....Next Generation Mobile Networks
NGN .....Next Generation Networks
NIC .....Network Interface Card
NTP .....Network Time Protocol
OSI.....Open Systems Interconnection
PCRF.....Policy and Charging Rules Function
QoE .....Quality of Experience
QoS.....Quality of Service
RDF .....Resource Description Framework
RMI .....Remote Method Invocation
RTSP.....Real Time Streaming Protocol
RUP .....Rational Unified Process
SAWSDL.....Semantic Annotations for WSDL
SBVR.....Semantics of Business Vocabulary and Business Rules

SECCO .....Service Experience and Context Collection framework
SGSN.....Serving GPRS Support Node
SLA.....Service Level Agreement
SLS.....Service Level Specification
SMI .....Structure and Identification of Management Information
SNMP.....Simple Network Management Protocol
SPARQL.....SPARQL Protocol and RDF Query Language
SQL.....Structured Query Language
SWRL .....Semantic Web Rule Language
UE .....User Equipment
UI .....User Interface
UML .....Unified Modelling Language
UMTS.....Universal Mobile Telecommunications System
URI .....Universal Resource Identifier
URL .....Universal Resource Locator
VLC.....The Videolan media player
WSDL .....Web ServicesDiscription Language
WSMO .....Web Service Modelling Ontology
XML.....Extensible Markup Language
XMPP.....Extensible Messaging and Presence Protocol
XPath.....XML Path language
XSL .....Extensible Stylesheet Language

Chapter 1
Introduction

1.1 Motivation

The NGMN [NGMN, 2007a] and the TM-Forum [TMF, 2009] have identified **automated optimization of end user service delivery** as an important feature in telecommunication networks [Aliu et al., 2013]. Automated optimization of end user service delivery presents three challenges [NGMN, 2007a]. Firstly, the *service expectations* for a set of services at the service consumption point must be set, agreed and actively managed. Secondly, once those expectations have been agreed, the actual *service experience* of service users must be monitored and analysed. Thirdly, changes to the *service delivery context* must be planned and executed to optimize service delivery. Any system that addresses these challenges must be adaptable, highly flexible, and operate with minimal human intervention [TMF, 2009], and operate in *near real time*, defined by the NGMN (2007b) as being a very small number of minutes.

Autonomic Management [Mortier & Kiciman, 2006] is the application of the autonomic computing [Kephart & Chess, 2003] reference architecture to automation of applications in management systems. An autonomic management system for optimization of end user service delivery must continuously execute a MAPE (Monitor, Analyse, Plan, and Execute) loop in near real time. *Monitoring* of service experience and context is often automated in current systems using mechanisms such as 3GPP Performance Management [3GPP, 2012e], SNMP [IETF, 2002] and terminal reporting [IETF, 2003a][3GPP, 2010b]. *Execution* of optimization for end user service delivery may be deployed using approaches such as NETCONF [IETF, 2006b] or 3GPP configuration management [3GPP, 2011a]. *Analysis* and *Planning* of end user service optimization is not generally automated [Sterle et al., 2011], but rather executed as a manual task with support from tooling [NGMN, 2007a][Batteram et al., 2010]. Approaches for automated analysis and planning of optimizations for end user service delivery have been described for specific network types [Sterle et al., 2011]

and services [Oyman & Singh, 2012][Latre et al., 2009] but an approach applicable to arbitrary network types and services has not yet been proposed.

A system for autonomic optimization of end user service delivery must have a holistic view of service expectations, experience, and context [Toutain et al., 2011]; using a model that is aware of the complex and dynamic relationships [Stankiewicz et al., 2011] between concepts. While there are many models for telecommunication networks [IETF, 2002][IETF, 2006b][3GPP, 2009c], services [TMF, 2011][Leijon et al., 2008] and service expectations [TMF, 2011][Lamanna et al., 2003], each model is disjoint, describing the structure and syntax of its own domain in isolation.

Semantic modelling shows promise in providing the holistic view necessary for a system for autonomic optimization of end user service delivery. Semantic modelling allows the structure, meaning, and references of models to be captured using ontologies [Vitvar et al., 2007b], and has already been successfully applied for modelling in health care [Feigenbaum et al., 2007] and web searching [Singhal, 2012]. The use of semantic approaches in solving modelling problems in the telecommunication domain has been proposed [Strassner et al., 2007][Lopez de Vergara et al., 2009] and has been applied to some problems in the telecommunication domain [Rana & Jennings, 2012][Seo et al., 2011]. Indeed, Toutain et al. (2011) suggest using semantics to model end user services. However, concerns have been expressed on the difficulty of designing comprehensive working ontologies for problem domains [Lopez de Vergara et al., 2009], the challenges of building a full working semantic system [Heitmann et al., 2009][Strassner et al., 2007], and the time performance of semantic technologies [Keeney et al., 2011].

1.2 Research Question and Objectives

This thesis addresses the following research question:

"To what extent can a semantic model of end user service expectations, experience, and context be used to autonomically optimize end user service session delivery in near real time?"

End user services in this thesis are services such as Web Browsing and Video on Demand that are executed by end users. *Optimization of session delivery* in the context of this thesis means providing the highest possible level of compliance of a set of service sessions with their expectations. In this thesis, it was chosen to model the *semantic model* using an ontology-based approach [Allemang & Hendler, 2008]. An

ontology-based approach was chosen because ontologies have the expressive power to capture the complex relationships in the end user service management domain. In particular, an ontology for End User Service Analysis and Optimization (EUSAO) was designed. *Autonomic* [Kephart & Chess, 2003] *optimization in near real time* is defined in this thesis as execution of an iteration of the MAPE loop in less than five minutes [NGMN, 2007b].

Four specific research objectives have been derived in order to support answering the research question above. They are:

Objective 1: Identify approaches that are used to manage end user services and technologies that are applicable to autonomic semantic-based optimization of end user service session delivery.

Objective 2: Establish the EUSAO ontology, a semantic model of end user service expectations, experience, and context.

Objective 3: Design a framework that uses semantic lifting[1], querying, rules, and reasoning upon the EUSAO ontology, to produce an analysis of the service experience and context reported by service users in near real time.

Objective 4: Develop semantic mechanisms using the EUSAO ontology that can be used to optimize service delivery in near real time by re-allocating network resources to high priority services when the available network resources are insufficient to optimally deliver all running service sessions.

Objective 1 addresses examining the background and related work for the scope of the research question. End-user service management research and practice was surveyed to confirm that research into autonomic analysis and optimization is required. A survey of end-user service modelling was conducted to provide a foundation for the research towards Objective 2, Objectives 3 and 4 were underpinned by a further survey on the application of semantic techniques to network management. In addition to those surveys, a study was conducted to establish what the most common end user services are, and what levels of

[1] Semantic lifting is an approach used to translate information into knowledge in a semantic form. For example, XML schemas can be annotated with semantic references and mappings that translate information in XML elements to semantic knowledge.

expectations are realistic for those services. That study goes on to examine how those services are affected by network impairments and how they are likely to perform in two of the most common networking domains that terminals operate in; DSL connected Home Area Networks and Mobile Broadband Connected terminals [OECD, 2011a].

Objective 2 addresses the *"semantic model of end user service expectations, experience, and context"* aspect of the research question. The EUSAO ontology meets the requirement of modelling the domain of end user service management by building new concepts and relationships for that domain that are linked heavily to concepts and relationships of existing models identified in the survey carried out to meet Objective 1.

Objective 3 addresses the entire research question with the initial autonomic scope of *monitoring* and *analysis*. This objective was achieved by developing the **Aesop** autonomic end user service management system, a semantic-based system that uses the EUSAO model to structure its knowledge base. Aesop uses semantic lifting to capture service experience and context from terminal reports and stores that knowledge into the knowledge base. Aesop then uses semantic queries and rules to analyse the compliance of service experience with service expectations.

The entire research question with full autonomic scope is the subject of Objective 4. To meet this objective, an optimization planning and execution mechanism was added to the Aesop engine. That mechanism plans optimizations by examining the compliance of running service sessions to determine if the expectations of high priority sessions are being met and executes actions to the service delivery network to prioritize delivery of those sessions if their expectations are not being met.

1.3 Research Process and Approach

A state of the art study was conducted to determine the state of automated end user service optimization in research and operational systems. The study revealed that measurement monitoring and optimization execution is often automated, while optimization analysis and planning for end user services is largely executed as a manual task with support from tooling. Where automation exists, it is implemented in very specific network domains and applications. Candidate models, vocabularies, and ontologies for capturing service expectations and context were surveyed. Finally,

approaches being used in ongoing semantic research in network management in general and autonomic management in particular were examined.

Following the state of the art study, the research work was structured into three further phases, as shown in Figure 1-1. The Semantic Representation Phase developed the EUSAO ontology and explored monitoring by lifting knowledge into that ontology. The Semantic Service Analysis Phase focused on building the Aesop framework and on building the algorithms for semantic service analysis. The Semantic Service Optimization phase developed the optimization planning algorithms and the mechanisms for executing traffic controls on the network.

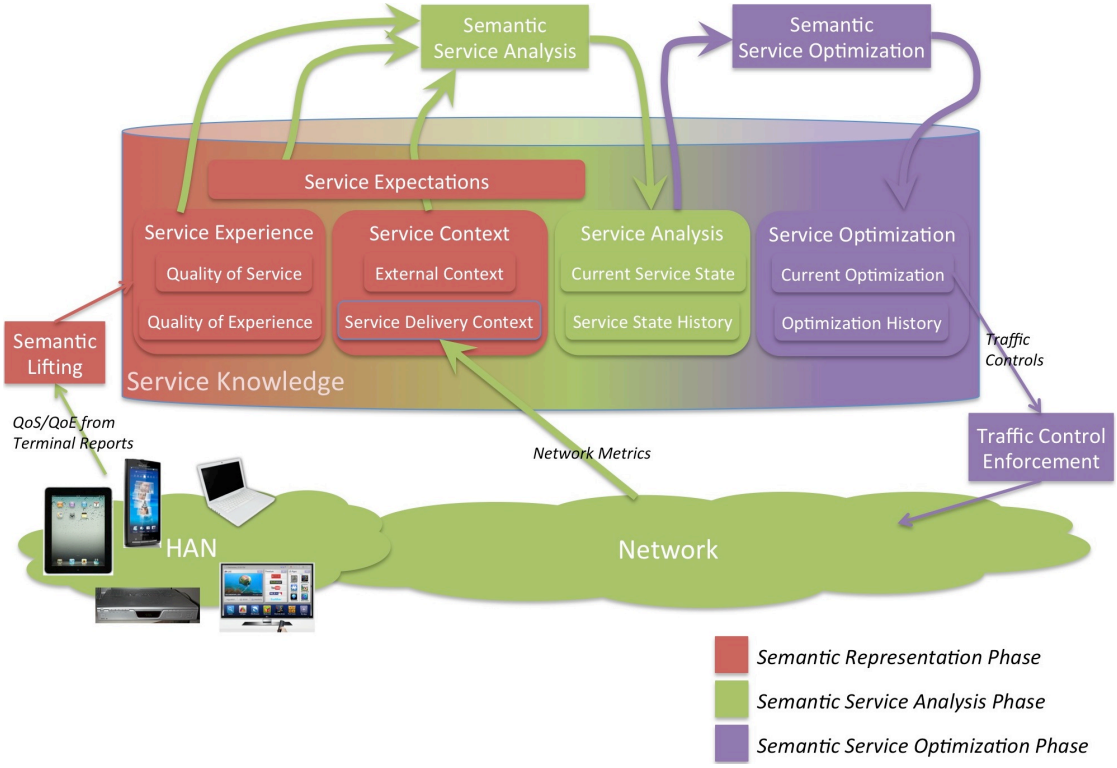


Figure 1-1: Research Phases

In order to conduct realistic experimentation, a test framework for running end user sessions was required. The **Service Experience and Context COllection (SECCO)** framework was developed to fulfil three roles. Firstly, it allows end-user service sessions to be run in a controlled manner. Secondly, it records end user service experience and context for running sessions. Thirdly, it compiles terminal reports for end user sessions and forwards them to Aesop for processing.

The research was organised as shown in Figure 1-2, with each phase incrementally producing research artefacts. Experimentation for each phase was conducted using the software research artefacts developed in that phase and the preceding phases.

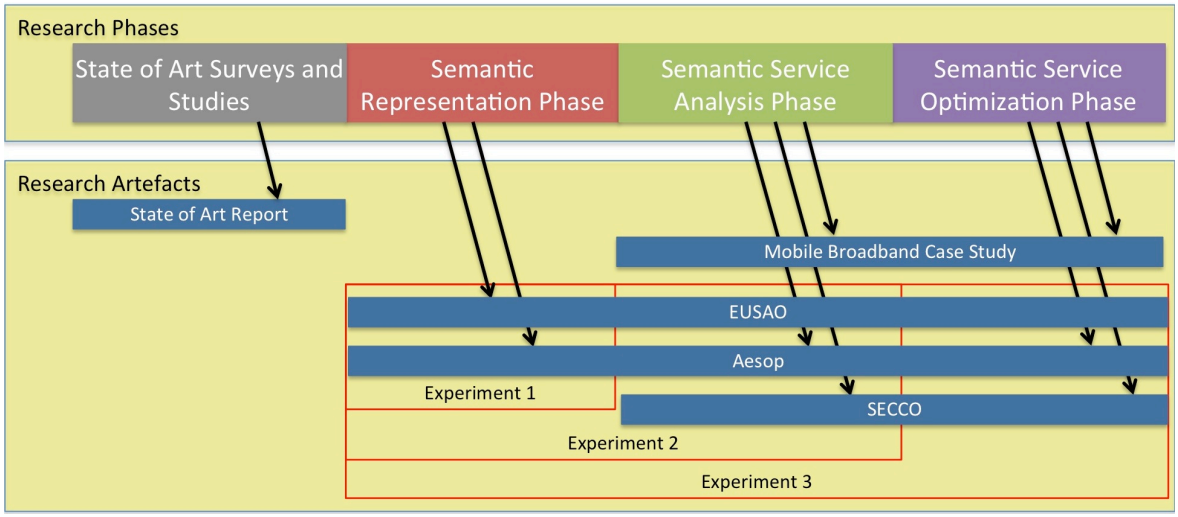


Figure 1-2: Organization of Research

1.3.1 Research Artefacts

The research produced the following five main research artefacts:

1. The **State of Art Report.** This report fulfils Objective 1 and appears as Chapter 2 in this thesis.
2. The **EUSAO** ontology. This ontology is the model produced to meet Objective 2. The design is described in detail in Chapter 3 of this thesis.
3. The **Aesop** autonomic end user service management system. Aesop is the software entity that delivers the system that fulfils Objective 3 and Objective 4, and is described in detail in Chapter 4 of this thesis.
4. The **SECCO** Service Experience and Context COllection framework. This framework is used in experimentation to evaluate Objectives 3 and 4 and is described in Chapter 5 of this thesis.
5. The **Mobile Broadband Case Study**. This case study evaluates Objectives 2, 3 and 4, demonstrating the applicability of this research in the Mobile Broadband domain and it is presented in Chapter 7.

1.3.2 The Research Phases

This sub-section explains the research carried out in each research phase.

1.3.2.1 The Semantic Representation Phase

This phase of the research was executed first. It delivered the knowledge base, the EUSAO ontology, and the software components that monitor end user services by semantically lifting service experience knowledge from terminal reports into the knowledge base.

The core of the EUSAO ontology was built to represent the main concepts required for end user service management and the relationships between them; specifically the concepts of service expectation, service experience, and service context. Those concepts reference existing ontologies for some contextual aspects, specifically FOAF [FOAF, 2010] for user context and WGS84 [W3C, 2006b] for positional context. The concepts of a *snapshot* and a *snapshot bucket* were introduced in the EUSAO ontology, and are fundamental for enabling efficient access to the knowledge base. These concepts enable partitioning of the knowledge base into separate models at run time. Each individual set of user session metrics is a snapshot and all metrics reported in a time period are collected into a snapshot bucket. Each snapshot bucket is held as a separate model in the knowledge base.

Based on insights gained during the state of the art study, SAWSDL [W3C, 2007] was selected to extract and encode semantic knowledge from terminal reports. Existing SAWSDL implementations do not support non-WSDL [W3C, 2001] XML schemas, so a SAWSDL framework for extracting and encoding semantic knowledge from streams of XML terminal reports was developed. Mappings were developed to translate data from XML elements into EUSAO service experience and context concept instances in RDF [W3C, 2004a] format for storage in the knowledge base.

1.3.2.2 The Semantic Service Analysis Phase

In this phase, the focus of the research was on developing the structure of the Aesop knowledge base and components, as well as the algorithms for Semantic Analysis of end user services. In addition, the session control and terminal reporting features of the SECCO framework were developed.

The Aesop knowledge base is structured as a series of semantic models, with a separate model being used for each snapshot bucket. The Aesop components monitor and store knowledge into the knowledge base and that execute optimization analysis, planning, and execution operations periodically on the knowledge base.

Aesop uses algorithms implemented using semantic queries to analyse service sessions for compliance with expectations specified by the subscriber to a service. The metrics for each service session are stored as snapshots in snapshot buckets. The analysis operation executes periodically to analyse the most recent snapshot buckets and to update the compliance of each service session in its snapshot bucket.

The main body of the SECCO framework was developed in this phase. A SECCO collection entity runs on each terminal and collects service experience from each session running on a terminal. The service experience for each session is collated with the service context information from the terminal into a terminal report, which is forwarded to Aesop for lifting as knowledge into its knowledge base.

1.3.2.3 The Semantic Service Optimization Phase

The research in this phase developed the algorithms for planning optimization of end user service quality, implemented those algorithms in Aesop, and implemented execution of optimization traffic controls in Aesop.

The Semantic Service Optimization planning algorithms act on service sessions identified by Semantic Service Analysis as not complying with service expectations. Semantic reasoning is applied to prioritize service sessions and to determine which of those sessions should get priority for access to network resources. Optimizations are executed on service sessions determined as being low priority using throttling. Deployed optimizations are represented semantically, and the effectiveness of those optimizations is tracked over time.

Aesop was extended with components to run semantic optimization planning and execution in this phase. Aesop runs semantic optimization planning as a series of queries that are executed on a reasoned model made up of the most recent sequence of snapshot buckets. The Aesop feature to execute application or release of throttling actions was also developed in this phase.

1.3.3 Evaluation

The evaluation was carried out as a series of three experiments and a case study. The goal of the experimentation was to evaluate the proposed Aesop approach in the experimental domain, specifically addressing the *"to what extent…"* and *"…in near real time"* facets of the research question. The goal of the case study, which examines the applicability of the approach in the mobile broadband access domain, was undertaken to show that the proposed approach was applicable in a second networking domain.

The experiments were designed to progressively assess the autonomic MAPE functions of Aesop. Experiment 1 evaluated Aesop-M, the semantic monitoring mechanism of Aesop. Experiment 2 evaluated Aesop-MA, with Aesop running as a

Semantic Service Analysis system carrying out monitoring and analysis. Experiment 3 evaluated Aesop-MAPE, with Aesop running as a Semantic Service Analysis and Optimization system, carrying out monitoring, analysis, planning, and execution.

Four properties (shown in Table 1-1) of the Aesop approach were evaluated in order to perform a quantitative assessment of Aesop.

Table 1-1: Aesop Evaluation Properties Assessed during Experimentation

Property	Description	Scope	Experiments
Effectiveness	A measure of how much the system improves the compliance of running service sessions with expectations	Full System	Experiment 3
Consistency	A measure of how similar optimization actions taken in different experimental runs of the same scenario were	Full System	Experiment 3
Accuracy	A measure of how well an autonomic function is performed; such as how many sessions did analysis calculate compliance for correctly.	Each of the MAPE functions	Experiment 2 Experiment 3
Performance	A measure of how much time taken to perform an autonomic function and how much system resources were used	Each of the MAPE functions	Experiment 1 Experiment 2 Experiment 3

The properties above were selected to evaluate the Aesop approach from the *"to what extent"* and *"in near real time"* aspects of the research question. *Effectiveness* and *consistency* are measured on the entire system and measure to what extent the system as a whole optimizes end user service session delivery. *Accuracy* and *performance* are measured separately on each of the MAPE functions, assessing how well each function works and the amount of time taken for each function to execute.

Experiment 1 was executed in a simulated environment because, in addition to its primary goal of evaluating the performance of semantic monitoring using the Aesop approach, it was used to gauge the effectiveness of the first part of the approach before a commitment was made to carry out further research and build experimental frameworks and test beds. Experiments 2 and 3 were executed in a full test HAN test bed, with SECCO being used to control end user sessions, for collection of end user service experience and context and for terminal reporting.

The domain of a DSL-connected Home Area Network (HAN) was chosen for experimentation because such networks are very common [OECD, 2011a], the characteristics and limitations of end user service delivery in HANs are well

understood [OFCOM, 2012] [Sundaresan et al., 2011] [Dobrian et al., 2011], and because it is feasible and inexpensive to set up a realistic HAN test bed.

Three scenarios were explored in experimentation, a *degradation* scenario where a session deteriorates from a perfect to an unusable level, a *congestion* scenario where the connection from the HAN to the Internet becomes congested due to over use by end user services, and an *impairment* scenario where a temporary impairment occurs on connection between the HAN and the Internet causing end user services to degrade. Web Browsing, Video on Demand and File Transfer were chosen as end user services because they are amongst the most common end user services used on the Internet today [Maier et al., 2009][Sandvine, 2012].

The case study on the application of the approach in the domain of Mobile Broadband Access networks demonstrated that the structure of the EUSAO ontology, architecture and structure of the Aesop approach could be applied unchanged to optimize end user service session delivery in that domain. The object and data properties of some concepts would be amended to represent Mobile Broadband Access knowledge. Extensions to the Aesop implementation are suggested in the case study to allow a deployment that could scale to manage a full Mobile Broadband Access Network. The case study, in describing Aesop's application in a second network type, provides strong evidence of Aesop's applicability in arbitrary network types.

1.4 Contributions

The contributions of this research are:

Major Contribution: The Semantic Service Analysis and Optimization (Aesop) approach. The approach uses the EUSAO ontology to model end user service management, and the Aesop autonomic end user service management system that uses an entirely semantic processing approach to analyse and optimize end user services.

Minor Contribution: The development of the SECCO framework. SECCO allows controlled execution of end user service sessions, collection and forwarding of terminal reports from those sessions, and implements throttling of those sessions.

The **major contribution** of this research when compared with the state of the art is that it is the first complete autonomic end user service analysis and optimization

system built entirely using semantic approaches. The research has shown that semantic techniques are applicable to analyse and optimize end user service delivery in near real time as service expectations and service delivery context changes. This approach is end user-centric, it enables automation of end user service management from a Quality of Experience (QoE) point of view, a major step forward compared to the manual and static Quality of Service (QoS)-centric approaches being used and proposed in research to date.

The EUSAO ontology is an end user service management model that can be used to manage any end user service set in a networking domain. The ontology captures the concepts, relationships, constraints, and complexity of end user service management. Its novel snapshot and snapshot bucket structure for metrics is designed to be transparent to the services and networking domains being managed whilst enabling efficient knowledge storage and retrieval. The model uses references to the TM Forum SID [TMF, 2011] for networking domain, user, and service provider concepts thus allowing it to be applied to any networking domain.

The domain specific logic in the Aesop approach is implemented using semantic techniques, without any domain specific source code. Monitoring mappings can be written to lift any arbitrary textual data source of service experience or context into the knowledge base. The analysis and optimization planning algorithms can be modified to consider new concepts and relationships by just changing semantic queries and rules. The throttling parameters used in optimization execution are configured in the EUSAO ontology. An Aesop engine would be aware of just the concepts of *service*, *service session*, and *compliance*. How the compliance of a service session is calculated and what service sessions are selected for throttling actions are matters for the optimization analysis and planning queries and rules that are invoked and fired by an Aesop engine. Because the complexity of the domain is captured in the ontology, the mappings, rules, and queries are relatively straightforward to implement.

The knowledge base in the Aesop approach is designed to use partitioned models for storage of individuals based on the EUSAO snapshot and snapshot bucket structure, and thus breaks new ground and is the key to an implementation's efficiency of storage and access. Because knowledge can be partitioned thus, storage operations operate with the scope of a single snapshot bucket and therefore only use one semantic model to store metrics. Analysis and optimization operations are also

scoped to use only the semantic models that cover the time period over which analysis and optimization is being carried out.

Aesop has been developed and implemented on commodity hardware using readily available semantic frameworks, libraries, and toolsets (Jena, Protégé, Pellet reasoner). This work shows the practicability and limitations of developing a semantic management system using the currently available semantic ecosystem and is a benchmark that can be used by further research efforts in the field.

The SECCO framework is a **minor contribution** of this work. SECCO makes it easy for users to set up, run, and monitor end user service sessions. To a user of SECCO, specification, execution, and monitoring of different types of end user services is carried out in a uniform manner. The motivation for SECCO is that it shields the user from the intricacies of mechanisms and interfaces provided by different end user service delivery clients such as web browsers and media players. All end user service sessions are handled by SECCO, without the need for writing complex scripts to launch clients and extract measurements from them.

SECCO is designed to be a highly customizable framework, so that addition of support for new service delivery clients is straightforward. It provides a customizable mechanism that allows end-user service sessions to be run in a consistent and controlled manner. Secondly it provides support for recording end user service delivery and context at the service delivery client. Thirdly it collects measurements and compiles terminal reports for all end user sessions on a host and forwards them to Aesop for processing.

A patent proposal on the Aesop approach that specifically addresses the partitioning structure of the knowledge base, the components of Aesop, and the semantic monitoring and analysis algorithms has been accepted for filing under the Intellectual Property review process of Ericsson and has been provisionally filed.

A further patent proposal that addresses the Aesop optimization algorithm and the application of the Aesop approach for optimization of end user service delivery in mobile broadband access networks has been submitted to the Intellectual Property review process of Ericsson and is currently under consideration.

Three papers have been published to date on research presented in this thesis:

L. Fallon and D. O'Sullivan. "Aesop: A semantic system for autonomic management of end-user service quality". In *Integrated Network Management, 2013. IM '13. IFIP/IEEE International Symposium on*, to Appear. IM 2013, May 2013.

This paper describes the Aesop approach, the partitioning structure of the knowledge base, and the components of Aesop. The paper describes the end user service analysis algorithm and presents an evaluation of the performance of Aesop.

L. Fallon and D. O'Sullivan. "Using a semantic knowledge base for communication service quality management in home area networks". In *Network Operations and Management Symposium*, 2012. NOMS 2012. 13th IEEE/IFIP. NOMS 2012, April 2012.

This paper describes the EUSAO ontology and its structure for holding end user service knowledge. The paper also describes the Aesop lifting mechanism and presents an evaluation of the performance of that mechanism.

L. Fallon, Y. Huang, and D. O'Sullivan. "Towards automated analysis and optimization of multimedia streaming services using clustering and semantic techniques". In R. Brennan, J. Fleck, and S. van der Meer, editors, *Modelling Autonomic Communication Environments*, volume 6473 of *Lecture Notes in Computer Science*, pages 12–23. Springer, October 2010.

This paper presents a high-level summary of the approach being proposed in this research. An autonomic approach to end user service management is proposed, with methods being presented for service session analysis and optimization.

A further paper describing the SECCO framework is planned for submission to the Experience Sessions of the IEEE/IFIP Network Operations and Management Symposium (NOMS 2014).

A journal article based on this thesis is planned for submission to the Journal of Network and Service Management (JNSM).

1.5 Thesis Overview

This thesis is structured as follows:

Chapter 1 introduces the thesis.

Chapter 2 describes the background and related work for this research. It examines end user services, and how different end user services are impacted by impairments in network performance. It identifies applicable models, vocabularies, and ontologies in the field of service quality management are. It looks at how semantics have been used for service analysis and optimization to date.

Chapter 3 describes the EUSAO ontology, explaining the structure, concepts, and relationships used to model the end user service management domain, as well as the snapshot concepts introduced to facilitate model partitioning.

Chapter 4 describes the Aesop autonomic end user service management system, describing the partitioning scheme used in the knowledge base and the components used to execute the various operations required for end user service management.

Chapter 5 describes the SECCO framework for end user service session control, terminal reporting, and session throttling.

Chapter 6 describes the experimentation carried out during the research. It details the simulated experiment, and experiments carried out on the experimental test bed.

Chapter 7 describes how the approach that is the subject of this research can be applied to end user services running in the Mobile Broadband Access networking domain.

Chapter 8 brings the thesis to a conclusion. It sums up the contributions and outlines how future work can further the research described herein.

A total of fourteen of appendices are attached at the end of the thesis. They provide detailed information on end user service delivery in DSL and mobile access networks, and give details of the implementation of the EUSAO ontology and the Aesop prototype implementation. They include the experimental environment, scenarios and run logs. Details of the parameters collected during experimentation are provided and samples of collected results are included. Examples of scripts used to analyse experimental data are provided, as are the detailed experimental results.

An optical disk is submitted with this thesis, which contains all source code, ontologies, mappings, queries and rules developed in the course of this work. It also includes all raw experimental results as XML documents, logs, and database dumps as well as scripts used to process and analyse that raw experimental data.

Chapter 2
Background and Related Work

2.1 Introduction

The research question being addressed by this thesis is:

"To what extent can a semantic model of end user service expectations, experience, and context be used to autonomically optimize end user service session delivery in near real time?"

In order to understand how to *optimize end user service session delivery*, one must first understand the *expectations* and *experience* an *end user* is likely to have on services, and what the *context* of service delivery is likely to be. The background material in Section 2.2 explores these topics.

In Section 2.3, related work is examined. Related work on end user service management is described in Section 2.3.1, covering *monitoring*, *analysis*, and optimization *planning* and *execution* for end user services. Section 2.3.2 surveys modelling work in the domain of end user service delivery, including approaches that use *semantic models*. Work on the application of *semantic* and *autonomic techniques* in related domains is described in Section 2.3.3 and Section 2.3.4 respectively.

A summary of the background and related work is presented in Section 2.4.

2.2 Background

An end user service is any service with which an end user interacts [Fensel et al., 2011]. End user services are executed by an end user to communicate, to access information, or for entertainment. Examples of end user services are Web Browsing, Video Streaming, or Voice. Telecommunication connectivity services such as DSL connectivity at certain bit rates, packet switching, or Internet connectivity are connectivity services and, while they may be used to carry an end user service, are not an end user service in themselves. End users judge the quality of a network not on its ability to transport data but rather on its ability to execute service sessions at a level that they find acceptable [Schatz et al., 2011].

The TM Forum technical report, "Managing the Quality of Customer Experience" [TMF, 2009], provides a comprehensive analysis of customer experience management, laying out requirements on systems and applications in that domain. The report points out that end user services must be provided *"with the optimal level of quality on an end to end service basis"*. The report states that, in order to manage customer experience, the *service expectations* for a set of services at the service consumption point must be set, agreed and actively managed. It highlights the importance of monitoring the *service experience* of service sessions of users. The report discusses the importance of *service delivery context*, pointing out that the performance of underlying networks and applications are but two of many factors that affect the perception of a service by a user. The report proposes an autonomic approach for customer experience management, where an ecosystem of autonomous virtualized applications delivers services and each application is self-healing.

The NGMN (Next Generation Mobile Networks) alliance is an organization where leading global mobile operators cooperate to formulate requirements and guide standardization bodies in the field of mobile communication The NGMN has been the driving force behind the concept of SON (Self Organizing Networks) where networks have *"self-configuring and self-optimizing mechanisms"* [NGMN, 2007a] that should be *"automatic or autonomous"*. The NGMN specified optimization of radio parameters [NGMN, 2007b] as a SON use case because they recognized that those parameters *"significantly influence the performance experience"*. For SON, the NGMN define *real time* as being less than one minute and *near real time* as being a small number of minutes [NGMN, 2007b]. The importance of quality of experience in mobile networks is highlighted by recent reports that Apple is certifying LTE mobile networks prior to enabling iPhone 5 devices on networks [Hibberd, 2013]. The survey by Aliu et al. (2013) gives an overview and details of research work in the field of SON.

Setting *service expectations* means striking a balance between quality, cost, and resource efficiency for the delivery of each service [Stankiewicz et al., 2011]. Typically, service stakeholders have differing expectations on services, a typical service user want an excellent service at a low price, and a typical service provider aims to deliver an adequate service at as high a price as possible. Similarly, communication providers strive to carry services as efficiently and cost effectively as possible through their networks for the best possible price.

The *service experience* is the quality of service (QoS) of a service measured at the point of delivery and the quality of experience (QoE) of the service as perceived by the end user [Stankiewicz et al., 2011]. The *service delivery context* is that set of factors which can affect service quality [Staelens et al., 2010]. Toutain et al. (2011) categorize five dimensions of service context: User, Network, Social, Physical and Device. Telecommunication service delivery context is usually managed quite closely, with a separate system managing each context dimension.

Web browsing, video on demand, and file transfer are the three end user services that generate the most traffic on DSL access networks [Maier et al., 2009] and mobile broadband access networks [Akamai, 2012] with traffic growth largely driven by the emergence of streaming services over HTTP [Sandvine, 2012]. The way in which impairments in the telecommunication service delivery context can affect the *service experience* of these services is summarised in the following paragraphs. For a more complete analysis, see Section A.3 of Appendix A.

In video on demand service sessions, frame freezes and picture distortion caused by packet loss and errors in the underlying network [Chen et al., 2011] are strongly correlated with drops in *service experience* [Dobrian et al., 2011]. Reductions in video playout quality have a much less severe impact [Hossfeld et al., 2011]. Large delays in the network have little impact on video session service experience because media players use buffering [Bae et al., 2009].

Page load time is the most important metric for measuring the *service experience* of web browsing sessions, as will be explained in Section 2.3.1.2. There is a linear relationship between web page load times and latency, and the relationship on web page load times as throughput decreases is exponential [Sundaresan et al., 2011]. Modern web browsers use parallel communication sessions to increase the speed of web page loads at the expense of using more bandwidth [Ihm & Pai, 2011]. Delay is an important factor in carrying out tasks on interactive web pages; a small increase in delay can cause a large increase in the time to execute a task [Staehle et al., 2008].

The *service experience* of file transfer sessions is directly correlated to download speed. Latency has little impact because for files of any significant size, the transfer time is many orders of magnitude greater than latency.

DSL access networks and mobile broadband access networks are two of the most common access networks used by end users to connect to the Internet, thus

representing an important part of the telecommunication *service delivery context* of end user services. As of 2011, DSL was the dominant fixed broadband access technology worldwide with over 50% of subscribers using it [OECD, 2011a]. In 2011 there were 403 million wireless broadband subscriptions globally, 28% more than the total number of fixed subscriptions and over twice the number of DSL subscriptions [OECD, 2011a]. For this reason, a Home Area Network connected using a DSL access network was selected as the domain of application for evaluation of the prototype implementation of the Aesop approach, described in Chapter 6. In the case study in Chapter 7 that examines the potential strengths and weaknesses of the Aesop approach when applied in another domain, the mobile broadband access network domain was selected as the domain of application.

The access network is the component of the telecommunication *service delivery context* that is most likely to influence end user service experience load [Maier et al., 2009]. Common impairments that can occur in the service delivery context of DSL access networks and mobile broadband access networks that can degrade the *service experience* of end user service sessions are summarised in the following paragraphs. For a fuller analysis, see Sections A.4 and A.5 of Appendix A.

Continuous or transient network impairment events that impact the *service experience* of end user service sessions can occur in DSL access networks. DSL access network connection characteristics such as download speed, latency and packet loss can vary by location and time of day [Sundaresan et al., 2011]. Significant numbers of subscribes can experience download speeds of less than 2Mbit/s [OFCOM, 2012]. DSL access networks contribute the most to overall latency on sessions, and latency can increase to seconds at times of heavy load [Maier et al., 2009].

Mobile broadband access networks exhibit lower bandwidth and higher latency than in fixed networks [OFCOM, 2011], thus making them more susceptible to degradation in *service experience* on end user sessions. The location of the terminal [Laiho et al., 2006] and the amount of traffic in the cell [Johansson et al., 2009] are the most important factors that affect the bandwidth that a session experiences. High latency is inherent in mobile broadband networks [Blajic et al., 2007], which reduces the service experience of conversational services and online gaming. Recent technological improvements have substantially reduced latency values [Laner et al., 2012]. The need for optimization of end user service session delivery in mobile access

networks is even more acute than in DSL access networks, and is another reason why this access type was chosen for examination in the case study in Chapter 7.

Section A.3 of Appendix A describes how the service experience of video on demand, web browsing, and file transfer end user service sessions are degraded by network impairments in DSL access and mobile broadband access networks. The specification of service expectations for experimentation in Section 6.2.2 and of experimental scenarios in Section 6.2.3 was guided by the insights described in that section. The manner in which DSL access network characteristics can vary, detailed in Section A.4 of Appendix A, justifies the experimental scenarios in Section 6.2.3, and was used to design the characteristics of the experimental degraded network connections described in Section 6.2.1.

2.3 Related Work

This section describes work that is related to the domain and technological approach proposed in this research.

Section 2.3.1 examines work related to end user service management to establish the current state of art in service monitoring and analysis, and to establish what approaches have been described for planning and execution of optimization of end user service delivery. In Section 2.3.2 models, vocabularies, and ontologies from practice and research applicable to capture service quality management knowledge are surveyed to identify modelling approaches that can be applied and extended in this research and to establish what degree of fresh modelling work is required to enable autonomic optimization. Section 2.3.3 examines semantic techniques that might be applied to management of end user service delivery, describing experiences from other research that applied those techniques and highlighting some of the capabilities and drawbacks inherent in the techniques. In Section 2.3.4, related work is examined to establish the degree to which semantic techniques have been applied in other autonomic management systems, to determine the extent to which the approaches used in that related work can be applied in this work, and to learn from the achievements and limitations described in that related work.

2.3.1 End User Service Delivery Monitoring, Analysis, and Optimization

This research proposes to take an autonomic approach to optimization of end user service delivery. In this section, related work in the telecommunication management

domain is examined for each of the autonomic MAPE functions. For each function, approaches described in the literature are described and critiqued.

Section 2.3.1.1 examines related work in *monitoring* of end user *service experience* and *service delivery context*. Section 2.3.1.2 describes approaches for setting *service expectations* and for *analysing* how monitored *service experience* complies with those expectations. In Section 2.3.1.3, approaches that have been applied to *plan* and *execute* optimizations on end user service delivery sessions.

2.3.1.1 Service Quality Monitoring

Service quality monitoring is used to monitor the service experience and network related service delivery context of end user services running in the network.

Monitoring of events and statistics from network nodes is a source of data for assessing the network related service delivery context because it measures the quality being delivered by resources carrying services. Metrics such as packet loss, delay, and jitter are available. Monitoring of metrics is ubiquitous and highly standardised in public networks [IETF, 1991] [3GPP, 2012e], but is not widely deployed in home networks [Sventek et al., 2011]. This source of data gives the telecommunication service delivery context, but gives at best an imperfect estimation of end user service experience because it is difficult to deduce end user service experience from network context [Stankiewicz et al., 2011].

Terminal reporting [IETF, 2003a][3GPP, 2010b] is increasingly being used to monitor the quality of service (QoS) and quality of experience (QoE) of end user services. The value of the richness and level of detail of data generated by terminal reports has been recognized by the NGMN [NGMN, 2010], who have made terminal based measurement one of their top recommendations. Service metrics from terminals give measurements of the actual service experience of the user of a service session. Quantitative metrics such as packet loss, delay, and jitter are measurable on the network connection of terminals. Because terminal reporting can be used to monitor service experience as well as service expectations, it is used as a mechanism for supplying information to the prototype implementation of Aesop in experimentation, as explained in Section 6.2.5.

2.3.1.2 Service Quality Analysis

Service quality analysis assesses the quality of an end user service session by determining the compliance of that session with its expectations. In order to carry out that analysis, service expectations must be set, the service experience of service sessions must be determined so that the level of compliance of those service sessions with expectations can be calculated.

2.3.1.2.1 Setting Service Expectations

The quality expectation on service sessions at their point of delivery must be specified using metrics that are appropriate to the service in question. Such metrics are often standardized. The three end user service types being used in the evaluation of the prototype implementation of the Aesop approach are video on demand, web browsing, and file transfer. The following paragraphs discuss standards that can guide the metrics that should be selected and enumerated to set service expectations for sessions of those services in experimentation.

For video services, the ITU-T specifies a large set of requirements that cover functions such as channel switch, pause and rewind as well as video and audio quality [ITU-T, 2008b]. For video and audio quality, they state that a Mean Opinion Score (MOS) may be used but point out that determining MOS values is difficult and that *"reliable objective quality assessment methods have not been established for transmitted video and audio"*. Dobrian et al. (2011) specify five quality metrics for video on demand services. *Join Time* is the time taken from selection of a video to playout of that video starting. *Buffering Ratio* is the proportion of the total session time where playout freezes due to buffering. *Rate of Buffering Events* is the number of buffering events that occur in a session. *Average Bit Rate* is the average rate at which the video stream was downloaded. *Rendering Quality* is the rate in frames per second at which playback occurred.

The ITU-T recommends an opinion model for web browsing applications based purely on session time [ITU-T, 2005a]. The recommendation describes three perceptual regions for response time as being Instantaneous (<=0.1s), Uninterrupted (<=1.0s) and Loss of Attention (>10s). The recommendation observes that, for all session durations *"perceived quality goes down linearly with the logarithm of the session time"*, and recommends the use of the equations in Table 2-1 for calculating

the MOS (Mean Opinion Score) of short, medium, and long web browsing sessions on fast network connections.

The ITU-T model must be used with caution because the raw performance of a web page is not the only criterion that end users will use to assess their service experience. However, in cases where different sessions to the same web site are being compared, the equation is applicable. Page load times appear more frequently than MOS values in literature discussing the performance of web browsing services, Sundaresan et al. (2011) and OFCOM (2012) both use page load times of a test web page as a service experience metric for web browsing.

Download and upload speeds can be used as a service experience metric for file transfer because it is a rather primitive service when compared to services such as video on demand or web browsing.

Of course, the expectation of an end user on a particular session in a particular context is subjective and can vary greatly [Toutain et al., 2011] depending on factors such as the task being carried out, the expectation that particular user has on delay, and the expectation that user has on the provider of the web site [Bouch et al., 2000] [Schatz et al., 2011]. However, objective values for service expectation metrics such as those listed in Table 2-1 can be set and used to check compliance of monitored service experience with expectations during analysis of service experience. The metrics in Table 2-1 were used to guide the specification service expectations for experimentation shown in Table 6-4 in Section 6.2.2.

2.3.1.2.2 Determining Service Experience

If terminal reporting is available, as discussed in Section 2.3.1.1, determining service experience is as trivial an exercise as reading the service experience for service sessions from the relevant terminal reports. However, as terminal reporting has historically not been available and is not always a practicable approach, implemented systems and research work have used other approaches to determine or estimate end user experience.

The use of key performance indicators (KPI) and key quality indicators (KQI) as defined in the TM Forum SLA handbook [TMF, 2010a] is well established for assessing network quality. The metrics from networks or servers that are important for assessing network quality are identified as KPIs. KQIs are aggregates of KPIs, *"usually expressed as a percentage of customers, resources or telecom entities (like a*

call or a session) meeting a certain level of quality" [TMF, 2010a]. KPIs and KQIs are widely researched [Toktar et al., 2007][Bhushan et al., 2004] and deployed in management products [Ericsson, 2012][IBM, 2012b][HP, 2012].

Table 2-1: Summary of Service Expectation Metrics

Service	Parameter	Values	Source
Voice	Speech Transmission Quality Categories	*Best* *High* *Medium* *Low* *Poor*	ITU-T G.109
Video	MOS	*A scale such as 0 to 5*	ITU-T G.1080
	Standard Quality Minimum Video Bit Rate	*15 Mbit/s Constant Bit Rate*	
	Standard Quality Minimum Audio Bit Rate	*384 Kbit/s*	
	Standard Quality Video-Audio Synchronization Lag	*45*	
	High Quality Minimum Video Bit Rate	*3.18 Mbit/s Constant Bit Rate*	
	High Quality Minimum Audio Bit Rate	*384 Kbit/s*	
	High Quality Video-Audio Synchronization Lag	*45*	
	Join Time	*Seconds*	Dobrian et al.
	Buffering Ratio (Freezes)	*Percentage of total session time*	Dobrian et al.
	Rate of Buffering Events (Freezing Rate)	*# Buffer Events* *Session Duration*	Dobrian et al.
	Average Bit Rate	*Kbit/s*	Dobrian et al.
	Rendering Quality	*Frames/s*	Dobrian et al.
Web Browsing	MOS (*sessiontime<10s*)	*4.38–1.30.ln(sessiontime)*	ITU-T G.1030
	MOS (*10s≤sessiontime<30s*)	*4.79–1.03.ln(sessiontime)*	
	MOS (*30s≤sessiontime*)	*5.76–0.948.ln(sessiontime)*	
	Page Load Time	*Seconds*	Practice
File Transfer	Transfer Speed	*Mbit/s*	Practice

One approach for estimating end user service experience is to identify KPIs [Stankiewicz et al., 2011][Toktar et al., 2007][Bhushan et al., 2004] and KQIs [TMF, 2010a] that are particularly important for assessing the quality of particular services. The use of KPIs and KQIs works well when a service is using physical or logical

network resources exclusively because it is possible to map resource KPIs to service KQIs. This has traditionally been the case for the voice service in telecommunication networks. In modern networks where resources are always shared by many services, KPIs and KQIs are more difficult to apply because there are no fixed mappings from resources to services [Staelens et al., 2010][Stankiewicz et al., 2011]. A user survey by Nyberg & Ronnblom (2008) found that QoE values recorded by users correlated poorly with service KPIs for video services. This is a particular problem in the case of counter-based statistics. If a network element merely reports the amount of packets being dropped as a KPI on an interface, it is very difficult to deduce corresponding KQIs if many services are using that interface.

Some research has been undertaken on using a set of standard network parameters such as bit rates and packet loss as input into a mathematical model that calculates a metric indicating the quality of the service. The ITU-T uses this approach to assess voice quality: the E-model [ITU-T, 2008a] uses a set of transmission parameters to calculate the R-factor, a metric ranging from 0 to 100 which gives a linear assessment of speech quality. This approach worked well for voice carried over circuit switched networks because voice is a telecommunication service with a single facet (audio) and circuit switched networks are rather homogeneous in their structure. Although there has been some work in applying models to other services such as video [Yamagishi & Hayashi, 2008], it is more difficult to apply models services which have many facets (audio, video, control etc.) because sub-metrics for the quality of each facet must be determined and then those sub-metrics must be used to calculate an overall metric for the service. The ubiquity of heterogeneous packet switched networks poses another problem for this approach because it is more difficult to build mathematical models for packet switching; the E-model specification [ITU-T, 2008a] describes some limitations of that model in cases of high packet loss and large packet bursts. Another problem with this approach is that the network quality metrics for sessions of the particular service in question must be isolated and input into the model. This is difficult to achieve, again because of the sharing of resources by service sessions [Stankiewicz et al., 2011].

Fiedler et al. (2010) propose *"a generic formula in which QoE and QoS parameters are connected through an exponential relationship, called the IQX hypothesis"*. The results they described using formulae derived based on the IQX hypothesis for the Voice and Web Browsing services are encouraging. However, as acknowledged by the

authors, this approach uses a single impairment factor to calculate a MOS value and MOS is a function of many factors. This approach also assumes that it is possible to isolate the network quality metrics for particular service sessions.

Another approach is to use techniques such as weighting, grouping, and statistics [Batteram et al., 2010][Kim et al., 2009] to estimate the service quality of multimedia services in a network or regions of a network. The drawback of this approach is that the weights, groups, and statistical techniques must be tuned for the particular services, network infrastructure, and service delivery context of each individual operator. In their STMS system, Choi et al. (2010), enhance this approach by correlating session information from multimedia streaming servers with network topology and network metric values to reconstruct service sessions and calculate service quality on the reconstructed sessions. This allows services and service sessions to be analysed. STMS must be tuned for a given operators' service delivery context and the correlation process is time consuming making near real time feedback impossible.

Simoens et al. (2008) evaluated three approaches for determining end user service experience from QoS parameters. Firstly, an analytical approach similar to approaches described above used mathematical equations to estimate end user service experience. This approach worked well in limited cases, but complexity was found to increase rapidly when many complex use cases were introduced. The second approach evaluated was the use of fuzzy logic in a rules based system. This approach was abandoned because it was found that autonomic derivation of rules was impossible, that the rules base expanded rapidly as use cases were added, and that the cross-interaction between variables in rules was difficult to control. The third approach used was the application of neural networks to determine end user service experience. The results obtained using this approach were promising in the use cases evaluated. The drawbacks of this approach is that a neural network must be trained for every end user service type and, as the authors point out, training the neural network is not trivial.

Song et al. (2012) have described a semantic approach for end user service analysis in Home Area Networks. They propose a framework where domain expert knowledge is used to semantically uplift raw monitored data into a form that is understandable to novice users. The uplifted knowledge is semantically processed to detect anomalies using knowledge input by expert users. Once an anomaly has been

detected, it is graphically presented to the user together with an analysis tree that illustrates the root cause of the anomaly.

Analysis of end user service experience and context determines compliance with expectations. If end user service experience is not being directly measured using a technique such as terminal reporting, an approach to estimate or reconstruct service experience from network context must be used. Many approaches have been identified in the literature towards achieving that goal but they all exhibit limitations and none are generally applicable to arbitrary end user services. Therefore, in this research, the prototype implementation of Aesop was evaluated using end user service experience supplied from terminals in terminal reports.

2.3.1.3 Service Quality Optimization Planning and Execution

In this section, related work that examines optimization of end user services is examined. The autonomic functions of *planning* and *execution* are taken together because, as can be seen from the work examined below, most approaches implement both functions together. Sterle et al. (2011) point out that, in the field of end user service delivery optimization, there is little research effort, standardization is weak, and that implementations are *"related to focused and specialised areas"*.

In current practice, service management systems [Ericsson, 2012] [IBM, 2012b] [HP, 2012] identify service constraints in networks and send reports to service and network planning systems recommending service improvements to operations staff. Although SON functions such as the 3GPP Automatic Neighbour Relations (ANR) feature [3GPP, 2009b] deployed in networks today do improve the service delivery context, they are not directly related to end user service optimization.

Sterle et al. (2011) propose an application based QoE controller for NGN (Next Generation Networks) that uses a quality context mathematical model to estimate the likely end user service experience at session setup or modification. The information in the session setup or modification message as well as user and service profiles are used as input to the model, which outputs an estimation of the expected end user service experience. The estimation is used to determine if optimization is necessary and, if so, the session setup or modification message may be modified, policies on the Policy and Charging Rules Function (PCRF) [Balbas et al., 2009] may be adjusted, or the session may be rejected. This approach works well where a central authority is controlling session execution. However, the service experience of the session can only

be adjusted at session setup or modification, and a separate model must be developed for each end user service type. In addition, because the approach assumes a central service control entity exists, this approach cannot be applied to conventional end user service sessions running over the Internet because they have no such central authority [Sventek et al., 2011].

End user service optimization of individual services can be implemented in the mechanisms used to deliver a particular service. For example, in HTTP adaptive streaming, the service implements control messages that allow the client to *"dynamically switch between different bit rate representations of media content as the available bandwidth changes"*, a strategy that is very effective for sessions of that service [Oyman & Singh, 2012]. Such mechanisms optimize end user service experience for a particular service session of a particular service type, ignoring all other end user service sessions.

Latre et al. (2009) propose a tiered autonomic architecture for end user service experience optimization in multimedia access networks. This approach uses a *monitor plane*, a *knowledge plane*, and an *action plane* around a semantic knowledge base [Latre et al., 2008]. The end user service experience of each service is determined using the approaches described in Simoens et al. (2008). A *Learning Controller* in the knowledge plane controls optimization of the currently executing set of end user services. The authors suggest the use of a cost function for monitoring and altering optimization actions but leave further specification of the learning controller for future work. In [Famaey et al., 2010], an approach where this architecture can be extended to co-operate with autonomic managers in other domains using the FOCALE [Jennings et al., 2007] autonomic management architecture is described. End user service optimization across network domains will certainly require cooperation between autonomic entities. However, applying a FOCALE based approach may be difficult where no formal agreement exists between network and service providers such as YouTube or TV catch up services. The FOCALE architecture assumes that participating autonomic entities take a policy-based approach so interworking with autonomic managers that do not use policies may be an issue.

In [Bouten et al., 2012b], the authors examine the issue of coordinated management across HTTP adaptive streaming service sessions raised by Oyman & Singh (2012). They examine how three policies for managing HTTP adaptive

streaming sessions in a segment might affect users with different service levels: *global quality optimization* where the goal is to provide the best average MOS score for all users irrespective of service level, *user-centric optimization* where the MOS score was maximized for users with the highest service levels, and *operator-centric optimization* where the penalties incurred by an operator for service unavailability were minimized. The experimental results show that applying such policies is effective in the case of HTTP adaptive streaming services. User centric optimization and operator centric optimization increase the MOS score of users with higher service levels at the expense of users with lower levels of service because higher levels of service have higher expectations on MOS and higher penalties when adequate MOS scores are not achieved.

Research in the field of optimization in home area networks and mobile broadband networks focuses on optimization of the network resources being used to carry end user services, the service delivery context, rather than on the services themselves.

The Homework project [Sventek et al., 2011] uses an information plane to monitor and persist measurements from home networks. A policy engine that accepts policies from UI applications can apply access control and traffic control policies to the HAN to shape traffic flows through a home gateway according to user expectations using Netfilter Framework (2012) controls. One exemplar application can identify flows in the home area network and can apply traffic controls that limit those flows based on their size, the user that initiated them, or the protocol being carried in the flow. Another exemplar application allows the traffic from particular machines connected to the home area network to be controlled in a similar manner using policies. Using this approach, service experience of end user service sessions is not considered, certain types of end user services consuming certain bandwidth are constrained even though those services may not be affecting the service experience of other services.

Rana et al. (2011) developed an algorithm that translates policies specified on end user services in home area networks into traffic control commands. End user policies are specified semantically as SWRL rules [W3C, 2004c] on end user service concepts in DEN-ng [Strassner, 2002]. The algorithm uses the relationships defined in DEN-ng to map the SWRL policies specified on service concepts into commands on concrete network entities. Traffic controls are implemented by executing those commands on the home gateway. The approach was extended as described in Rana & Jennings (2012) to support triggering of execution of policies from monitored data. Packet

inspection is used to read 5-tuple identifiers from flows crossing the home gateway, which are semantically enriched and stored in the system ontology. When the system ontology is updated with monitored knowledge, a policy selector uses inference to build a list of all SWRL rules that might be fired as a result of the updated knowledge. Each rule is passed through a verification algorithm prior to translation and execution. In this approach, end user service experience is not considered in the decision to constrain end user service sessions.

Aliu et al. (2013) summarize mobile access network research for SON and include a section that summarises research on self optimization. They categorize ongoing self optimization research into three categories: research into *load balancing* of the traffic load in and across cells, research on radio *interference control* between cells, and research on *capacity and coverage* improvement via relaying. The authors do not document any research that examines the quality of experience of end user services running in mobile access networks, but point out that *"self optimization still remains a broad and open area needing more research"*. Approaches such as those described above by Sterle et al. (2011) and Oyman & Singh (2012) are effective for particular end user service types but, as pointed out by Oyman & Singh (2012), *"specialized end to end network protocols, architectures, and algorithms"* are required to manage end user service experience in a holistic manner in mobile broadband access networks.

The related work discussed in this section bears out the observation of Sterle et al. (2011) that optimization of end user service delivery is an immature field in practice and in research. The proposed approaches are not generally applicable to optimization of an end user service session of any type; some assume a central session control mechanisms, others optimize single end user sessions at the expense of all other sessions, and yet others use traffic control of streams without taking end user service experience into account. The research approaches that take a holistic view to end user service management show promise but are as yet immature. The survey in this section shows that there is much research work remaining to be done in the field of optimization of end user service delivery.

2.3.2 Related Models, Vocabularies, and Ontologies

Toutain et al. (2011) make the case for modelling end user services and make the point that for contextual service models, *"the most promising approach uses semantic models based on ontologies".* This section briefly presents some models, vocabularies,

and ontologies from practice and research that are applicable in the domain of end user service management.

Although there is a very long history of modelling in the telecommunication domain, the focus has been largely on models of network equipment and resources rather than on the services being carried by networks. The use of SNMP [IETF, 2002] MIBs described in SMI [IETF, 1999] and the DMTF's CIM [DMTF, 2012] to model equipment and resources is typical of such models. XML is increasingly being used to specify models; the IETF specifies YANG [IETF, 2010c] for NETCONF [IETF, 2006b] models, the 3GPP uses Network Resource Models [3GPP, 2009c], and the Broadband Forum specifies templates [Broadband Forum, 2010b] for modelling devices being managed by TR-069 [Broadband Forum, 2010a]. These models limit their scope to their own networking domain, are structural and syntactic, and often represent semantic relationships in formats that are not machine readable such as flowcharts, interworking diagrams, state transition diagrams, or even textual descriptions.

Some work has been undertaken in modelling telecommunication services. Garschhammer et al. (2001) describe a generic telecommunication service model in UML that identifies and defines service relationships such as user, functionality, and QoS parameters at a high level. Rodosek (2003) uses service template models in UML to describe a service in terms of a generic service model. That service template model can be specialized by providers of a specific service and by a specific service provider for a specific customer. The TM-Forum's SID [TMF, 2011] and DEN-ng [Strassner, 2002] are much more comprehensive UML models. Because the SID and DEN-ng share a common ancestry [Jennings et al., 2007], both model services identically [Davy et al., 2007]. They define *"customer facing services"* as services or service packages a customer buys and *"resource facing services"* as telecommunication connectivity services such as a broadband access connection. They describe relationships between customer facing and resource facing services with *"uses"* and *"requires"* relationships. The SID and DEN-ng focus is on communication services, they do not model end user services, end user service sessions, or any service experience concept. It is, therefore, not possible in SID or DEN-ng to map the QoS measured on resource facing services to individual end user service sessions. UML was developed as a notation for human readable models with support for tooling to allow machines to process the models [OMG, 2010]. Although methods for translating

UML into executable forms exist [Quinn & Matusikowa, 2010], conventional UML models are *"too informal to support execution"* [Sommerville, 2010].

SALmon [Leijon et al., 2008] is a domain specific language for service management. Services, SLAs, and network entities and their properties are defined as objects in the language. The language provides support for specifying relationships between objects and aggregation of objects. The values of properties are updated at run time using expressions specified in the language. These expressions are time aware and allow the use of properties of classes and input values in their specifications. An execution engine runs a SALmon model, keeping track of the status of services using monitored service experience and context. SALmon was used to model and monitor the SLAs for the IPTV service in the MAGNETO project [Handurukande et al., 2011]. The approach of having a programmable run time component managing service sessions is interesting, and the support for time dependent calculations is a powerful feature of SALmon. However, the language is not inherently aware of service management concepts, the user must build up a service model from the beginning for each domain. The model must be programmed manually in the SALmon language; support for importing models from other sources is not available. Another drawback is that the SALmon engine must be present in any system using this approach.

In telecommunication networks, service expectations are modelled as SLAs. The TM Forum SLA handbook [TMF, 2010a] specifies templates and models for telecommunication service level agreements. The SID [TMF, 2011] defines model entities for Customer SLAs and for Service/Provider SLAs in a static formal business-oriented manner on customer facing services. The extension of the SID model in Bhushan [Bhushan et al., 2004] is typical of how a service model references a SLA model in that it ties QoS metrics for a particular resource facing service to a SLA of a customer facing service. Such mappings are useful when SLAs are specified in network QoS terms but are inadequate if compliance with end user service experience is to be assessed [Stankiewicz et al., 2011].

There have been some research initiatives into specification of SLAs in formats that can be processed by machines. SLAng [Lamanna et al., 2003] is an XML-based language for specifying service level agreements that provides a format for automatic negotiation of service expectations between service providers and users. Toktar et al. (2007) describe KISLA (Key-Indicator SLA), another XML model for SLAs that can handle mappings from SLAs to KPIs and KQIs. The OMG's Semantics of Business

Vocabulary and Rules [OMG, 2008] is a specification for formally specifying business facts and rules. SBVR is interpretable in predicate logic. Such approaches could be applied to translate end user service expectations possibly expressed as SLAs into a semantic form that could be inserted as knowledge into an ontology.

In DEN-ng [Strassner, 2002], SLAs are modelled as policies [Raymer et al., 2006] that trickle down through the *policy continuum*. Policies are specified to monitor and apply changes to *"adjust the availability and performance of services and resources"*. There are policies at business level that model the SLA itself, policies on the resource facing services that deliver the customer facing service to which the SLA is applied, and policies on the resources delivering the customer facing services. Sheridan-Smith et al. (2006) describe an extension of this approach where services are defined as policies in a service definition language at each level of the policy continuum. A *domain expert* runs policies at each level, while interacting with the domain experts at the next level below and above it. It is only possible to model end user service expectations as policies and enforce compliance of service delivery in a networking domain with those expectations using a policy based management system if the underlying model on which the policies are built models end user service delivery.

Because semantic modelling is such a powerful approach for expressing structure, meaning, and references [Vitvar et al., 2007b], it is no surprise to find that some research work has been undertaken in using semantic approaches to model services in the telecommunications domain. Although full automation of mappings from non-semantic models such as the SID models to ontologies is considered impractical [Strassner et al., 2007], methods exist for translating concepts and relationships in such models into ontologies [Lopez de Vergara et al., 2009][Quinn & Matusikowa, 2010][Bedini et al., 2011]. The results of such mappings can provide an initial structure of concepts and references for a domain. While Bedini et al. (2011) have reported considerable progress using patterns to implement transformations; they comment that semantic knowledge is usually missing in such transformed ontologies because full semantic information is not present in the originating model.

The FOCALE autonomic management architecture description in Jennings et al. (2007) proposes a knowledge base that uses *"DEN-ng information and data models, combined with domain-specific ontologies for augmenting those models with required semantics"*. It is proposed to use reasoning on those ontologies to infer new knowledge. The Ontology for Support and Management (OSM) [Serrano et al., 2009]

is built to augment information in conventional data models with semantics; allowing user, location, device, and service domain specific context to be added to existing data model entities. OSM allows links between related concepts from data models in different domains to be specified. Reasoning and inference engines can then be used to *"orchestrate interactions between these different domains"*. The use of semantic techniques to augment existing models is very useful for addressing certain problems such as interoperability. However, because the core data model is not semantically enabled, the scope of where semantic approaches can be applied is limited.

The BREIN ontology [Munoz Frutos et al., 2009] is a conceptual ontology that defines basic QoS concepts for telecommunication services. The ontology groups concepts related to QoS into categories: system, network, infrastructure, security, and economical. The BREIN ontology is used for specifying SLAs in a way that allows them to be shared and understood by any customer and provider system; specific customers and providers specialize the ontology for their particular context but because the core concepts remain the same, SLAs remain interoperable. Although this ontology does not model end user service delivery, the approach of using a simple core ontology providing a common understanding of concepts is compelling.

Rodrigues et al. (2012) describe an ontology for managing network services quality by monitoring that the SLSs (Service Level Specifications) for each service are being complied with. Modules deliver the functions of the ontology. A *service management module* is built to represent the SLS of each service being managed. A *network module* represents each networking domains on which services are delivered, with each networking domain modelled as a black box. A *multiservice monitoring module* monitors each service management and networking monitoring module, determining the QoS and performance parameters of the SLSs. End user quality of experience *"has to be identified and mapped into objective and quantifiable QoS parameters"*. This ontology models SLA, SLS, and network QoS concepts and their relationships well. However, it has no concept of end user services, and uses a direct mapping of QoS parameters to quality of experience. Ce Junior et al. (2012) describe NetQoSOnt, which also uses modules to deliver the functions of the ontology. A module is used to model a layer in the OSI model [ITU-T, 1994]. Quality of Experience concepts such as *MOS*, *R Factor* [ITU-T, 2008a], and *Subjective Quality* are defined and are inferred using SWRL rules. The rules implement QoS to QoE mappings for a particular end user service specified by standards bodies. This approach is flexible in

that it is easy to deploy SWRL rules to define different end user service experience evaluations but suffer from the same drawbacks as the approaches discussed in Section 2.3.1.2 that calculate end user service experience from network QoS values: the network quality metrics for sessions of the particular service in question because of the sharing of network resources by service sessions.

Latre et al. (2008) describe an ontology for capturing knowledge that can be used to optimize end user service experience in multimedia access networks. The ontology models network topology information, service and policy information, end user service experience optimization actions, and monitored information using OWL and SWRL rules. Topology, service, policy, and optimization knowledge is populated statically, with monitored knowledge is added during execution. SWRL rules can be added to infer end user service experience *"as long as the rules are not too complex"*. The authors recommend that the approaches for determining end user service experience described in Simoens et al. (2008) be used in complex situations. The ontology uses reasoning heavily and is not structured to optimize performance.

In the Web Services domain, WSMO [Roman et al., 2005] uses the concept of a Goal to express service expectations as user desires, with requirements such as properties, capabilities, mediators, and interfaces being possible to define. Stollberg & Norton (2007) extend WSMO with goal templates that specify the capabilities required of a service. Such approaches are commonly in that domain used to find and/or compose web services that can deliver the capabilities required to meet service expectations.

The context of some service aspects can be captured with a general ontology or vocabulary. The Friend of a Friend vocabulary [FOAF, 2010] can be used to represent users of services. The Semantically Interlinked Online Communities (SIOC) [Bojars et al., 2008] ontology describes social networks. The WGS84 Geo positioning vocabulary [W3C, 2006b] allows the geographical location of a user to be represented. Traffic measurements may be captured with the Monitoring Ontology for IP Traffic (MOI) [Salvador et al., 2010]. Using such known and proven ontologies and vocabularies for such concepts in the domain of end user service management is sensible for many reasons, not least of which is that it facilitates interoperability with other systems.

2.3.3 Semantic Techniques Applicable to End User Service Management

Semantic modelling allows the structure, meaning, and references of models to be captured using ontologies [Vitvar et al., 2007b], which can then be executed.

Operations can be carried out on models directly using semantic frameworks, queries and rules can be run on models, and reasoners can automatically deduce knowledge and relationships from the current set of knowledge in a model. The strength of semantics is that complex relationships across disparate knowledge domains can be modelled, and knowledge can be captured [Allemang & Hendler, 2008].

Ontology mapping is used to map the relationships between concepts in different ontologies. O'Sullivan et al. (2007) describe the OISIN process for developing and deploying such mappings. Keeney et al. (2007) use ontology mappings in a Knowledge Based Network to enable semantic interoperability between producers and consumers of network information. The Knowledge Based Network allows producers to publish and consumers to subscribe to knowledge. Information models describing knowledge to be published into the system are analysed off line to produce deployable runtime mappings. This means that information in any form, once mapped, can be published into or consumed from the system. Reasoning is used to determine the set of mappings to use for a particular event forwarding operation.

Queries and rules are alternatives to reasoning for using semantic knowledge. SPARQL [W3C, 2008] is a query language for RDF documents. SWRL [W3C, 2004c] is a rule language that allows rules to be specified on ontologies. If the conditions in body of a rule hold, then the head conditions also hold.

Semantic lifting can be used to translate information into a semantic form. SAWSDL [W3C, 2007] is used to annotate element definitions in XML schemas with semantic references and mappings that translate information in XML elements into semantic knowledge. SAWSDL is primarily used to annotate WSDL [W3C, 2001] web service definitions [Verma & Sheth, 2007a]. Lehtihet & Agoulmine (2008) describe an approach for reusing information from SMI [IETF, 1999] and CIM [DMTF, 2012] models by mapping those models to a common UML structure, which is then transformed to XML schemas. Those XML schemas are then annotated with SAWSDL. Munoz Frutos et al. (2009) describe an approach where semantic annotation is used to annotate SLA templates that are used in automated contract negotiations in service composition. A service composition QoS ontology is used as a model; services advertise their capabilities using SAWSDL annotations to reference the QoS model.

User service context varies over time so it is important to handle the temporal aspect of context efficiently. Even though time can be encoded as semantic data, time

information is not treated in any special way by reasoners. O'Connor & Das (2011) describe a method for handling temporal information in OWL. They developed a valid-time temporal model that can be associated with a measurement. SWRL rules can then query that model and return measurements that match the SWRL temporal query. Keeney et al. (2010) added temporal support to their Knowledge based Network, allowing temporal based subscriptions to be supported. Reasoning is used to determine when a temporal match occurs.

Strassner et al. (2007), Lopez de Vergara et al. (2009), and Heitmann et al. (2009) outline the advantages and drawbacks of semantic representation of knowledge. Ontologies enable model interoperability, facilitate incremental modelling, are expressively rich, and allow the use of models from various sources, all very desirable qualities for modelling management systems. However, initial costs of building models are substantial, semantic content in models varies in detail; tooling for ontologies is immature, and expressing behaviour in rule languages like SWRL [W3C, 2004c] is difficult. They predict that ontologies will gain acceptance in highly dynamic networking domains such as cognitive networks and MANETs much earlier than in more static networks such as backbone networks where models have already been developed and deployed.

Lopez de Vergara et al. (2009) point out that performance issues arise when using semantic technologies. Large ontologies are difficult to handle in memory and reasoners are slow when they have to work with ontologies with large numbers of individuals. [Latre et al., 2008] found that single read and write operations took 2.2 milliseconds and 5.7 milliseconds respectively on persistence operations using the Protégé API [Stanford University, 2012]. Keeney et al. (2011) carried out experiments on semantic operations using two ontologies with knowledge base sizes varying from small to huge. They report execution times of between 1 and 40 seconds for the Pellet reasoner [Clark & Parsia, 2013] and Jena rules [Apache, 2013c] to perform a knowledge transformation, and between 0.6 and 6 seconds for SPARQL queries [W3C, 2008] to perform the same knowledge transformation as knowledge base sizes increase. It is clear, given these metrics, that any system using such semantic tools must be designed with care in order to give adequate performance.

Tai et al. (2011a) describe COROR, an OWL reasoner that runs on resource constrained devices. COROR uses algorithms that dynamically dimension it at run time to match the *"features of the particular semantic application"* for which it is being

used. A selective rule loading algorithm looks at the rule set being loaded into the reasoner, and selects only those rules that are absolutely necessary for reasoning in the domain of application [Tai et al., 2009]. A reasoning algorithm tunes the performance of the reasoning process by reordering the rules so that the rules most specific to the domain of application are executed. RESP, a Reasoner Selection Process [Tai et al., 2011b] is a process that can be used to automatically select a reasoner that best suits the characteristics of a semantic application. Application characteristics include the frequency of knowledge base changes, the complexity of queries required, and whether a closed world assumption may be made. The approach of reducing the rule set is interesting in the telecommunications domain because, for example, where information from network elements is being monitored, cardinality checking rules on attributes can be dropped because network elements should never send attribute values that are out of range. In the telecommunication management domain, a closed world assumption may be made in many applications, but the amount of knowledge base updates will be large. Selection of an appropriate reasoner based on the required characteristics of an application is an interesting facet of this research that could be applied in the telecommunication management domain.

2.3.4 Semantics in Autonomic Network Management

Autonomic Network Management [Mortier & Kiciman, 2006] [Agoulmine et al., 2006], an autonomic computing [Kephart & Chess, 2003] approach to management, provides a framework within which management techniques such as policy based management systems [Strassner, 2004], peer to peer approaches [Mischke et al., 2003] and semantic approaches [Ye et al., 2007] can be integrated to build automated management systems. This section examines related work that has applied semantic approaches in autonomic management.

The Autonomic Service Management Framework (ASMF) described by Cheng et al. (2008) is an autonomic management framework for composition and management of web services. Available web services are described semantically as are SLAs and SLSs in service requests. ASMF uses semantics to automatically compose communication applications from common service components that are managed autonomically.

Lopez de Vergara et al. (2009) describe a case study they carried out on the use of ontologies by a telecommunications operator to autonomously deploy services in home networks. They found that ontologies provided many advantages. They

provided a dynamic model of the services and device configuration concepts and relationships to be shared between operators and home devices. SWRL rules were very useful for expressing the desired behaviour of the system and inference was a useful mechanism for getting a complete view of distributed information.

Garcia Clemente et al. (2011) define a System Description language (SDL), an OWL ontology and a Security Policy Language (SPL) (a usage of SWRL), to define policies. A Border Gateway Protocol deployment on routers is modelled in SDL, and rules are written in SPL. The system manages routing policies for the network using reasoning.

Jennings et al. (2007) describe a conceptual control loop for autonomic network management. An Information Processing component marshals information, which is passed to a Knowledge Generation and Analysis component. That component produces business policies, which are deployed by a Policy Processing component. The FOCALE autonomic network management architecture is a realization of the MAPE control loop. FOCALE has a knowledge base that uses DEN-ng at its core, which is augmented to with semantics. This allows models to be updated with analysed information gathered from the network, ensuring they *"accurately reflect the current operational status"*. The FOCALE architecture specifies a fully-fledged policy driven autonomic management system that uses semantics for interoperability. Implementing even a part of its architecture is a daunting task because there are no reference implementations of the architecture or of DEN-ng available.

There has been considerable work on using semantic techniques to enhance DEN-ng and FOCALE. In the PRIMO architecture described by Davy et al. (2007), ontologies manage policy execution context, transforming the content and representation of DEN-ng policies in response to context changes. Strassner et al. (2009a) describes how the DEN-ng model was extended so that ontologies can be generated from it to govern behaviour. The DEN-ng model can reference those ontologies because *"facts in the information model serve as indices into concepts in the ontologies"*. Reasoning can be applied to those ontologies so that, as the context of a service changes, policies can be applied to adapt the service delivery network. Strassner et al. (2009b) describe a context aware policy model that uses policies and ontologies to map business concerns to DEN-ng entities. OSLAM [Seo et al., 2011] is an architecture that uses SWRL rules on ontologies tied to a DEN-ng information model to detect SLA violations. This combined effort brings considerable semantic support to DEN-ng and

FOCALE. However, because DEN-ng does not support end user service delivery concepts, it is difficult to bring the fruits of this work to bear in that domain.

[Zrelli et al., 2012] proposed a service oriented architecture *"for deploying ontology driven network management services on large scale heterogeneous infrastructures"* based on the FOCALE architecture and have developed a proof of concept implementation that monitors and diagnoses quality of service issues on data paths between two routers. The system has a knowledge base at its core and uses data interfaces to interwork with network infrastructure. Applications communicate with distributed domain managers to build cross-domain applications. The application of a semantic autonomous approach and the distributed aspect of this ongoing work are interesting. However, the evaluation of the proof of concept implementation was executed with a single domain manager and no analysis of the performance of the system was given.

2.4 Summary

As opposed to connectivity services offered by network operators, end user services are those services executed by end users. The end user services that generate most Internet traffic are video streaming, web browsing, and file transfer [Sandvine, 2012]. Network context can impact the experience of different services as perceived by end users in different ways. Video streaming service quality is heavily impacted by packet loss [Bae et al., 2009] whereas throughput and latency impairments impact web page load times [Sundaresan et al., 2011].

DSL access networks and mobile broadband networks are the most common types of access networks to which end user terminals are connected [OECD, 2011a]. In DSL access networks, impairments to throughput and latency commonly occur at peak times [Sundaresan et al., 2011]. In mobile broadband access networks, latency is a known issue [Blajic et al., 2007] and the location of the terminal and the limited total capacity of a cell affect throughput [Johansson et al., 2009]. These impairments impact the experience of end user sessions during use.

Optimization of end user services has been raised as a requirement in the TM Forum "Managing the Quality of Customer Experience" technical report [TMF, 2009] and optimization features heavily in SON [Aliu et al., 2013] use cases from the NGMN [NGMN, 2007a]. To plan and execute optimize end user services, *service expectations*

must be set, *service experience* [Stankiewicz et al., 2011] and *service delivery context* [Toutain et al., 2011] must be monitored and analysed.

Monitoring of network metrics on physical and logical network resources is widely deployed in centrally managed networks [IETF, 1991] [3GPP, 2012e]. Service metrics are increasingly being reported from terminals [IETF, 2003a][3GPP, 2010b], and there have been some efforts to assess service quality on terminals [Sidibe & Mehaoua, 2008][Soldani, 2006]. Apps on smart terminals [Vodafone, 2013] and even analysis of social media communications [Qiu et al., 2010] can be used to determine end user service experience.

Analysis of end user service experience and context determines compliance with expectations. There are standards that specify parameters for qualitative assessment of service experience for some services [ITU-T, 2008b][ITU-T, 2005a]; those parameters can be used to specify expectations on service experience. Using network metrics to analyse service quality is challenging [Stankiewicz et al., 2011][Bhushan et al., 2004] because there is no fixed mapping from network resources to services. Weighting, grouping, and statistical techniques [Batteram et al., 2010][Kim et al., 2009] must be tuned to a particular service context; reconstructing service sessions [Choi et al., 2010] is time consuming. Existing systems [IBM, 2012b][Ericsson, 2012] recommend service improvements or use mathematical models [Sterle et al., 2011] to suggest and deploy optimizations. Research efforts that use machine learning [Simoens et al., 2008] and semantics [Song et al., 2012] to analyse end user service experience is producing encouraging results but is as yet immature.

Optimization of end user service delivery is an immature field in practice and in research [Sterle et al., 2011]. Approaches have been proposed for optimization of centrally controlled end user services using central session control mechanisms [Sterle et al., 2011], but such approaches do not work when a central authority is not present. Other approaches such as HTTP adaptive streaming optimize single end user sessions at the expense of all other sessions [Oyman & Singh, 2012]. [Latre et al., 2009] suggest a *learning controller* that can optimize across end user service sessions. [Bouten et al., 2012b] show that policies applied across sets of end user sessions can improve the overall compliance of the running service session set with expectations. Policy based [Sventek et al., 2011] and semantic [Rana et al., 2011] approaches have been described for traffic control of streams in home area networks, but these approaches do not take end user service experience into account. Optimization in

SON for mobile broadband access networks as surveyed by [Aliu et al., 2013] also focuses on network traffic optimization as opposed to end user service optimization.

While there are many models for telecommunication networks [IETF, 2002][IETF, 2006b][3GPP, 2009c], services [TMF, 2011][Leijon et al., 2008] and service expectations [TMF, 2011][Lamanna et al., 2003], few are semantically enabled [Latre et al., 2008][Serrano et al., 2009]. Semantic representation of web services is more mature [Roman et al., 2005][Stollberg & Norton, 2007], and modelling service context semantically is seen as promising [Toutain et al., 2011]. Some common vocabularies and ontologies such as FOAF [FOAF, 2010] and WGS84 [W3C, 2006b] are applicable to represent communication service context. Semantically representing telecommunication knowledge allows true knowledge sharing but the initial modelling costs are substantial [Strassner et al., 2007][Lopez de Vergara et al., 2009].

Reasoning, querying, and applying rules may be used to analyse semantic service knowledge. In addition, semantic mapping [O'Sullivan et al., 2007] and lifting [W3C, 2007] are promising approaches for translating and using existing telecommunication models and information that is not semantically enabled. Although some work has been done in handling temporal context semantically [O'Connor & Das, 2011], handling knowledge in a time sensitive manner is a relatively immature area of study.

Performance issues can arise when using semantic techniques [Lopez de Vergara et al., 2009], an issue underlined by research results [Latre et al., 2008][Keeney et al., 2011]. Tai et al. (2011a) have shown that with appropriate reasoner selection and rule conditioning, substantial improvements can be achieved.

Some work has been undertaken that applies semantic techniques in autonomic management systems. Ontologies are very useful for ensuring interoperability of models and SWRL rules allow flexible definition of behaviour [Lopez de Vergara et al., 2009]. There has been considerable work in augmenting DEN-ng [Strassner, 2002] and FOCALE [Jennings et al., 2007] with semantics. [Zrelli et al., 2012] have built a proof of concept implementation of the FOCALE architecture. All of this work shows promise but, so far, reference systems and deployments have been very limited.

Chapter 3
The EUSAO Ontology

3.1 Introduction

The discussion in Section 2.2 points out that in order to optimize end user service delivery in a networking domain in *near real time*, the *service expectations*, *service experience*, and *service context* must be known. It is crucial, therefore, that the information in any system that is managing service delivery is modelled in a flexible and efficient manner.

The analysis in Section 2.3.2 argues that no model exists, semantic or otherwise, that models the domain of end user service delivery. However, that analysis shows that there is a body of modelling work in related domains that can be used to aid modelling of service expectations and context. The analysis suggests that ontologies are a very flexible way of expressing the structure and interrelations in a domain.

The semantic techniques described in Section 2.3.3 represent a powerful toolkit that can be brought to bear upon an application that is to be developed using a semantic approach. However, in order to apply those techniques in a deployable system, the performance issues raised in using such techniques (as highlighted in Section 2.3.3) must be addressed in the design of both the model and the application.

This chapter describes the EUSAO ontology, a general-purpose model for end user service management designed for application in any networking domain in a transparent and efficient manner. The requirements on the EUSAO ontology are described and justified in Section 3.2. Section 3.3 describes the design of the ontology, laying out its structure and main concepts. In Section 3.4, implementation details of the ontology are described. Section 3.5 describes how the ontology could be used by an application. Section 3.6 provides a summary of the information in the chapter.

3.2 Requirements

This section describes and justifies the requirements on the EUSAO ontology.

1. **Model the domain of end user service delivery.** The EUSAO ontology must capture the main concepts of the problem domain and the relationships between them; namely service expectations, service experience, and service context. This requirement is specified because, as established in Section 2.3.2, there is no generally available model, semantic or otherwise, for the end user service management domain.
2. **Structure the model to enable efficient deployment.** The ontology must have a structure that, when deployed, allows the model to be updated and queried in a resource efficient manner. This allows the ontology to be deployed in a system that runs in near real time. This requirement is specified because the use of semantic techniques is known to have performance issues, as highlighted in Section 2.3.3.
3. **Support any end user service and networking domain.** The ontology must be structured so that characteristics of specific service types or networking domains do not appear in the basic structure of the ontology. This allows the ontology to be applied to any end user service or networking domain. The discussion in Section 2.3.2 reveals many fragmented models and ontologies that address aspects of end user service quality management in specific domains for specific services. This requirement is specified to ensure that the EUSAO ontology models services and networking domains using the same principles that the SID [TMF, 2011] uses to model networking domains and connectivity services. The SID models a domain using generic concepts and specializes those concepts to model specific networks and connectivity services.
4. **Reference existing models where possible.** The ontology must reference and use concepts from other models, vocabularies, and ontologies such as those described in Section 2.3.2 where possible. This requirement is specified to allow existing modelling work to be re-used and to enable semantic interoperability of any system using the EUSAO ontology with systems such as those described in Section 2.3.1 that use the referenced models.

3.3 Design

Any application that is managing end user service delivery must be designed to handle the frequent measurements that monitoring of end user sessions generates over time. The EUSAO ontology is structured using an approach similar to those used

by O'Connor & Das (2011) and Keeney et al. (2010) explained in 2.3.3, in that explicit support is added to the ontology to handle the temporal aspects of concepts. Each concept to be held in the ontology has two boolean properties, *dynamicity* and *temporality,* as shown in Table 3-1. The dynamicity property of a concept is used to indicate a concept is changed by execution of a system using the ontology. Concepts whose individuals change only as a result of configuration changes are categorized as *static concepts*; concepts whose individuals are changed by execution of a system using the ontology are categorized as *dynamic concepts*. The temporality concept is used to indicate that individuals of a concept are created periodically, for example as a result of regular metrics being collected. Concepts whose individuals are not created periodically are classified as *global concepts* and concepts whose individuals are created periodically are classified as *snapshot concepts*.

Table 3-1: The Dynamicity and Temporality Properties of Concepts in the EUSAO ontology

Category	Description	Value	Meaning
Dynamicity	Indicates if individuals of a concept are changed by execution of the system or not	Static	Individuals of a concept are changed by configuration only
		Dynamic	Individuals are changed during execution of the system
Temporality	Indicates if individuals of a concept have a time dimension	Global	Individuals are not created periodically
		Snapshot	Individuals are created periodically

Examples of each possible combination of dynamicity and temporality appear in Table 3-2. Of course, the content of the static concepts does change, but those changes are controlled and happen so infrequently as to be considered static in a near real time system using the ontology. It is assumed that user expectations are unlikely to be entered into the system more often than daily, whereas web page download times may be reported a number of times per minute. Static concepts are always considered to be global.

The EUSAO ontology models concepts with these categories so that a system using the ontology can optimize handling of individuals of concepts based on their categories. A system using the ontology must carry out maintenance on dynamic concepts but maintenance is not required on static concepts, and it may handle snapshot concepts differently to global concepts because there will be many more instances of snapshot concepts than global concepts.

Table 3-2: Concept Categories as used by Concepts in the EUSAO Ontology

Dynamicity	Temporality	Examples
Static	Global	Expectations set by an end user on the page load time of web pages
Dynamic	Global	The network connection used by a user session to browse a web page
Dynamic	Snapshot	The web page download times reported every 10 seconds on a user session browsing a web page

The core concepts and relationships in the EUSAO ontology are shown in Figure 3-1, the full ontology appears in Figure B-1 of Appendix B. Concepts depicted with rounded rectangles are static whereas concepts depicted with ovals are dynamic. The design of the end user service management concepts in the ontology was influenced by the author's previous experience in designing a server and relational database for Ericsson's User Service Performance (USP) Framework [Holm-Oste & Norling, 2008].

The object relationships between individuals are specified as being *static* or *direct*. Static relationships are relationships between individuals that do not change during the lifetime of the system such as the priority a user has when running a service. Direct relationships such as the relationship between a service session and a user are relationships that are asserted during execution of the system.

The ontology references concepts in a number of other ontologies, vocabularies, and models, the use of which provides a set of concepts and relationships that have already been modelled. The colour of concepts indicates the ontology to which they belong. Table 3-3 lists referenced ontologies, vocabularies, and models.

The concepts referenced from the SID and from the Time ontology are central to the EUSAO ontology because they supply the temporal and network domain concepts of the ontology. Using other models to replace the SID or the Time ontology would change the design and implementation of the EUSAO ontology and any system using it considerably. The other ontologies are merely referenced by the EUSAO ontology; replacing them with another model in the design of the EUSAO ontology would be straightforward. However, care must be taken by any implementing system to avoid creating dependencies to any such model. An approach such as the use of the Strategy design pattern [Gamma et al., 1994] can be employed to encapsulate the use of any such ontology behind an interface.

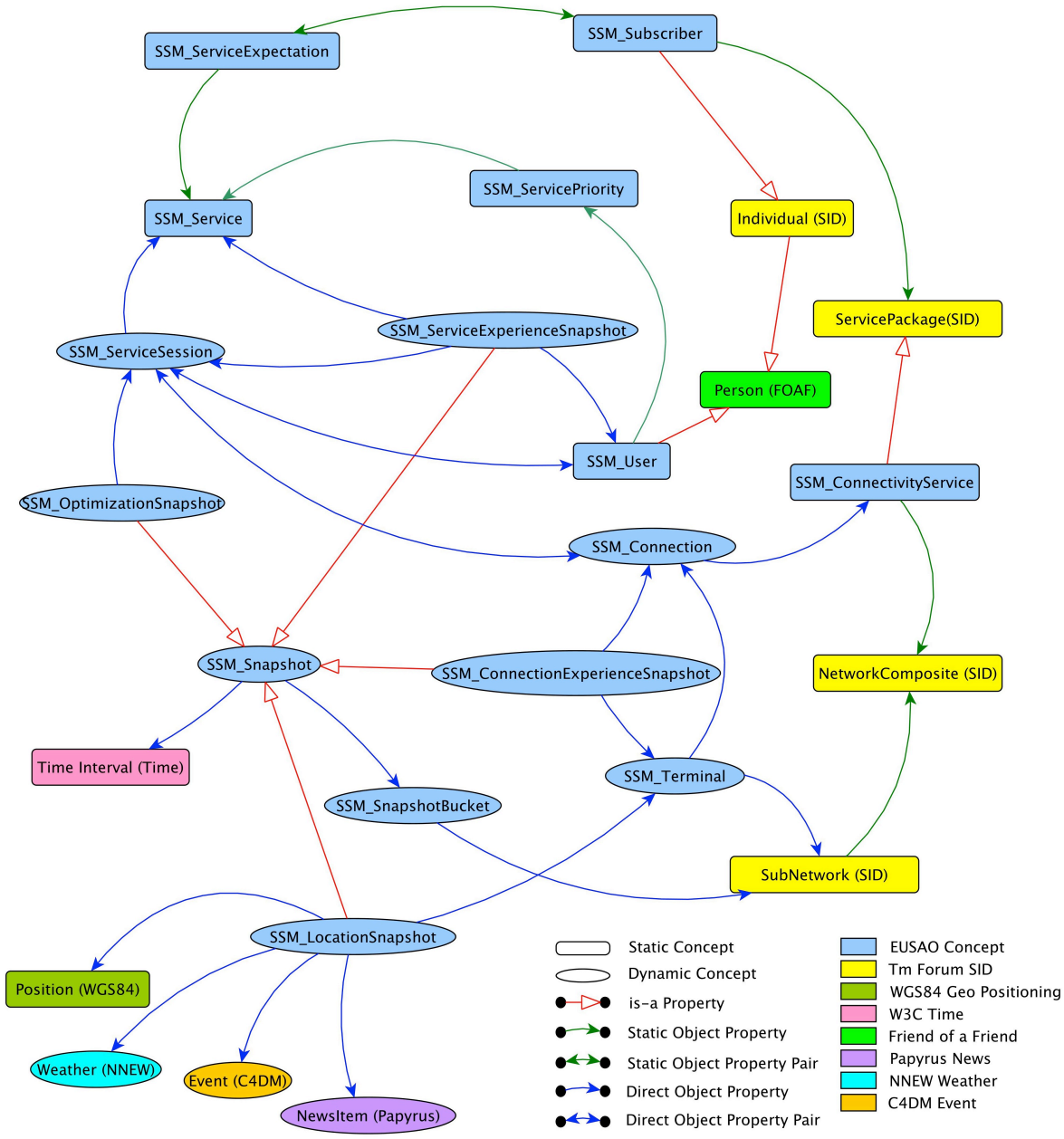


Figure 3-1: Core Concepts in the EUSAO Ontology

In order to better illustrate the structure of the EUSAO ontology, the core global concepts of the ontology are shown in Figure 3-2. Figure 3-3 shows the snapshot concepts and core global concepts they reference.

In the EUSAO ontology, service expectations are modelled as shown in Figure 3-2. The *SSM_Subscriber* concept models the subscriber who owns the subscription which end user services are using, and who sets expectations. The subscriber sets service expectations for each service type being managed, modelled as *SSM_ServiceExpectation* and *SSM_Service* concepts respectively. Examples of service types are video on demand or web browsing. Users of services are modelled using the *SSM_User* concept. Each user is assigned a priority on each service they are

authorized to use, modelled as an *SSM_ServicePriority* concept. Modelling service expectations in this way means that, during service execution, each service type running under the subscription of a subscriber is assessed against the service expectations for that service type. The priority of a session of a service type depends on the user running that session. This means that, even though two sessions being run by different users under the same subscription may have the same level of compliance with expectations, the different priorities associated with those two sessions can be used to treat those sessions differently.

Table 3-3: Ontologies, Vocabularies, and Models Used and Referenced

Ontology	Description
Information Framework (SID)	The TM Forum's information Framework (SID) UML model [TMF, 2011]; selected because it is the recommended model of the TM Forum, is widely used, and has a service model in common with DEN-ng [Strassner, 2002].
WGS84 Geo Positioning	A RDF vocabulary from the W3C that provides a namespace for entering location information such as latitude and longitude [W3C, 2006b]; selected because it is the standard vocabulary for positioning from the W3C.
Time	An ontology from the W3C that captures time related concepts as instants and durations [W3C, 2006a]; selected because it is the standard ontology for time from the W3C.
Friend of a Friend	An ontology that describes people and their interactions [FOAF, 2010]; selected because of its wide use and popularity.
Papyrus News	The Papyrus News ontology captures news items and the relationships between news items and other concepts such as persons, organizations, and [Papyrus, 2011]; selected as an example news ontology because it was developed by a EU project and the ontology and its documentation are available online.
NNEW Weather	An ontology that describes current and forecasted weather concepts such as wind, precipitation, pressure, and location [NNEW, 2011]; selected as an example weather ontology because it is supported by US Government agencies such as the Federal Aviation Authority.
C4DM Event	An ontology that captures the concept of an event and its relationship with concepts such as location and time [C4DM, 2011]; selected as an example event ontology because it is the event ontology used by organizations such as the BBC [BBC, 2013].

The network connectivity context from a terminal to the network is modelled using the following core concepts, as shown in Figure 3-2. The *SSM_Terminal* concept represents the physical terminal that the user is using. It is connected to a *SubNetwork*, a part of a *Network*. A *SubNetwork* might be a Home Area Network or a

cell in a mobile broadband access network. An *SSM_Connection* is opened from a terminal using an *SSM_ConnectivityService* that is provided by a network operator as a *ServicePackage*. An *SSM_Connection* might be IP connectivity being provided using an interface on the *SSM_Terminal*.

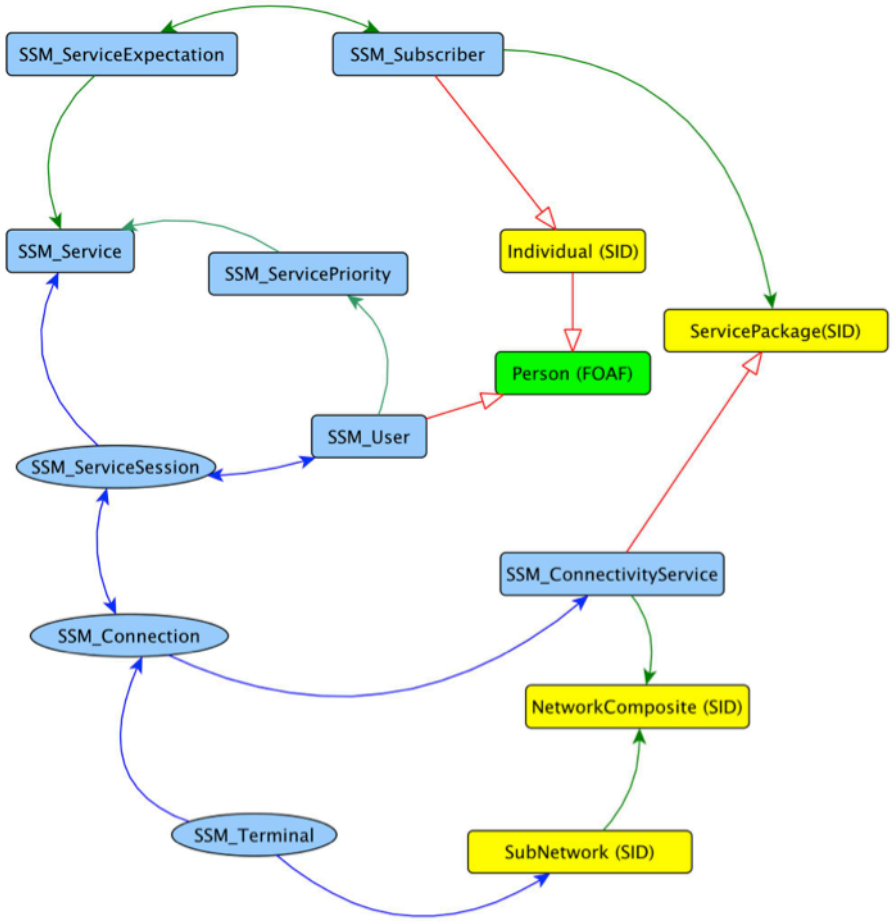


Figure 3-2: Core Global Concepts in the EUSAO Ontology

The *SSM_ServiceSession* concept, as shown in Figure 3-2, is the core concept that models an end user service session. A session has a user and runs a service over a connection. This central concept links concepts representing service experience, service context, and service expectations together.

It is important to note the relationship in the model between the EUSAO *SSM_Service* concept and the SID *ServicePackage* concept. An *SSM_Service* models an end user service such as web browsing. An *SSM_ServiceSession* individual models an end user service session such as a web browsing session. A SID *ServicePackage* models a *ServicePackage* made up of SID *CustomerFacingService* individuals, each of which provides connectivity to a subscriber. An *SSM_ServiceSession* of an end

user *SSM_Service* may be carried over a SID *CustomerFacingService* connectivity service.

The SID *NetworkComposite* and *SubNetwork* concepts provide a link into the SID resource domain, the *ServicePackage* concept provides a link into the SID service domain, and the *Individual* concept provides a link to SID Common Business Entities. The *SSM_Subscriber* and *SSM_User* are modelled as a FOAF Person, which may be used to reference other user and service information, thus providing very fertile sources of knowledge for analysis.

The EUSAO ontology uses the concept of a snapshot to model knowledge that has a time dimension, as shown in Figure 3-3. Examples of such knowledge are the web page download time on a web browsing session, the packet loss on a connection, or the location of a smart phone. Because such knowledge changes over time, it is convenient to segment time into snapshots and use a single individual to model the knowledge during that time interval, as shown in Table 3-4. Using this approach, a sample value or an average of the value of each concept is captured for each snapshot. If the time interval represented by each snapshot is short enough for the domain of application in question, this approach can give a good representation of the variability of a concept over time.

Table 3-4: Snapshots Recording Knowledge with a Time Dimension

Concept	Snapshot 1	Snapshot 2	Snapshot t
Web Page Download Time	2.3 Seconds	2.4 Seconds	1.9 Seconds
Connection Packet Loss	0.32%	0.25%	0.27%
Smart Phone Location	N 53° 25' 47.2352" W 7° 54' 7.2084"	N 53° 25' 36.7412" W 7° 54' 6.7449"	N 53° 25' 42.2644" W 7° 54' 54.3295"

The snapshot concepts in the EUSAO ontology and their references to core global concepts are shown in Figure 3-3. The central *SSM_Snapshot* concept models a snapshot time interval, and all concepts that are measured for a time interval are sub-concepts of that concept. The time interval of a snapshot is represented as a time interval in the W3C Time ontology, allowing concepts in other ontologies with a time dimension to be linked to the snapshot.

There are four snapshot concepts in the EUSAO ontology.

1. The *SSM_ServiceExperienceSnapshot* concept models the metrics collected on an end user service session during a snapshot. Metrics such as video playout freeze

counts, mean opinion score estimates, and web page load times are modelled. This concept represents the service experience of a session during a snapshot.

2. The *SSM_ConnectionExperienceSnapshot* concept models the metrics collected on the connection of a terminal during a snapshot. Metrics such as packet loss and latency are modelled. This concept represents the network context on a connection for a session during the snapshot in question.
3. The *SSM_OptimizationSnapshot* concept models optimization actions executed on a session during a snapshot. The type of optimization, such as a throttling action, that was executed is modelled. This concept represents the optimization context of a session during the snapshot in question.
4. The *SSM_LocationSnapshot* models the location of a terminal during a snapshot; the location is modelled using the WGS84 Geo Positioning vocabulary. This concept represents the spatial context of a session during the snapshot in question.

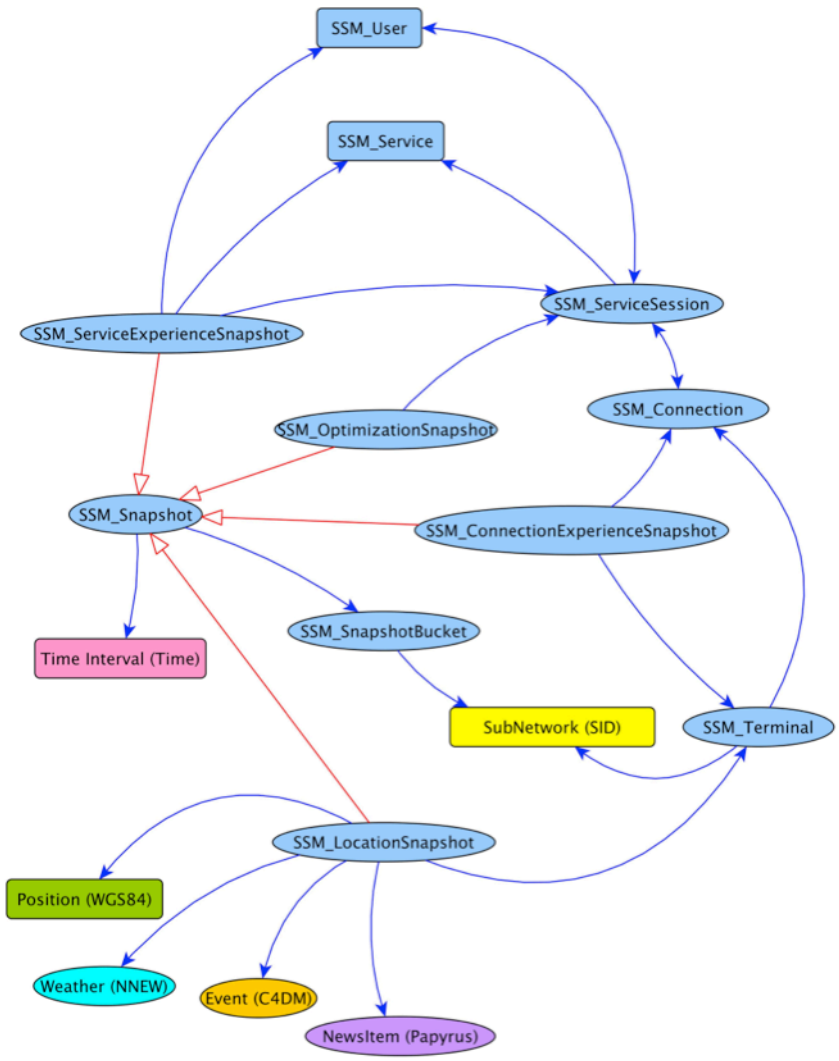


Figure 3-3: Snapshot Concepts and Referenced Core Global Concepts in the EUSAO Ontology

Location is important in mobile networks. Terminal locations can be used to reference sporting, music, or cultural events in the C4DM event ontology, news events in the Papyrus News ontology, or the current meteorological conditions in the NNEW Weather ontology. Location can be used to analyse service experience in the context of knowledge drawn from any of these sources.

A snapshot bucket is represented as an *SSM_SnapshotBucket* concept. A snapshot bucket is used to group snapshot individuals from different concepts together for common processing. Snapshot individuals for all concepts from a particular *SubNetwork* reference a single *SSM_SnapshotBucket*. An application using the EUSAO ontology might employ snapshot buckets as a partitioning concept. For example, in the design of the Aesop knowledge base a separate model is used for each snapshot bucket, as explained in Section 4.3.2.

Figure 3-4 shows how sub-concepts are used to represent knowledge for particular service types. The *SSM_ServiceExpectation*, *SSM_ServiceSession*, and *SSM_ServiceExperienceSnapshot* concepts have sub-concepts that model service expectations, sessions, and service experience for each service type modelled in the ontology. In the case of the *SSM_ServiceSession* concept, it has *SSM_<ServX>Service Session* sub-concepts, one for each service being managed, namely *SSM_WebBrowsing ServiceSession*, *SSM_FileTransferServiceSession*, and *SSM_VideoOnDemand ServiceSession*.

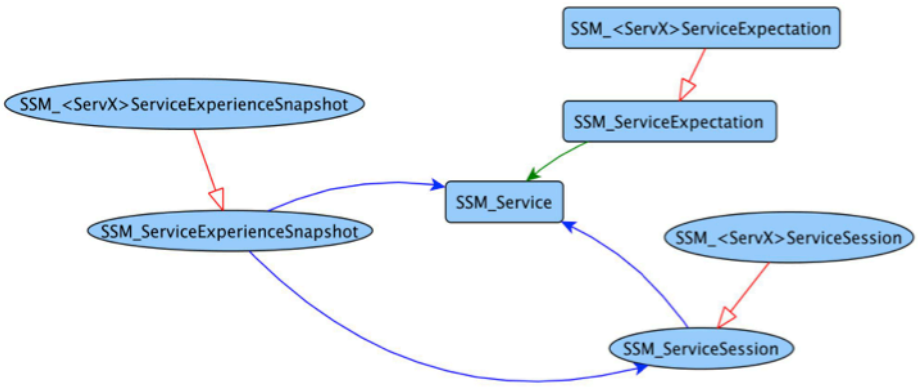


Figure 3-4: Sub-Concepts Representing Service Type Concepts

In each case, the base concept models knowledge that is common for that concept for all service types, and the sub-concepts model service type specific knowledge. For example, the base *SSM_ServiceExperienceSnapshot* concept models the number of bytes sent downstream and received upstream on an end user session for a snapshot as a data property because bytes are sent downstream and received upstream on all

end user sessions regardless of their type. However, the *SSM_VideoOnDemandService ExperienceSnapshot* concept models the number of video freezes in a snapshot because this property is specific to the video on demand service.

The EUSAO ontology is designed so that it can be held in memory, individuals are assumed to be volatile. If the boolean *ssm_persistence* data property is set on a concept in the ontology, individuals of this concept should be saved to persistent storage by any system using the EUSAO ontology

3.4 Implementation

The EUSAO ontology was implemented in OWL 2 [W3C, 2009]. A diagram of the full ontology appears in Figure B-1 of Appendix B, and the ontology is available in RDF/XML format [W3C, 2004b] on the optical disk accompanying this thesis.

OWL was chosen as a representation language for the EUSAO ontology because it has the expressive power to capture the complex relationships in the end user service management domain. The OWL 2 [W3C, 2009] flavour of OWL was chosen because it is the latest version of OWL recommended by the W3C. The EUSAO ontology was implemented using the Protégé ontology editor [Stanford University, 2012]. The Pellet reasoner [Clark & Parsia, 2013] was used as a reasoner with Protégé to check the structure and validity of the EUSAO ontology. The Pellet reasoner was chosen because it can be used with Protégé during ontology implementation and at execution time as a reasoner in a run time system.

All concepts were modelled as classes, all references are explicitly specified as object properties, and all attributes are explicitly specified as data properties in Protégé. All concepts, references, and attributes are documented with comment annotations. The cardinality is specified on all references and attributes of concepts. The domains, ranges, and characteristics of references and attributes are also specified. Screenshots of the Protégé ontology editor showing the concepts, references, and attributes of the EUSAO ontology appear in Figure B-2, Figure B-3, and Figure B-4 of Appendix B.

The screenshot of Protégé in Figure 3-5 shows the *SSM_Terminal* concept specification as an illustration of the level of detail specified on concepts in the ontology. The various references and attributes and their restrictions and cardinalities are shown.

Figure 3-6 shows a screenshot of the *ssm_communicatesUsing* reference that has the *SSM_Terminal* concept as its domain and the *SSM_Connection* concept as its range. The reference provides the link between a terminal and a connection the terminal is using to communicate. The characteristics specified on the reference are also shown.

The *ssm_terminalModel* attribute of the *SSM_Terminal* concept is shown in the screenshot in Figure 3-7. This attribute has a range of *string*, and is used to hold the model of a terminal being used by a user to communicate. The *SSM_Terminal* is modelled to this level of detail because end user service experience problems such as hardware or software faults may occur with particular types of terminals and may be quite easy to infer.

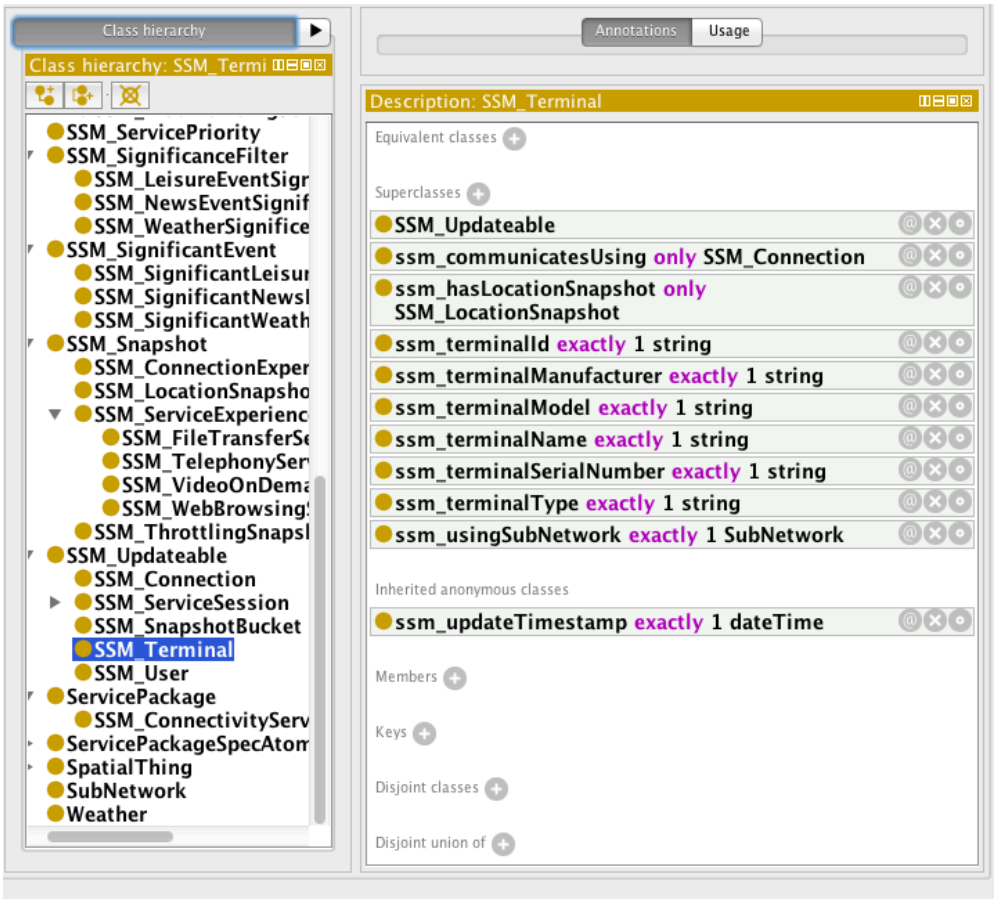


Figure 3-5: The *SSM_Terminal* Concept Specification in Protégé

During implementation of the ontology, the external ontologies and vocabularies referenced by the EUSAO ontology (see Table 3-3) were downloaded from their respective web sites and imported into Protégé. The TM Forum's information Framework (SID) UML model [TMF, 2011] was converted into an OWL ontology using tools developed by Theisz et al. (2011) and Quinn & Matusikowa (2010) and imported. The EUSAO ontology was periodically checked for internal consistency and

consistency with the imported ontologies using the Pellet reasoner [Clark & Parsia, 2013]. These external ontologies and vocabularies are available on the optical disk accompanying this thesis.

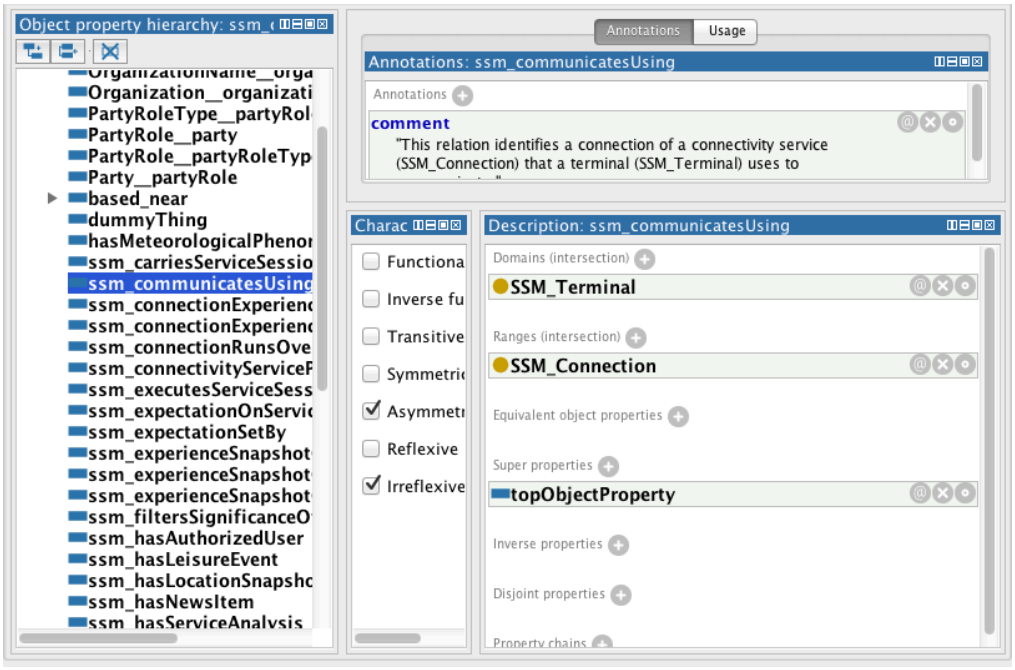


Figure 3-6: The *ssm_communicatesUsing* Reference in Protégé

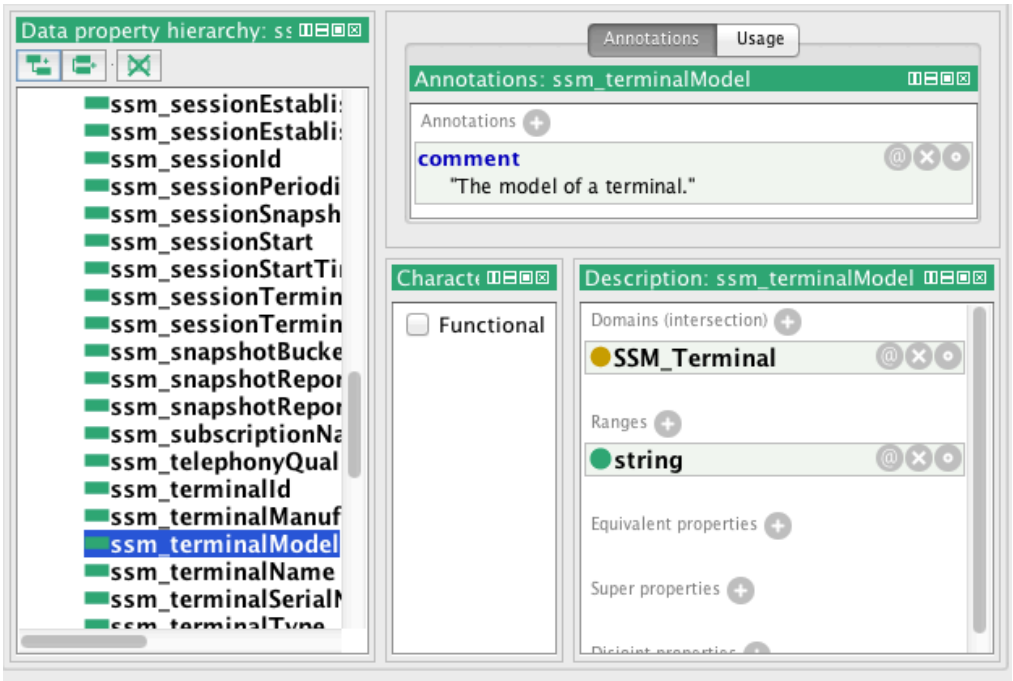


Figure 3-7: The *ssm_terminalModel* Attribute in Protégé

3.5 Using the Ontology

A system using the EUSAO ontology creates individuals of static concepts such as service expectations at system start using mechanisms such as reading them from a semantic store, file, or external system.

During operation, users start and run service sessions, each of which is modelled as an individual of a session. Each session is being run at a particular location and delivers a certain service experience, which may vary over the session lifetime. Service experience and context knowledge is monitored by a system using the EUSAO ontology as sessions run, and is used to populate individuals and object relationships of dynamic global and snapshot concepts. Service experience and context individuals are populated using mechanisms such as RDF [W3C, 2004a] streams carrying metrics from terminals or networks, or information from weather or news services.

The EUSAO ontology can be used by a system to analyse and to plan and execute optimizations to end user service delivery. The system can use querying, reasoning and inference to determine how changes in network resource state, changes in the location of service delivery, changes in weather at the service delivery location, and the occurrence of current events impact end user service experience over time.

The relationship between *SSM_ServiceExperienceSnapshot* and *SSM_Service Expectation* is interesting because it allows the service experience of a session to be compared to the service expectations on that service giving the level of compliance of an end user service session with expectations for a snapshot. Another important relationship is the priority of a service session, which can be inferred by finding the service type of a session and determining the priority of that service type for the user running the session. Once the compliance and the priority of each session is known for all running sessions, optimizations on that running session set can be planned and executed by a system using the EUSAO ontology.

When the connectivity experience of connectivity services is collected as *SSM_ ConnectionExperienceSnapshot* individuals, the performance of each connection modelled as an *SSM_Connection* over time is known. All connections use a connectivity service modelled as *SSM_ConnectivityService* concepts, sub-concepts of the SID *ServicePackage* concept. The SID *ServicePackage* concept has relationships to the SID SLA concepts such as the *ServiceSpecification* concept. Therefore, one can reason from the service experience to the connectivity experience, and onto the SLA specified in SID on a connectivity service package. This means that the relationship between problems on connectivity services and end user services be inferred. The relationship between problems on connectivity services and compliance with the SLAs of those connectivity services can also be inferred.

The EUSAO ontology can be specialized to model end user services running in a particular networking domain. Individuals of the SID concepts of *NetworkComposite* and *SubNetwork* (see Figure 3-2) would model access networks and sub networks. Connectivity services offered on the access network are modelled as a *NetworkComposite*. In addition, Dynamic Global Knowledge concepts of *SSM_Terminal*, *SSM_Connection*, *SSM_Session*, and *SSM_User* and Snapshot Knowledge concepts of *SSM_ConnectionExperienceSnapshot*, *SSM_Location Snapshot*, and *SSM_ServiceExperienceSnapshot* would be specialized to hold object and data properties representing knowledge specific to the networking domain in question.

3.6 Summary

The EUSAO ontology can model the concepts, relationships, constraints, and complexity of end user service management in a domain independent way, and can be used by a management system to manage the knowledge of that domain, thus meeting the first requirement specified in Section 3.2.

The ontology categorizes concepts by their *dynamicity* and *temporality*, and uses *snapshots* to partition knowledge with a time dimension, allowing a system using the ontology to optimize handling of the knowledge in each of these categories. This meets the second requirement in Section 3.2.

The ontology uses sub-concepts to manage particular end user service types, and models connections to networks generically, thus allowing it to be applied to any end user service type or networking domain. These capabilities meet the third requirement in Section 3.2.

The ontology references other ontologies heavily, using their concepts as building blocks where possible. The SID model is used particularly heavily, and is used to represent concepts in networks over which end user services are running. This capability meets the fourth requirement in Section 3.2.

An end user service delivery management system using the EUSAO ontology can employ the constructs of the ontology to manage temporal knowledge in an efficient manner. Such a system is Aesop, which is described in Chapter 4.

Chapter 4
Aesop

4.1 Introduction

In Section 2.2, the discussion highlights the need for management of end user service delivery identified by the TM Forum [TMF, 2009]. The NGMN have highlighted the need for optimization to be autonomic and execute in near real time [NGMN, 2007a].

The survey of end user service management in Section 2.3.1 shows that no system exists that can manage end user service delivery of all service sessions in a networking domain. Approaches have been identified and applied for setting expectations on end user services and for monitoring the service experience and context of end user service sessions. However, approaches for analysing sessions to determine compliance with expectations either depend on static mappings from network context or are specific to a particular service type. Current practice is to use planning tools for optimization; optimization approaches in research are either specific to individual service sessions or service types, or rely on using packet signatures to shape IP flows irrespective of the end user service experience on those flows.

This chapter describes the design of the Aesop approach for autonomic end user service management system which has a classic autonomic MAPE loop [Kephart & Chess, 2003] around a knowledge base structured with the EUSAO ontology, allowing Aesop to draw on the semantic techniques described in Section 2.3.3 in its design. Aesop monitors and stores end user session experience and context knowledge into the knowledge base and uses semantic rules and queries to analyse service session knowledge to ensure sessions comply with expectations. Aesop plans optimizations with algorithm that uses semantic reasoning to maximize the overall compliance of all sessions in a managed domain with expectations and executes actions to deploy those optimizations on the network.

The survey in Section 2.3.4 shows that semantically enabled autonomic management architectures and frameworks have been proposed in the literature. However, concrete reference systems and deployment experiences are not described. Aesop, as described in this chapter, is a fully specified, designed, implemented, and executable autonomic management system.

The requirements on Aesop are described and justified in Section 4.2. Section 4.3 describes the design of the Aesop approach. In Section 4.4, the manner in which the Aesop approach was applied to implement a prototype is described. Section 4.5 provides a summary of the chapter.

4.2 Requirements

This section describes and justifies the requirements on Aesop as an approach for managing end user service delivery.

1. **Optimization of end user service delivery.** Aesop must be designed so that it can optimize a set of end user service sessions in a networking domain so that compliance of that set of end user service sessions with expectations is always maximized. This requirement is derived from the research question in Section 1.2.
2. **Autonomic optimization in near real time.** Aesop must execute autonomically in near real time. The time taken to monitor service experience and context, analyse the compliance of service experience with expectations, and to plan and execute optimization actions must be a small number of minutes. This ensures that service delivery changes are proactively executed in a short timeframe. This requirement is derived from the research question in Section 1.2.
3. **Use a common model to hold all system knowledge.** A common model that holds all knowledge must be used. This ensures that the knowledge in the system is always consistent, that no function has unmanaged local knowledge, and that all functions in the system access and manipulate the same knowledge. This requirement is specified in order to comply with the principle of using central knowledge in an autonomic system [Kephart & Chess, 2003].
4. **Support new end user services and networking domains without source code changes.** The system must be designed with mechanisms that allow the introduction of new end user services and network domains by changing metadata, mappings, queries, and rules. This means that the development effort

for implementing new services and networking domains is lowered and they can be added to the system at run time without the need for upgrades or restarts. This requirement is specified to ensure Aesop complies with the principles of Model Driven Software Development [Stahl et al., 2006] where domain changes are implemented using Metadata where possible. The discussion in Section 2.3.1 and Section 2.3.2 reveals many fragmented approaches that address aspects of end user service quality management in specific domains for specific services. This requirement is specified to ensure that Aesop takes a model driven approach with the EUSAO model at its core to allow the approach to be easily applied to arbitrary services in multiple networking domains.

5. **Use commodity hardware and readily available semantic frameworks, libraries, and toolsets.** The approach must be designed and implemented using well researched and well tried semantic approaches. This means that the unknown factors in designing the system were limited to application of known semantic approaches and components in the system, rather than on developing new semantic components. Once the system is designed and implemented, the performance of individual semantic components can be addressed, and can be tuned, enhanced, or replaced with a more suitable component. This requirement is specified to set the scope for the use of semantic techniques, the application of which is fundamental to the research question in Section 1.2.

4.3 Design

The conceptual structure of Aesop is shown in Figure 4-1. Aesop consists of a *Knowledge Base* and an *Engine* made up of components designed to provide the autonomic MAPE functions and components designed to provide support functions. The engine components use the *Knowledge Bus* to access the Knowledge Base.

The *Knowledge Base* manages the knowledge of the system using the EUSAO ontology. It holds knowledge on the service expectations, experience, and context of end user service sessions running in the managed domain. To overcome the performance constraints of semantic querying, reasoning, and rule execution, the end user service session knowledge in the Knowledge Base is partitioned in a novel manner. Service session knowledge that has a time dimension is partitioned into snapshots of a certain configurable length. This Knowledge Base partitioning means that semantic queries, reasoning, and rules used by components in the Aesop engine

need only act on relevant snapshots thus reducing the volume of input knowledge and increasing performance.

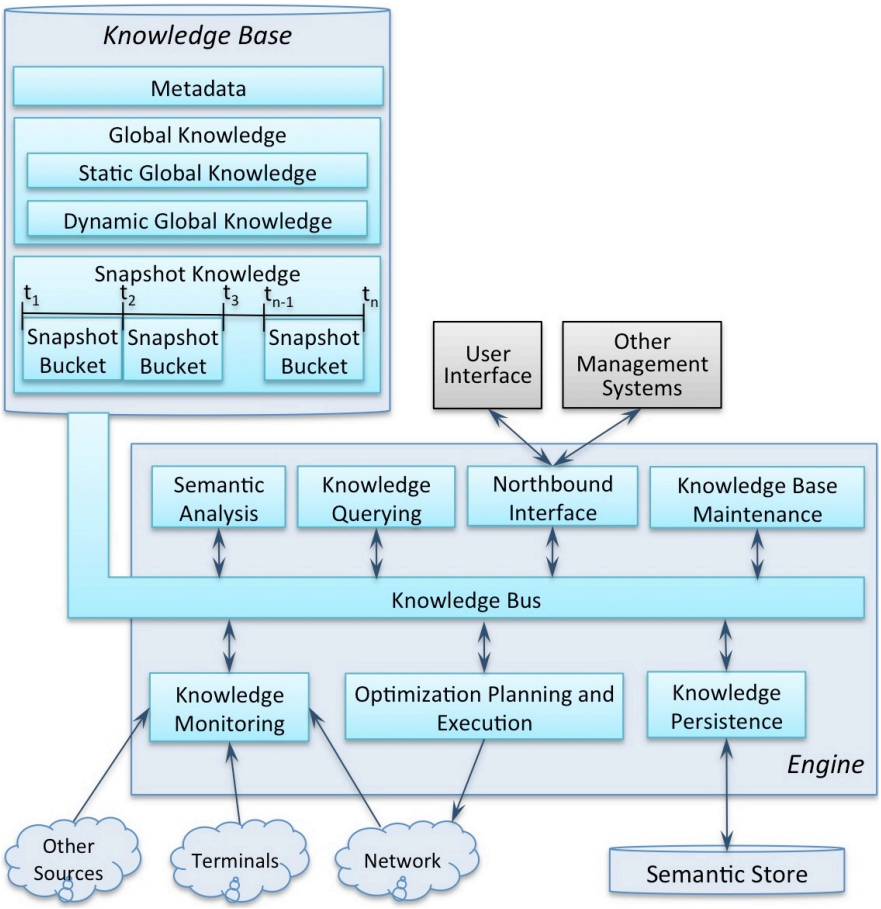


Figure 4-1: The Aesop System

The *Knowledge Bus* is a software interface that allows components to store, access, and update knowledge in the Knowledge Base in a consistent manner. The Knowledge Bus ensures that simultaneous access operations by components to the Knowledge Base do not interfere with each other.

The Aesop *Engine* components are designed to execute independently of each other. All interaction occurs through the Knowledge Base. This design choice ensures that each component always uses consistent and controlled knowledge and that the functions of each component are independent and standalone.

The MAPE loop is realised by executing a series of components that interact indirectly through the Knowledge Base. Execution of *Knowledge Monitoring* is triggered when incoming information is detected; the incoming knowledge is read and stored into the Knowledge Base. *Semantic Analysis* operations execute periodically, reading any recently stored end user service session knowledge from the Knowledge Base, analysing it, and storing the result to the Knowledge Base.

Optimization Planning and Execution operations also execute periodically, reading knowledge on the running end user service set from the Knowledge Base and executing optimization actions based on that knowledge.

The main software components in the design of Aesop and their relationships are shown in Figure 4-2. The design of Aesop is elaborated with reference to in Figure 4-2 in the following sub-sections.

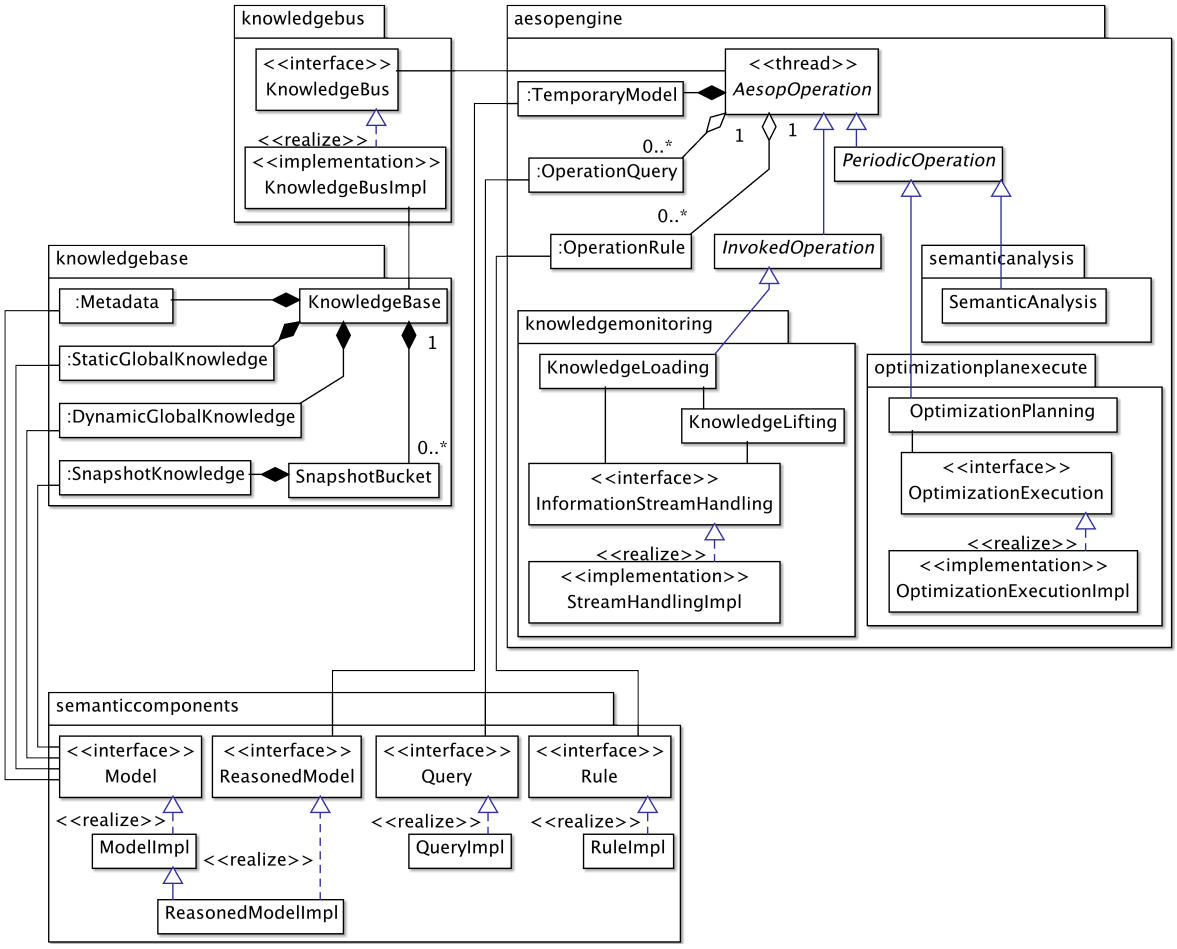


Figure 4-2: UML Class Diagram of Aesop Software Components

4.3.1 Semantic Components

The `semanticcomponents` package shown in Figure 4-2 is used to provide semantic functions and features to the Aesop Knowledge base and Engine. The package provides interfaces to the semantic functions and features that encapsulate the implementation details of those functions and features. Two interfaces are provided to hold semantic models, a `Model` interface to hold a semantic model as asserted individuals and a `ReasonedModel` interface to hold a semantic model as asserted individuals and inferred individuals reasoned by a reasoner from those

stored asserted individuals. A `Query` interface is provided to hold and manage queries, and a `Rules` interface holds and manages rule sets.

Each of the four interfaces provided by the `semanticcomponents` package can have multiple implementations using different specific semantic components and libraries. In this way, the semantic frameworks, libraries, and toolsets that underpin Aesop can be changed, enhanced, and tuned without changing the design or implementation of the Aesop Knowledge Base or Engine.

4.3.2 The Knowledge Base

In this section, the structure of the Knowledge Base and the knowledge that must be configured in the Knowledge Base in order for Aesop to execute as an autonomic end user service management system are described.

4.3.2.1 Knowledge Base Structure

The Knowledge Base holds the end user service delivery knowledge of the Aesop system modelled using the EUSAO ontology described in Chapter 3.

The knowledge holds the following types of knowledge:

1. **Metadata**: the EUSAO ontology, specifying the structure of the Knowledge Base.
2. **Global Knowledge**: concepts in the EUSAO ontology with a *global temporality* property, as explained in Section 3.3. This knowledge is knowledge that does not have a temporal dimension such as configured service expectations or the terminal on which an end user session is running.
3. **Snapshot Knowledge**: concepts in the EUSAO ontology with a *snapshot temporality* property, as explained in Section 3.3. This knowledge changes during execution of a service session and is made up of concepts such as service quality metrics, latency on connections carrying services, and the location of the user using a service. The snapshot design choice was inspired by the similar concept of partitioning used in relational databases to improve performance on large tables [Coronel et al., 2009].

The Knowledge Base holds its knowledge in a number of models, as shown in Figure 4-1 and Figure 4-2. Semantic query, rule, and reasoning operations perform well on small models but performance deteriorates significantly as the size of the models on which those operations act increases [Keeney et al., 2011]. Using a number of models allows requests by semantic operations to the knowledge base to be scoped

to a small number of models thus maximizing the performance of those operations. In order to keep the size of knowledge base as small as possible, the knowledge in all models in the knowledge base is held as the asserted individuals explicitly stored in the models.

The first model in the Knowledge Base is the *metadata model*. This model holds the structure or Tbox [Nardi & Brachman, 2007] of the knowledge in the Knowledge Base and is loaded at system start. The metadata is the EUSAO ontology. The metadata model also holds any SWRL rules specified on the EUSAO ontology.

A number of models are used to hold individuals or Abox [Nardi & Brachman, 2007], the knowledge in the Knowledge Base, which is partitioned as global knowledge and snapshot knowledge using the temporality property of concepts.

Global Knowledge is further partitioned into two models. A *static global model* holds global concepts with a *static* dynamicity property, and a *dynamic global* model holds global concepts with a *dynamic* dynamicity property, as explained in Section 3.3. The static global knowledge model holds individuals that are static or configured such as user expectations. The dynamic global knowledge model holds individuals that are loaded from incoming service session knowledge and do not change over the duration of service sessions such as the user that is running a session or the terminal on which a session is terminated. Only one individual of a concept stored as global knowledge may be associated with a particular service session.

The concept of a snapshot is fundamental to the performance of the Aesop Knowledge Base. Snapshots represent the state of a concept for a particular time period, as explained in Section 3.3. All the snapshots for a time period are partitioned into a *snapshot bucket* for that time period. The snapshot concept is similar to the concept of *horizontal fragmentation* used in relational database systems to increase the performance of queries on large tables [Coronel et al., 2009]. Tables are partitioned into fragments based on a single attribute such as the timestamp in each row. Relational database management systems such as PostgreSQL [The PostgreSQL Group, 2013] support partitioning transparently.

The Knowledge Base maintains a model for each snapshot bucket in addition to the metadata model and the two global knowledge models, as shown in Figure 4-2. All individuals of with dynamic dynamicity and snapshot temporality for the time period of a snapshot bucket are stored in that snapshot bucket model. Individuals in a

snapshot bucket reference other individuals in the same snapshot bucket or individuals in the global knowledge models, but never reference individuals in other snapshot buckets.

Aesop functions such as Knowledge Monitoring, Semantic Analysis, and Optimization Planning and Execution use semantic operations that act on temporary reasoned models created from subsets of the Knowledge Base, usually composed of the metadata, both global knowledge models, and one or more snapshot bucket models, as shown in Figure 4-2. For example, during analysis of the compliance of end user sessions with expectations for a certain time period, Aesop builds a model containing the knowledge in the metadata model, the global knowledge models, and the snapshot models that span the time period in question. This model is sufficient for full analysis of compliance for the period in question. Using this approach greatly reduces the size of models used for semantic operations, thus increasing the performance of those semantic operations.

4.3.2.2 Configured Knowledge in the Static Global Knowledge Model

In order for Aesop to operate as an autonomic system for end user service management, some knowledge must exist in the Knowledge Base. This knowledge, which is stored in the static global knowledge model, must be configured and inserted into the Knowledge Base.

The static concepts and relationships in the EUSAO ontology shown in Figure 3-2 that are configured in the static global knowledge model are shown in Table 4-1.

4.3.3 The Knowledge Bus

Aesop engine components use the Knowledge Bus interface to access knowledge in the model, as shown in Figure 4-2. When accessing knowledge, components use the interface to specify the type of knowledge required, which can be *static global*, *dynamic global*, and/or *snapshot* knowledge. For snapshot knowledge, a time range can be specified to filter the snapshot buckets from which knowledge is returned. Filters can also be used to specify the types of individuals to be returned. The Knowledge Bus copies the requested individuals from the Knowledge Base into a new model and returns that model to the caller. The caller can request two types of returned model: a plain model containing only asserted individuals or a reasoned model containing asserted and inferred knowledge. The component can then use the

information in this copied model without any risk of interference with other operations using the Knowledge Base.

Table 4-1: Static Global Knowledge Configured in Aesop

Concept	References From	Description
NetworkComposite (SID)	*None*	A single individual representing the network on which end user services are running.
ConnectivityService a ServicePackage(SID)	*NetworkComposite (SID)*	An individual per connectivity service offered on the network on which services are running.
Subsciber an Individual(SID) a Person(FOAF)	*ServicePackage(SID) ServiceExpectation*	An individual for each subscriber that has subscribed to a connectivity service on the network.
SSM_Service	*None*	An individual for each end user service offered on the network.
ServiceExpectation	*Subscriber Service*	An individual for every end user service to which each individual subscriber has subscribed. A *Service Expectation* individual specifies the particular expectations a subscriber has on a service type
User a Person(FOAF)	*ServicePriority*	An individual for each authorized user of services on the network.
ServicePriority	*Service User*	An individual for each service that a particular user has access to. A *Service Priority* individual specifies the priority a particular user has on a particular service and the throttling parameters that will be applied to sessions of that particular service type for that user if selected for throttling

The Knowledge Bus interface also allows components to add and update knowledge to the Knowledge Base. The interface provides support for adding, deleting, and updating individuals in the Dynamic Global model and provides specific support for managing knowledge in snapshot buckets. When a snapshot individual is added to the Knowledge Base, the Knowledge Bus uses the time stamp of the individual to ensure that the individual is placed in the correct snapshot bucket transparently. If no snapshot bucket exists for the time stamp of the incoming snapshot individual, a new snapshot bucket is created prior to insertion of the snapshot individual. The Knowledge Bus also supports update operations on individuals in snapshot buckets and deletion of snapshot buckets.

The Knowledge Bus interface is designed so that it may have many implementations. This design allows different implementations of the Knowledge Base or Knowledge Bus itself to be used without affecting the Aesop Engine components because the interface provided by the Knowledge Bus does not change if the implementation changes.

Simultaneous access operations by Aesop Engine components to the Knowledge Base must be guaranteed not to interfere with each other; prevention of interference is the responsibility of the Knowledge Bus implementation. A simple implementation might use a queuing mechanism that prevents interference by enforcing sequential access to the Knowledge Base; such an implementation would have the drawback of acting as a bottleneck in the system. A faster and more elegant implementation would provide concurrency control [Coronel et al., 2009] using mechanisms such as transactions and locking that are found in relational databases such as PostgreSQL [The PostgreSQL Group, 2013].

4.3.4 The Aesop Engine

The design of the Aesop Engine and its components is shown in Figure 4-2. The engine executes operations, which are modelled as an *AesopOperation* instance. An *AesopOperation* instance always uses a *TemporaryModel*. A *TemporaryModel* is a reasoned model; a reasoner runs over the model and uses the EUSAO ontology to infer new knowledge when the model is created and whenever knowledge is added to the model. An *AesopOperation* may be an *InvokedOperation* triggered by an event such as the arrival of new knowledge into the system, or a *PeriodicOperation* triggered by a timer.

Each *AesopOperation* concrete specialization is designed to run a semantic algorithm. That algorithm may use one or more *OperationQuery* instances and *OperationRule* instances to find, assert, or infer knowledge on the *TemporaryModel*, and may store knowledge back into the Knowledge Base. The choice of whether to use queries, rules, or both is a matter for the design of the algorithm in a particular *AesopOperation* specialization. The actual queries and rules used by that *AesopOperation* specialization are specific to the purpose of the semantic algorithm that specialization runs.

The design of the Aesop engine generalizes the usage of models, queries and rules by *AesopOperation* specializations. They all use temporary models in the same way,

and use the same mechanisms for handling queries and rule sets. Timing handling of `PeriodicOperation` instances is also generalized. Even though specializations of `AesopOperation` have specific semantic algorithms, all those algorithms are run in a common way in the engine. This design allows further semantic algorithms to be designed; a new algorithm would be developed as a new `AesopOperation` specialization.

4.3.4.1 Knowledge Monitoring

In Aesop, the *Knowledge Monitoring* component realises the autonomic *monitoring* function. It takes end user service experience and context information from external sources, transforms that information into knowledge as individuals of the EUSAO ontology, and stores that knowledge in the Knowledge Base.

Knowledge monitoring has three distinct functions, as shown in Figure 4-3, a *Knowledge Lifting* function that translates data from non-semantic sources into semantic form, a *Knowledge Loading* function takes that knowledge, infers new knowledge using reasoning and rules, and stores the knowledge into the Knowledge Base, and an *Information Stream Handling* function that handles interfaces towards network nodes, domain managers, or external systems that are supplying information to Aesop.

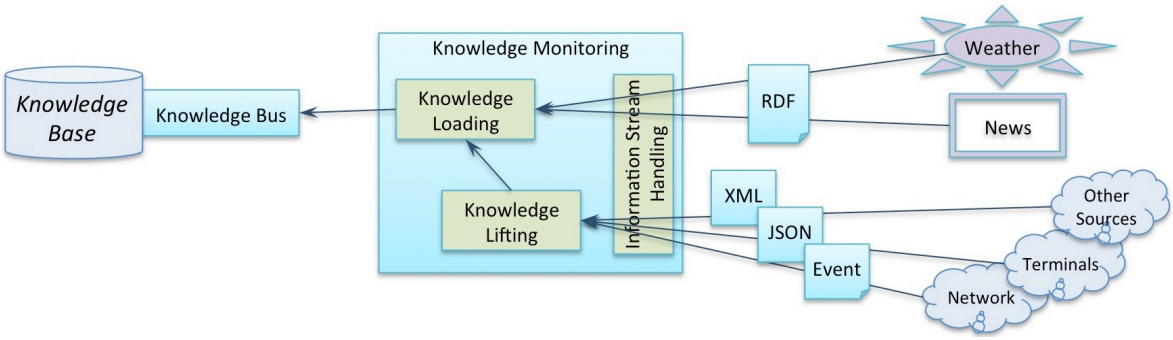


Figure 4-3: Knowledge Monitoring

Knowledge lifting takes information from external systems that is not in semantic form and maps it into semantic knowledge as individuals in the EUSAO ontology or another ontology that the EUSAO ontology references. Knowledge lifting receives a stream of information, which may be in formats such as XML documents, JSON objects [IETF, 2006a] or Syslog events [IETF, 2009]. Knowledge lifting employs techniques such as SAWSDL [W3C, 2007] and XSL transformations [W3C, 1999b] to lift XML documents. For example, terminal reports in XML format can be lifted into semantic

form to determine the service experience and context of end user services. Lifted information is passed to Knowledge Loading for storage into the Knowledge Base.

Knowledge Loading receives individuals of concepts in the EUSAO ontology or an ontology referenced by the EUSAO ontology and stores it into the Knowledge Base. Knowledge is received from Knowledge Lifting and from external systems that supply knowledge in semantic form.

The Aesop Knowledge Loading semantic algorithm shown in Figure 4-4 was developed to load dynamic global knowledge and snapshot knowledge into the Knowledge Base. The algorithm runs in instances of the *KnowledgeLoading* specialization of an *AesopOperation*, as shown in Figure 4-2. The Knowledge Loading algorithm uses a set of domain specific rules to infer data properties on dynamic global and snapshot individuals. Rules rather than queries are used for this algorithm because extra rules added to the rule set of an operation will fire anonymously whereas queries must be explicitly invoked. The input knowledge and the data properties inferred during loading is likely to be quite diverse so the more flexible rule-based approach to inference was chosen for use with this algorithm.

A temporary reasoned model composed of the metadata and global knowledge from the Knowledge Base is created. The incoming knowledge is added to the temporary model, causing the reasoner to run and infer extra knowledge using the structure of the EUSAO ontology. The rule set of the *KnowledgeLoading* individual now fires, causing data properties to be set in dynamic global and snapshot individuals. Loading of dynamic global and snapshot knowledge now proceeds.

The dynamic global individuals for each dynamic global concept are read from the temporary model, and the individuals and their properties for each concept are stored to the Knowledge base. Similarly, the snapshot individuals are read from the temporary model one by one and stored to the Knowledge Base together with their properties. As explained in Section 4.3.3, the knowledge bus ensures that each snapshot is stored in the correct snapshot bucket in the knowledge base. When a snapshot is stored to a snapshot bucket, a dynamic property on the snapshot bucket individual is marked as being dirty, a flag for Semantic Analysis to analyse that snapshot bucket. All dynamic global knowledge and snapshot knowledge is stored as asserted individuals and properties, even if those individuals and properties were

inferred in the temporary model during execution of the Aesop Knowledge Loading algorithm.

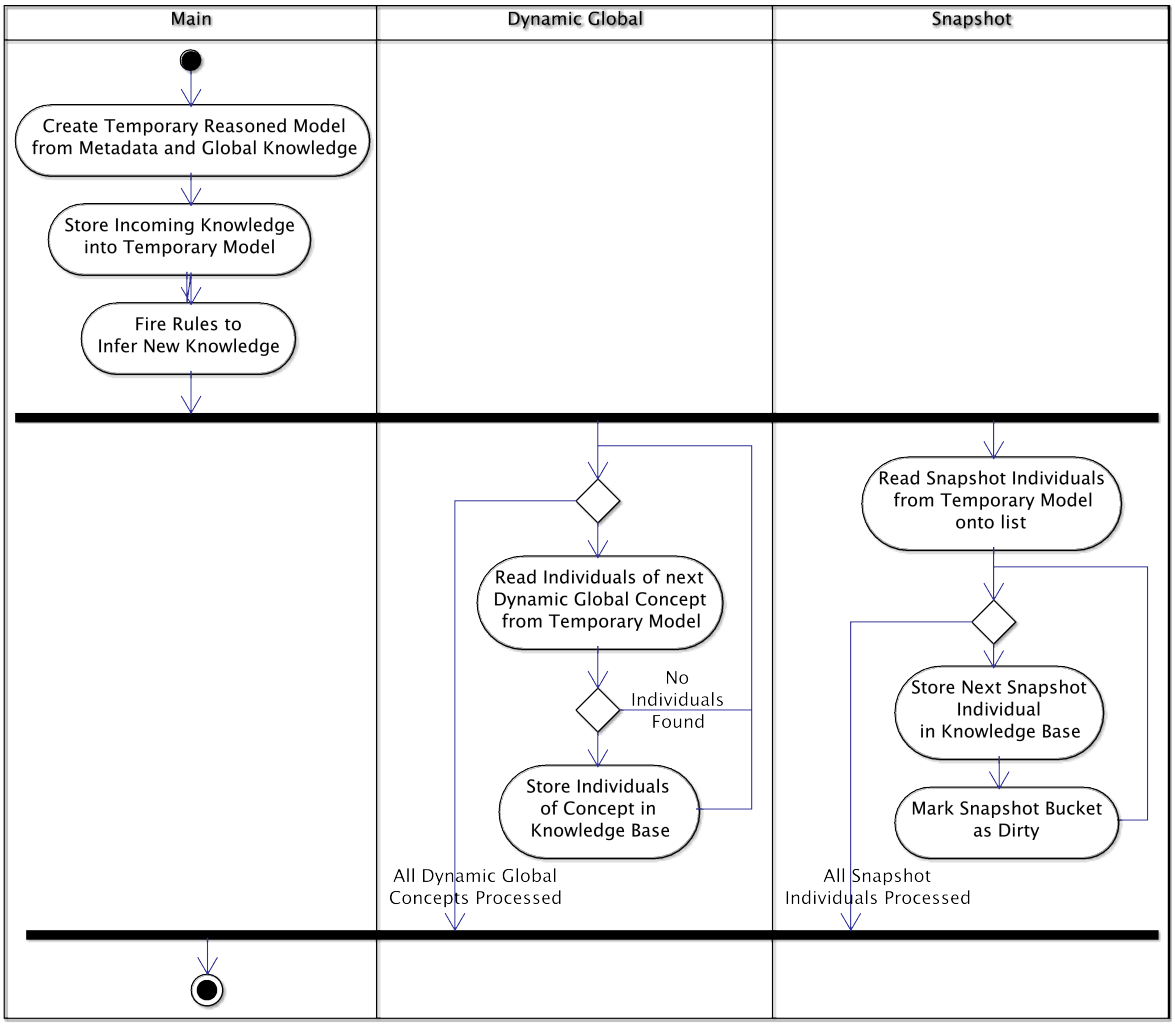


Figure 4-4: The Aesop Knowledge Loading Algorithm UML Activity Diagram

The Information Stream Handling function provides an interface that transparently passes information to Knowledge Lifting and Knowledge Loading as a stream, as shown in Figure 4-2. Implementations of Information Stream Handling abstract interfaces towards information sources such as files, streams, or protocols such as XMPP [IETF, 2011]. Information Stream Handling implementations manage the communication mechanism used by the information source in question, reads incoming information, and streams it towards Knowledge Lifting or Knowledge Loading.

Using this approach for semantic monitoring restricts the scope of reasoning and rule execution to a subset of the total knowledge in the Knowledge Base during Knowledge Loading. It also allows extra rules to be deployed to reason more knowledge if required.

4.3.4.2 Semantic Analysis

The *Semantic Analysis* component realises the autonomic *analysis* function. End user service experience and context stored in the Knowledge Base by Knowledge Monitoring is analysed to determine its compliance with expectations.

The Semantic Analysis algorithm designed for Aesop is shown in Figure 4-5. The algorithm runs periodically at a configurable interval as a `SemanticAnalysis` instance, a specialization of an `AesopOperation PeriodicOperation` as shown in Figure 4-2. It analyses each EUSAO `ServiceExperienceSnapshot` individual in each snapshot bucket that has been updated since the last run of the Semantic Analysis algorithm. For each `ServiceExperienceSnapshot` individual, it calculates a metric that represents the compliance of that `ServiceExperienceSnapshot` to its expectations and stores that metric back into that individual in the snapshot bucket model. The algorithm was designed using an approach that is similar to a database transaction; it reads knowledge from the knowledge base using the least number of reads possible and writes all the knowledge to the database in a single operation. The algorithm was designed to use semantic queries rather than rules because the request-response nature of queries and the fixed return values expected by the algorithm steps is more suited to queries.

The Aesop Semantic Analysis algorithm runs iteratively over each dirty snapshot bucket. For each dirty snapshot bucket it creates a temporary reasoned model composed of the metadata, global knowledge, and the snapshot in question. Composition of the model causes the reasoner to infer all non-asserted relations defined in the EUSAO ontology in the temporary model. The Snapshot Bucket is immediately marked as being clean because the algorithm does not read it again. The knowledge in Snapshot Bucket will not be analysed again unless it is marked dirty by a subsequent Knowledge Loading operation. A generic query that uses the subsumption hierarchy of the EUSAO ontology is run on the temporary model to find new or updated `ServiceExperienceSnapshot` individuals and their equivalent `ServiceExpectation` individuals. The service type of a `ServiceExperienceSnapshot` individual is used to find a service specific query for each `ServiceExperience Snapshot` individual found, into which the URIs of the `ServiceExperienceSnapshot` individual and `ServiceExpectation` individual are passed as parameters. This query is run on the temporary model to determine the compliance of that particular `ServiceExperienceSnapshot` individual with that particular `ServiceExpectation`

individual. The compliance of that `ServiceExperienceSnapshot` individual is then saved on a working list in the `SemanticAnalysis` instance. When all Snapshot Buckets have been analysed, the compliance values for all `ServiceExperienceSnapshot` individuals on the working list are saved to the Knowledge Base as a single operation.

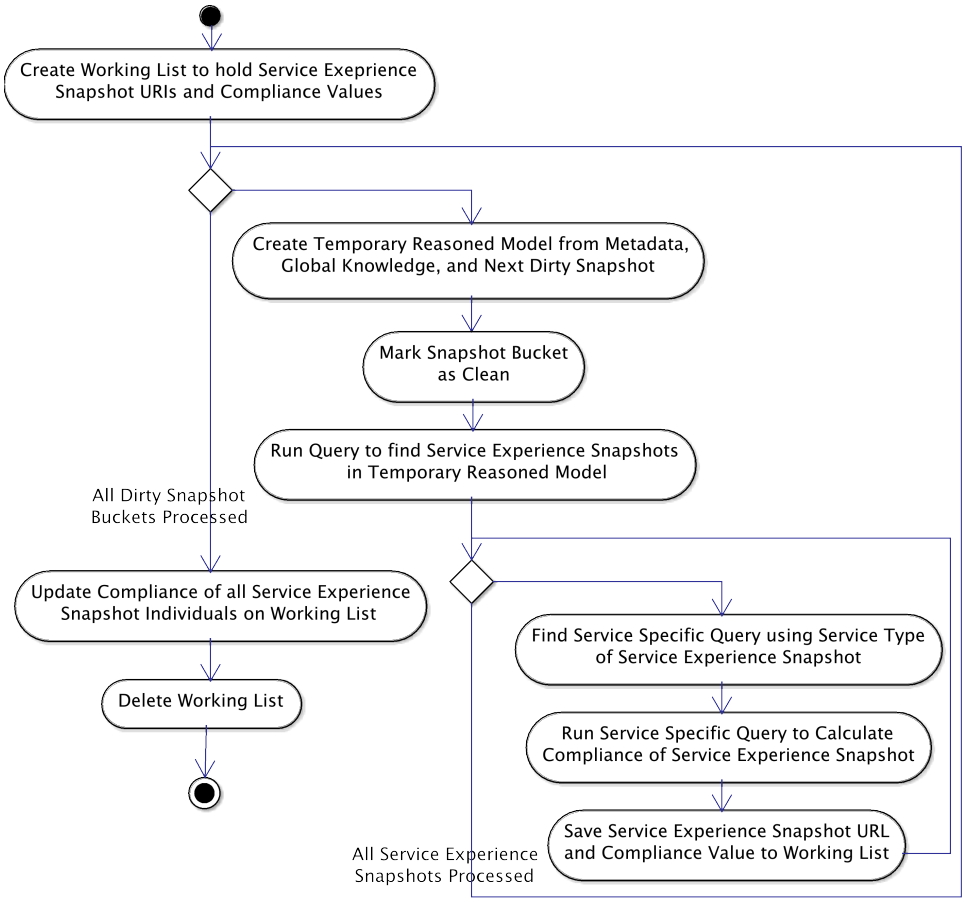


Figure 4-5: The Aesop Semantic Analysis Algorithm UML Activity Diagram

The approach used in the design of the component and algorithm for Semantic Analysis is efficient in two important manners. Firstly, it restricts reasoning during semantic analysis to global individuals and the individuals of a single snapshot, thus reducing the scope over which reasoning and queries execute. Secondly, it is efficient in its use of the knowledge base, performing only one Knowledge Base read for each Snapshot Bucket and updating the Knowledge Base in a single operation. In addition, at run time, queries can be updated to change calculations and inferences for determining compliance for existing services, and queries can be added to support new services.

4.3.4.3 Optimization Planning and Execution

The *Optimization Planning and Execution* component of Aesop realises the autonomic *plan* and *execute* functions. *Optimization Planning* examines the

compliance of the current running set of end user sessions to decide if execution of an optimization action is appropriate. If Optimization Planning decides to execute an optimization action, it calls *Optimization Execution* to execute that optimization action towards the network. The Optimization Planning and Execution component runs periodically at a configurable interval.

The Optimization Planning and Execution semantic algorithm designed for Aesop is shown in Figure 4-6. The algorithm runs periodically as an `OptimizationPlanning` instance, a specialization of an `AesopOperation PeriodicOperation`, as shown in Figure 4-2. It optimizes the current set of end user sessions by applying and releasing throttling actions on sessions. If the compliance of high priority sessions is degraded for a configurable time, Aesop progressively throttles lower priority sessions so that extra network resources are made available for high priority sessions. If high priority sessions are no longer degraded for a configurable time, Aesop progressively releases throttling on lower priority sessions so that their compliance can improve. The way in which this algorithm adapts traffic to cope with currently available networks resources was inspired by streaming algorithms such as HTTP-DASH [3GPP, 2012c] that dynamically adapt video quality as available session bandwidth changes. As in the case of the Semantic Analysis algorithm, this algorithm was designed to use semantic queries rather than rules because the request-response nature of queries and the fixed return values expected by the algorithm steps is more suited to queries.

When the Optimization Planning and Execution algorithm executes, it runs two sub-activities as shown in Figure 4-6, once to plan and execute application of throttling actions, and once to plan and execute release of throttling actions. While the same sequence of actions is used in both sub-activities, the input parameters and queries used by the sub-activities are different in each case.

Checking and Application of Throttling Actions

The first sub-activity of the Optimization Planning and Execution algorithm checks if throttling application is necessary to improve the compliance of the running set of high priority end user sessions and, if so, applies throttling. A temporary reasoned model is created composed of the metadata, global knowledge, and a set of recent snapshots. The number of snapshots stored in the temporary reasoned model is the configurable *lookback interval*, which controls how far back in time the optimization algorithm will go when examining the compliance of sessions with expectations. A

reasoner automatically infers all non-asserted relations defined by the EUSAO ontology in the temporary reasoned model when it is created.

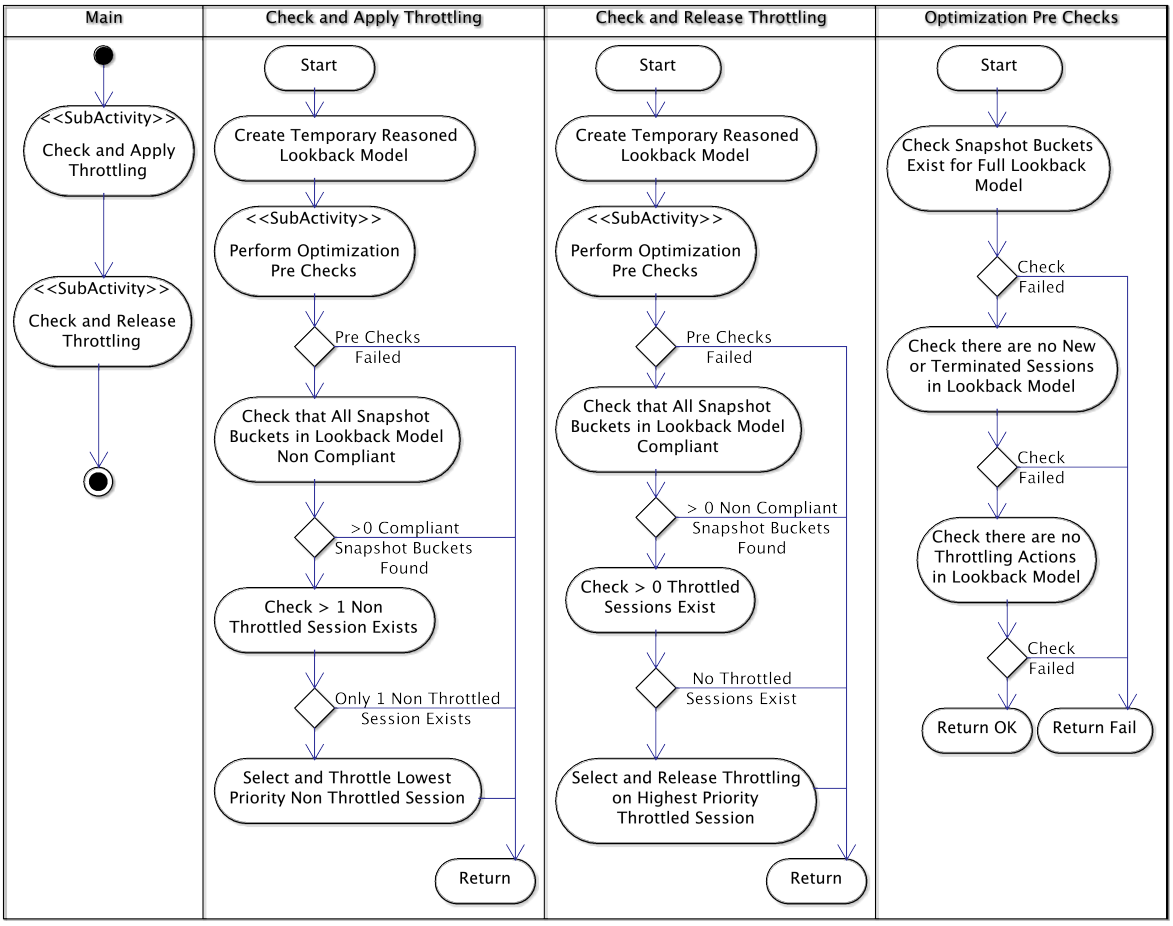


Figure 4-6: The Aesop Optimization Planning and Execution Algorithm UML Activity Diagram

The Check and Apply Throttling sub-activity of the algorithm then executes a pre check sub-activity, which carries out the following series of pre-checks on the knowledge in the temporary reasoned model using a set of queries that have been initialized in the `OptimizationPlanning` instance running the Optimization Planning and Execution algorithm, one for each pre-check.

1. Is there knowledge in the temporary model for the entire lookback interval? This check is performed to prevent optimization when there are not enough snapshot buckets in the Knowledge Base, which can occur, for example, at system start.
2. Are there new or terminated sessions in the temporary model? This check is performed to prevent optimization for at least the lookback interval when a new session has started or when a session has terminated. This check is used to prevent optimization during the period when the session delivery context is adjusting to the presence of a new session or the absence of a terminating session.

3. Are there sessions on which throttling actions have been performed in the temporary model? This check is performed to prevent optimization for at least the lookback interval when a throttling action has been performed on a session. This check is used to prevent optimization during the period when the session delivery context is adjusting to a throttling action application or removal on a session.

If all pre-checks pass, the Check and Apply sub-activity of the Aesop Optimization Planning and Execution algorithm uses a query to check if all snapshot buckets in the temporary model are degraded; that they contain non-throttled sessions that are non compliant with expectations. This check ensures that degradation has occurred in all snapshot buckets for the lookback interval. If this check passes, a further query is used to check that at least two non-throttled sessions are running. If so, a final query is used to find the lowest priority session on which throttling has not been applied and to read the throttling parameters that should be set on that session. Those parameters are read from the appropriate *`ServicePriority`* individual in the static global knowledge of the Knowledge Base for the user running the session. Throttling is applied on the session using those parameters by calling an *`OptimizationExectutionImpl`* instance using the *`OptimizationExecution`* interface, as shown in Figure 4-2.

The *`OptimizationExecutionImpl`* instance throttles the session in the manner appropriate to the networking domain in which the end user service sessions are running. Optimization Execution sets the session as throttled and records the throttling action performed in service session individual in the dynamic global model of the Knowledge Base. It also creates an *`SSM_OptimizationSnapshot`* individual in the current snapshot model of the Knowledge Base as a marker that an optimization action has been performed.

Checking and Release of Throttling Actions

The second Check and Release Throttling sub-activity of the Aesop Optimization Planning and Execution algorithm, shown in Figure 4-6, checks if it is appropriate to release throttling to improve the overall compliance of the running set of end user sessions and, if so, releases throttling on an end user session. A temporary model is created as in the Check and Apply Throttling sub-activity. A lookback interval suitable for release checking is specified which should be longer than for application runs to

avoid interference between the run types. The same pre check sub-activity is called as in the Check and Apply Throttling sub-activity.

The Check and Release sub-activity runs a query that checks if all snapshot buckets in the temporary model are compliant; that all non-throttled sessions in all snapshot buckets are compliant with expectations. This check ensures that all snapshot buckets are compliant for the lookback interval. If this check passes, a query is used to check that at least one throttled session is running. If so, the throttled session with the highest priority is selected for release of throttling using a further query. Throttling is released on the session by calling an *`OptimizationExectutionImpl`* instance using the *`OptimizationExecution`* interface, as shown in Figure 4-2.

The *`OptimizationExecutionImpl`* instance releases throttling on the session in the manner appropriate to the networking domain in which the end user service sessions are running. Optimization Execution sets the session as not throttled and records the throttling action performed in service session individual in the dynamic global model of the Knowledge Base. It also creates an *`SSM_OptimizationSnapshot`* individual in the current snapshot model of the Knowledge Base as a marker that an optimization action has been performed.

`OptimizationExectutionImpl` instances handle interfaces towards network nodes or domain managers, carrying out whatever operation is necessary to apply and release throttling on a session in the network. Typical operations might be execution of a command script, application of a policy or a firewall rule, or invocation of an operation on an information base.

The design used for Optimization Planning and Execution and its algorithm is efficient because, in both runs, temporary models use lookback intervals to reduce the amount of knowledge that must be queried during optimization. In addition, it is straightforward to add, change, or remove checks carried out prior to throttling actions because they are all implemented as queries.

4.3.4.4 Support Functions

The support functions of the Aesop Engine are not part of Aesop's autonomic loop, but provide functions required to make Aesop a functioning and useful system. The support functions shown in Figure 4-1 include a Northbound Interface component that exposes public semantic and non-semantic interfaces for a user interface client or for other management systems, a Knowledge Persistence component used to load

and store knowledge from and to persistent storage, and a Knowledge base Maintenance component that performing initiation and housekeeping tasks such as cleaning expired knowledge.

4.4 Implementation

A prototype implementation of Aesop, shown in Figure 4-7, has been developed as a Java application that optimizes delivery of the end user service sessions running in a Home Area Network. It uses terminal reports sent by terminals in a home area network to read end user service experience and context, and optimizes service delivery by sending Linux traffic controls [Netfilter Framework, 2012] to the home gateway to throttle end user sessions.

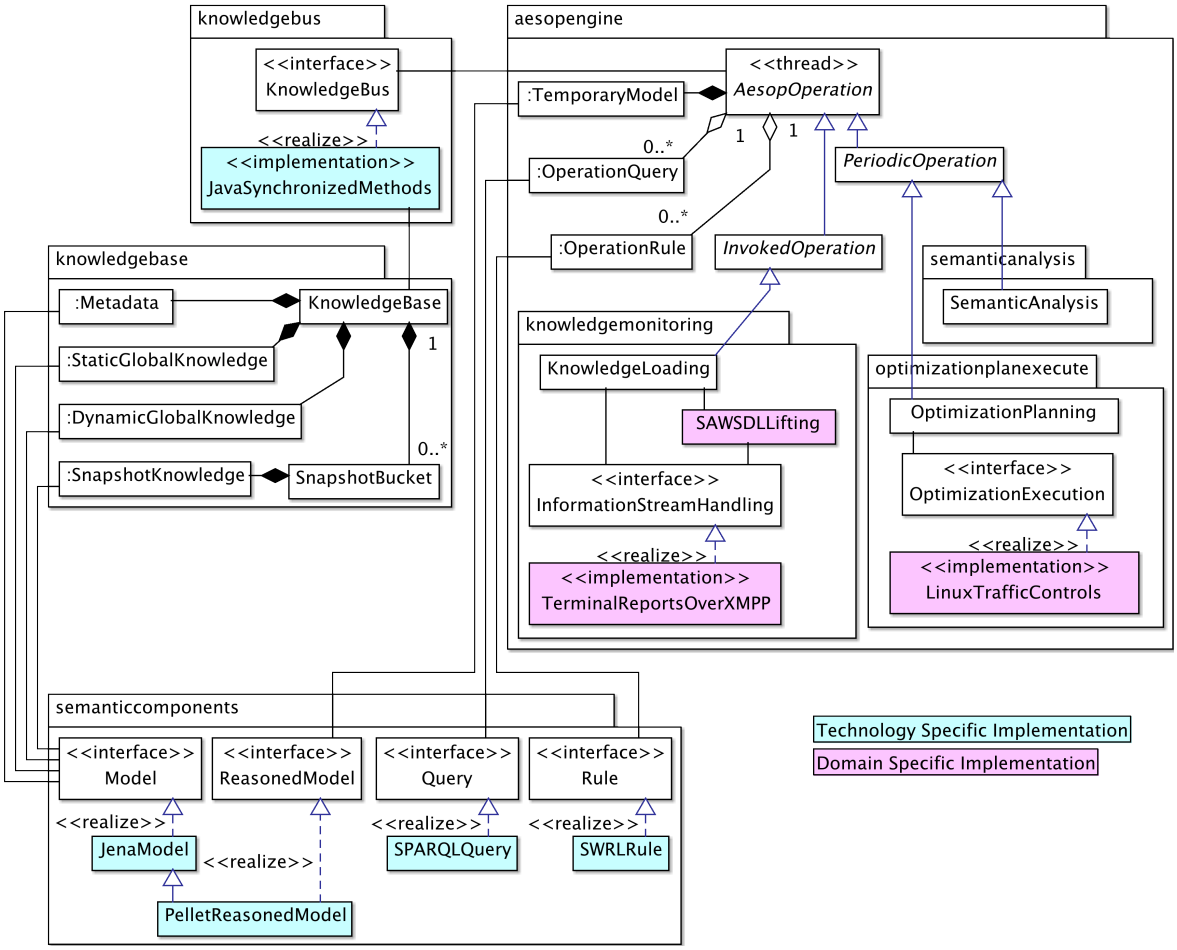


Figure 4-7: UML Class Diagram of a Prototype Implementation of Aesop

The classes in light blue in Figure 4-7 show where technology specific implementation choices were made and the classes in pink show where domain specific implementation choices were made. All other classes in the design are implemented in a domain and technology independent manner in the prototype. Technology specific implementations are used to provide the semantic components required by Aesop and to implement the knowledge bus. Domain specific

implementations are used to monitor knowledge coming from the Home Area networking domain and to apply throttling actions to that domain. As can be seen from Figure 4-7, in all but one case, specific design choices are encapsulated behind interfaces. In the case of the *SAWSDLLifting* implementation, this implementation is encapsulated in the *knowledgemonitoring* package and the implementation is applicable to any text-based input source, as will be explained in Section 4.4.4.1.

4.4.1 Semantic Components

In making choices on which semantic components to use in Aesop, the fifth requirement in Section 4.2 provided guidance. Readily available components with a record of use in semantic applications that meet the needs of Aesop were chosen for use in the prototype. The evaluation of the performance of the prototype implementation of Aesop described in Section 6.6 can be used to assess performance of individual semantic components and decisions can be taken on whether it is necessary to tune, enhance, or replace components to improve the performance of the Aesop implementation.

Apache Jena [Apache, 2013c] was chosen as a semantic framework for handling semantic models because it provides APIs (Application Programming Interfaces) for handling RDF and OWL models and supports queries towards models using the SPARQL semantic query language [W3C, 2008]. Jena is widely used and is integrated with other semantic tools such as Protégé [Stanford University, 2012] and the Pellet reasoner [Clark & Parsia, 2013]. The *Model* interface in Figure 4-7 was implemented using Jena models.

As explained in Section 3.4, OWL 2 [W3C, 2009] was chosen as the language for semantically representing the EUSAO ontology. The Pellet reasoner [Clark & Parsia, 2013] was selected as a reasoner for the Aesop prototype implementation because it can be used at ontology implementation time in Protégé and can be used transparently with Jena to provide the reasoned models the Aesop engine *AesopOperation* specialized classes need at run time. The *ReasonedModel* interface in Figure 4-7 was implemented as Jena models reasoned using the Pellet reasoner.

SPARQL was selected as the language of implementation for semantic queries in the Aesop prototype implementation because it is the recommended RDF query language specified by the W3C and because it is very widely used in the semantic community. The *Query* interface in Figure 4-7 was implemented as a *SPARQLQuery*

Java class. That class responds to a request for a query by name on the *Query* interface by reading the text of a SPARQL query from a directory on the file system and returning it to the caller.

SWRL [W3C, 2004c] was selected as the semantic rule language in the Aesop prototype implementation because, in addition to being a member submission of the W3C and having a solid record of use in research [Latre et al., 2008][Rana et al., 2011], it is the semantic rule language supported in Protégé and the Pellet reasoner. The Rule interface in Figure 4-7 was implemented by providing a Java method that reads the text of SWRL rules from a directory on the file system and stores them into a reasoned model.

4.4.2 The Knowledge Base

When the prototype is started, it reads metadata, SWRL rules, and static global knowledge into the Knowledge Base from the file system.

Each model in the Knowledge Base is held in a Java class as a Jena OWL model. Snapshot bucket models are held in a hash map indexed using the starting time of the snapshot bucket. To ensure efficient use of memory, all models are held as simple ontologies with asserted individuals, no reasoned or inferred individuals are held in the Knowledge Base. A snapshot bucket retention time of 10 minutes was used, allowing the last 10 minutes of snapshot knowledge to be queried directly from the Knowledge Base.

The metadata for the EUSAO ontology was loaded into the metadata model. The metadata for the referenced ontologies shown in Table 3-3 was not loaded because it is not required for reasoning. The EUSAO ontology uses specialized concepts from those ontologies and references asserted individuals in static global knowledge. Because the Aesop prototype implementation infers knowledge only from the EUSAO ontology, it is necessary only to have the metadata (or Tbox) of that ontology available for reasoning at run time. As explained in Section 3.4, the metadata of those ontologies was loaded into Protégé when the EUSAO ontology was being implemented, and the Pellet reasoner was used to ensure that the specializations and references in the EUSAO ontology to those ontologies were valid and consistent. Loading the metadata of those ontologies into the Aesop knowledge base at run time would have caused reasoning to run over those ontologies on creation of temporary reasoned models, causing unnecessary extra load on the system.

In the prototype, all static global knowledge individuals listed in Table 4-1 were specified in Protégé [Stanford University, 2012] and are loaded at system start by Knowledge Base maintenance from OWL files. Figure D-1 in Appendix D shows the specification of *SSM_ServiceExpectataion* and *SSM_ServicePriority* individuals. In a deployed system, expectations should be set using a client interface such as in Sventek et al. (2011).

4.4.3 The Knowledge Bus

The Knowledge Bus provides unreasoned or reasoned models to calling components that contain a filtered subset of the knowledge in the Knowledge Base. It is implemented as a set of thread safe synchronized Java methods, allowing Aesop operations such as monitoring, analysis, and Optimization Planning and Execution to run in parallel. Using this implementation means that only one operation can access the Knowledge Base at a time because Java synchronization queues the calls to the knowledge base.

When a reasoned model is instantiated by the Knowledge Bus, the reasoner runs, leading to inference of new knowledge. The SWRL rules shown in Section C.1 of Appendix C are also loaded into the reasoned model and fired to infer further knowledge.

The most important methods on the Knowledge Bus interface implemented in the prototype are described in Table D-1 of Appendix D. These methods copy entire models. Support for filtering by concept type on Knowledge Bus is not implemented in the prototype.

4.4.4 The Aesop Engine

When the Aesop engine starts, it initiates timers with an interval of 10 seconds that are used for Semantic Analysis, Semantic Optimization, and Knowledge Base Maintenance Aesop *PeriodicOperation* instances. The snapshot bucket interval is also set at 10 seconds. A 10 second interval is used because it is sufficiently long to mitigate momentary network fluctuations, is frequent enough to be perceived as being near real time, and is a reasonable compromise between speed of operation and resource consumption by Aesop. The time from reception of a terminal report to the report being analysed is at most 20 seconds and the time from an observation being loaded into the system to that observation being considered in optimization is at most 30 seconds.

4.4.4.1 Knowledge Monitoring

The prototype implements the functions of Knowledge Monitoring shown in Figure 4-3. It monitors knowledge reported by terminals in the form of XML terminal reports carried over XMPP [IETF, 2011], which report metrics on the connections and end user service sessions running on each terminal and on the connections in the home gateway. That knowledge is lifted into individuals of the OWL ontology and is loaded into the Knowledge Base.

Information Stream Handling opens and maintains an XMPP client connection towards the XMPP server. When it receives a terminal report as an XML document, it passes it to Knowledge Lifting and Knowledge Loading for processing. A new Aesop `InvokedOperation` thread is spawned to handle lifting and loading of each terminal report, as shown in Figure 4-7. This allows Aesop to process many terminal reports simultaneously.

Knowledge lifting receives terminal reports at service session start, session termination, and periodically every 10 seconds during a session. Knowledge is lifted from terminal reports into individuals of the EUSAO ontology specified in RDF [W3C, 2004a] using XSL transformations [W3C, 1999b] triggered by SAWSDL annotations [W3C, 2007]. Those annotations mark the schema elements from which semantic information will be extracted. Examples of SAWSDL annotations on element definitions of an XML schema for terminal reports are shown in Listing D-1 of Appendix D.

At system start, Knowledge Lifting reads the XML schema document for terminal reports and builds an Annotated Element Lookup Reference (AELR) for each element annotated with SAWSDL, as shown in Table 4-2. The AELR records the XSL transformation to be used to lift an XML element in a terminal report into EUSAO individuals and an XPath expression [W3C, 1999a] that can be used to find occurrences of that element in a terminal report.

When Knowledge Lifting receives a terminal report, the XPath expressions in each AELR entry are run against the terminal report document, producing a list of matching nodes. The XSL transformations specified in the AELR entry are then run on each node in turn producing EUSAO individuals in RDF. Those individuals are forwarded to Knowledge Loading for storage into the Knowledge Base.

Table 4-2: An Annotated Element Lookup Reference (AELR) entry

Field	Description
Name	Element or attribute name
Type	XML *Element* or *Attribute*
XPath Expression	XPath expression identifying the location of this element or attribute in the XML hierarchy of a terminal report
Transformation References	List of URIs identifying XSL transformations to run to lift this element into EUSAO individuals

The XSL transformations used to lift elements to EUSAO individuals create the individuals of the dynamic concepts shown as ovals in Figure 3-1 and the references of those concepts shown as blue arrows in in Figure 3-1. The full list of XSL transformations used in Aesop appears in Table D-2 of Appendix D. The XSL transformations used to lift the *Terminal* and *User* terminal report XML elements to a *SSM_Terminal* and *SSM_User* EUSAO individual appear in Listing D-2 and Listing D-3 of Appendix D.

The EUSAO individuals in RDF format are sent by Knowledge Lifting to Knowledge Loading and are loaded into the temporary working reasoned model and stored in the Knowledge Base using the Aesop Knowledge Loading semantic algorithm shown in Figure 4-4.

The Knowledge Loading algorithm uses SWRL rules to infer new knowledge during the loading process. Listing C-1 in Appendix C shows the SWRL rule that infers the priority of a new service session by using the *SSM_User* and *SSM_Service* of the incoming session to reference the *SSM_ServicePriority* for that service type in static global knowledge.

For test and debugging purposes, Aesop can monitor terminal reports in "file" mode. In this mode, Knowledge Monitoring reads terminal reports from a directory in the file system rather than over XMPP. Thus, if terminal reports representing a network scenario are recorded, they can be replayed into Aesop.

4.4.4.2 Semantic Analysis

The Aesop Semantic Analysis algorithm described in Section 4.3.4.2, is implemented as a *SemanticAnalysis* instance, a specialization of an *AesopOperation PeriodicOperation* as shown in Figure 4-7 that starts as a new thread every 10 seconds. The operation analyses each new *ServiceExperienceSnapshot* individual in

turn, calculates its compliance with expectations, and stores that compliance value back into the *ServiceExperience Snapshot* individual.

The Semantic Analysis algorithm, shown in Figure 4-5, is implemented as shown in Figure 4-8. In step 1, a temporary reasoned model is created composed of the metadata, global knowledge, and the snapshot in question. A generic SPARQL [W3C, 2008] query, shown in Listing C-3 in in Appendix C, is run on the temporary model in step 2 to find new or updated service experience snapshots and their equivalent service expectations. In step 3, the service type of each service experience snapshot is used to look up a service specific query for each service experience snapshot found, into which the URIs of the service experience snapshot and service expectation individuals are passed as parameters. This query is run on the temporary model to determine the compliance of that particular service experience snapshot individual with that particular service expectation individual. The compliance of that service experience snapshot individual is then updated in the Knowledge Base in step 4.

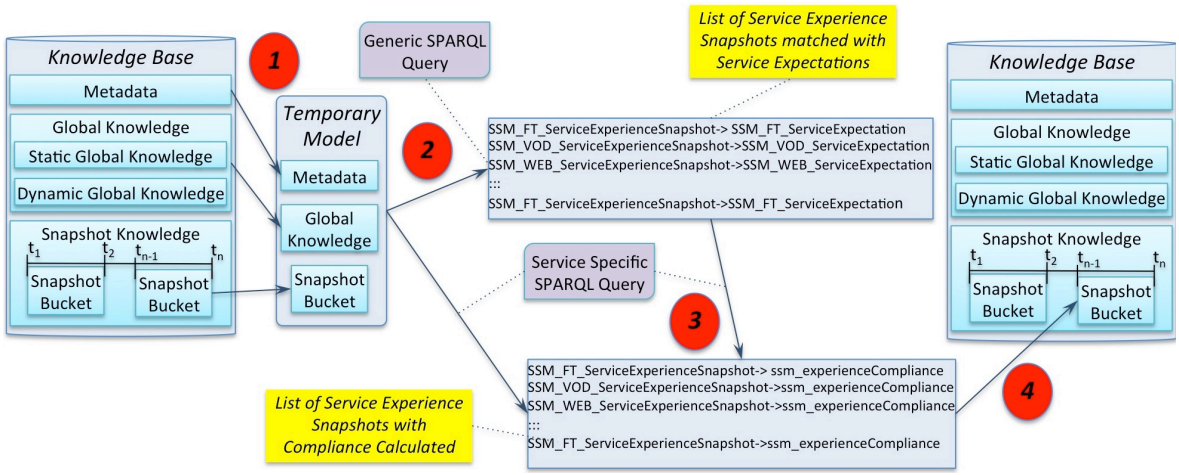


Figure 4-8: Semantic Analysis Implementation

The sequence of query execution in the scenario shown in Figure 4-8 is as follows:

a. The generic SPARQL query shown in Listing C-3 in in Appendix C is run to create the matched list of service experience snapshots and service expectations
b. The File Transfer service specific SPARQL query shown in Listing C-4 is run to calculate the compliance of the first file transfer snapshot.
c. The Video on Demand service specific SPARQL query shown in Listing C-5 is run to calculate the compliance of the first file transfer snapshot.
d. The Web Browsing service specific SPARQL query shown in Listing C-6 is run to calculate the compliance of the first file transfer snapshot.

e. The File Transfer service specific SPARQL query shown in Listing C-4 is run to calculate the compliance of the second file transfer snapshot.

4.4.4.3 Optimization Planning and Execution

The Optimization Planning and Execution algorithm described in Section 4.3.4.3, is implemented as an `OptimizationPlanning` instance, a specialization of an `Aesop Operation PeriodicOperation` as shown in Figure 4-7 that starts as a new thread every 10 seconds. The algorithm shown in Figure 4-6 runs two sub-activities, one to check and apply throttling actions and one to check and release optimization actions.

In the Check and Apply Throttling sub-activity, the Optimization Planning and Execution algorithm uses a lookback interval of 35 seconds. This ensures that four measurements for all end user sessions are considered before a throttling application action is applied so that transient degraded measurements do not trigger throttling applications. A longer lookback interval of 65 seconds is used in the Check and Release Throttling sub-activity, causing seven measurements for each end user session are considered. This is to ensure that throttling applications and throttling releases do not interfere and cause flip-flop of throttling actions on end user sessions.

The implementation of the Optimization Planning and Execution algorithm uses SPARQL for querying. Two example queries appear in Appendix C. The SPARQL query that checks the temporary model for new or terminated end user sessions is shown in Listing C-7 and the SPARQL query to check compliance of all sessions in a set of snapshot buckets with expectations is shown in Listing C-8.

When a session is selected for throttling, the throttling parameters that are to be applied or released on the session are read from the `SSM_Priority` individual associated with the user of the session. Other parameters such as the IP address of the terminal machine on which the end user session in question is running are also read from static global knowledge in the knowledge base. An `OptimizationExecution` implementation applies the optimization action recommended by `Optimization Planning` to the home gateway, as shown in Figure 4-7. The `OptimizationPlanning` operation calls the `OptimizationExecution` interface to apply the optimization. The `OptimizationExecution` implementation in the Aesop prototype invokes a script with the parameters passed by `OptimizationPlanning`. The script runs Linux traffic shaping commands [Netfilter Framework, 2012] on the home gateway to apply or release throttling as requested. It then updates the throttling status on the

SSM_ServiceSession individual and adds an *SSM_OptimizationSnapshot* individual in the Knowledge Base, recording the throttling action and its parameters.

4.4.4.4 Support Functions

The support functions of Aesop shown in Figure 4-1 are described below.

Northbound Interface

This component exposes public semantic and non-semantic interfaces for a user interface client or for other management systems using a machine interface such as Web Services or Java Remote Method Invocation (RMI) [Oracle, 2013]. A client or another management system can use this interface to read knowledge from the Aesop Knowledge Base or to configure individuals of the concepts shown in Table 4-1 in the Knowledge Base.

A Northbound Interface is implemented on the Aesop server as a Java RMI service. When a request is called on the service, a SPARQL query is invoked on the Knowledge Base to fetch the requested knowledge. A request can be scoped with a date; in which case only knowledge newer than the requested date is returned. The query result is parsed and converted into Java serializable objects, and returned to the caller. The Northbound Interface supports fetching and translation of instances of the following concepts: *SSM_Connection*, *SSM_ServiceSession* and all its sub-concepts, and *SSM_ServiceExperienceSnapshot* and all its sub-concepts. In this way, an external system using the Northbound Interface can get a full picture of the end user services being optimized by Aesop.

A prototype user interface was implemented as a Java thick client. It polls the Aesop Northbound RMI interface for updates periodically. It displays the current status of connections, end user sessions and session metrics in tabular form and plots the compliance level of end user sessions in graphical form. A snapshot of the user interface is shown in Figure K-1 in Appendix K.

A simple Command Line Interface (CLI) was developed for testing and debugging purposes. This CLI supports four commands. The *load* command loads an OWL file into the Knowledge Base. The *dump* command dumps the entire contents of the Knowledge Base into an OWL file. The *query* command runs a named SPARQL query on the Knowledge Base and displays the result on the terminal. The *exit* command causes Aesop to stop.

Knowledge Persistence

Knowledge Persistence is a component used to load and store knowledge from and to persistent storage. It is used by Knowledge Base maintenance to load static knowledge from persistent storage at system start up. It stores knowledge that has been identified as being useful for off-line reporting and analysis to persistent storage periodically.

Knowledge persistence provides an interface to other components that allow RDF documents to be streamed to and from persistent storage. Two mechanisms for knowledge persistence are implemented in Aesop. The first mechanism persists knowledge to a Sesame semantic store [Aduna, 2013]. The second mechanism persists knowledge to a PostgreSQL database [The PostgreSQL Group, 2013].

The Sesame semantic store runs as a Java servlet in Apache Tomcat [Apache, 2013d] and is deployed with a PostgreSQL back end. A Java API (Application Programming Interface) is provided for Sesame that allows applications to read from and write to the semantic store. Knowledge Persistence wraps this API to allow components to write RDF to the semantic store. This persistence mechanism was used in Experiment 1, described in Section 6.3.

The PostgreSQL persistence mechanism was implemented so that knowledge created during execution by Aesop could be saved to a relational database because relational databases offer a richer set of tools for post processing than semantic stores currently do. This persistence mechanism was implemented as a standalone Java program which, when started, reads the knowledge published on the Northbound Interface every 10 seconds and stores that knowledge into tables in a PostgreSQL database. This persistence mechanism was used in Experiments 2 and 3 described in Sections 6.4 and 6.5 respectively.

Knowledge Base Maintenance

Because the Knowledge Base is held in memory, it is imperative to have active maintenance of the Knowledge Base to ensure memory usage is optimized. This component manages the Knowledge Base, performing initiation and housekeeping tasks.

At start up, Knowledge Base Maintenance initializes the Metadata and Static Global Knowledge models using knowledge read from two OWL files on the file system.

Rules are loaded into the Metadata model from a third OWL file. It then initializes all other components and starts the Java timers to trigger execution of periodic operations.

To clean the Knowledge Base, the Knowledge Base Maintenance runs every minute and deletes all snapshots that are older than the snapshot bucket retention interval, which is set at 10 minutes. Although knowledge base maintenance should also clean dynamic global knowledge, removing service session and network connection knowledge that is no longer referenced by knowledge in snapshots, this function was not implemented in the prototype implementation of Aesop. The Knowledge Bus interface methods shown in Table D-1 in Appendix D are used to perform Knowledge Base Maintenance.

4.5 Summary

Aesop is a system designed to optimize end user service delivery in any networking domain in which the experience and context of end user service delivery can be monitored and in which the network resources allocated to specific end user service sessions can be controlled. This meets the first requirement specified in Section 4.2.

Aesop is an autonomic system for optimization of end user service delivery. Aesop's Knowledge Base is structured to allow semantic reasoning, rules, and queries to operate on scoped sections of the knowledge in the system. This structure is used by Aesop's autonomic operations to minimize the size of models used in semantic operations thus maximizing performance. In addition, all loading, analysis, and optimization operations are threaded, thus enabling simultaneous execution. Aesop is deigned to execute its autonomic loop in a time period of not more than 40 seconds, thus meeting the second requirement specified in Section 4.2.

Aesop uses a central Knowledge Base in which all knowledge is modelled using the EUSAO ontology and its referenced ontologies, vocabularies and models. Individual components do not hold or use any local knowledge. Thus, Aesop is designed to comply with the third requirement in Section 4.2.

The domain specific logic in Aesop is implemented using semantic techniques, without any domain specific source code. Because Aesop uses the EUSAO ontology, which is structured to handle new end user services and networking domains, Aesop's Knowledge Base can handle new services and networking domains by

introducing sub-concepts for new services and networking domains on the EUSAO ontology. Knowledge Monitoring can monitor end user service experience and context from any domain. Because knowledge is lifted using mappings specified as XSL transformations, new transformations can be deployed to lift information from any textual format. Semantic Analysis can support new end user services by installing a SPARQL query to analyse end user service experience for that service type and determine its compliance with expectations. The SPARQL queries used by Optimization Planning are aware of just the concepts of *service*, *service session*, and *compliance*, and do not consider specific end user services or networking domains. In Optimization Execution, throttling actions are executed by invoking scripts. Those scripts can be modified to execute throttling actions using the mechanism supported by a given networking domain. Thus, Aesop complies with the fourth requirement specified in Section 4.2.

Aesop has been designed and implemented using readily available semantic components and can run on any system on which a standard Java virtual machine can run. This design and implementation complies with the fifth requirement in Section 4.2.

The evaluation of the Aesop prototype implementation is described in Chapter 6. Chapter 7 describes a case study that was undertaken to show how the Aesop design could be applied in the Mobile Broadband Access networking domain.

Chapter 5
The SECCO Framework

5.1 Introduction

In order to determine the degree to which Aesop answers the research question, it is necessary to evaluate Aesop by carrying out experiments in a test network. For experimentation, end user service sessions must be run in a controlled and repeatable manner on the test network, the end user service experience and context of those end user sessions must be measured and recorded as close to the end user as possible, and those measurements must be collated and delivered to Aesop.

The closest point to the user for measuring end user service delivery and context is in service delivery clients such as web browsers and media players. While service delivery clients support automation of end user sessions and expose mechanisms for retrieval of end user service experience and context, the discussion in Section 2.3.1.1 shows that there is no extant generic framework for automation of end user sessions, for collection of service experience and context measurements, or for collation and delivery of those measurements to a management system.

When using an end user service such as web browsing or video on demand, users typically browse a number of web sites and may watch a number of videos. Tools such as the iMacros plugin [iOpus, 2013] allow web browsing session to be automated and most media players such as VLC [VideoLAN, 2013] support playlists, where a sequence of videos can be specified for sequential playout. Using such tools and features allows scenarios for end user sessions of a particular service type such as those described in Appendix I to be automated.

End user service delivery clients also expose mechanisms for determining the service experience and context of end user service sessions they are running. VLC can be configured to output detailed logs of media sessions it is playing, and Firefox [Mozilla, 2013b] plugins such as Firebug [Mozilla, 2013a] and HttpFox [Martin Theimer, 2013] report detailed metrics on web browsing sessions. These mechanisms can be harnessed to report end user service experience and context.

The discussion in Section 2.3.1.1 observes that terminal reporting is emerging as a delivery mechanism for reporting metrics. However, a terminal reporting mechanism that can collect end user service experience and context measurements from service delivery clients and forward those measurements as terminal reports has not been found to exist by the author.

This chapter describes SECCO, a Service Experience and Context COllection framework. The goal of the framework is to make it easy for users to set up, run, and monitor end user service sessions. To a user of SECCO, specification, execution, and monitoring of different types of end user services is carried out in a uniform manner. The motivation for SECCO is that it shields the user from the intricacies of mechanisms and interfaces provided by different end user service delivery clients. All end user service sessions are handled by SECCO, without the need for using the time honoured method of composing a plethora of custom scripts.

SECCO is designed to be a highly customizable framework, so that addition of support for new service delivery clients is straightforward. It provides a customizable mechanism that allows end-user service sessions to be run in a consistent and controlled manner. It provides support for recording end user service experience and context at the service delivery client. It collects measurements and compiles terminal reports for all end user sessions on a host and forwards them to Aesop for processing.

In summary, Section 5.2 lists the requirements on SECCO. The design of SECCO is described in Section 5.3. In Section 5.4, the implementation of SECCO used to evaluate Aesop is explained. Section 5.5 provides a summary of the chapter.

5.2 Requirements

This section describes and motivates the requirements on Aesop as a system for managing end user service delivery.

1. **Support automatic execution of arbitrary end user service sessions.** SECCO must provide a mechanism that allows specification of end user service session scenarios for each service type using the means supported by the service delivery client for automation of sessions of that service type. The mechanism must isolate the specification of a scenario from the configuration parameters for a particular test terminal running that scenario, thus avoiding the need to write a different script for each service session scenario on each test terminal. This requirement is

derived from the research question in Section 1.2 because it enables automated execution of end user service sessions for the evaluation of Aesop.

2. **Record end user service experience and context measurements for service types in a generic manner.** In order to provide a common method for recording end user service experience and context, SECCO must provide a framework that wraps and isolates the mechanism exposed by each particular service delivery client for reporting end user service experience and context. That common framework must be possible to extend to support new service types by providing an adapter for the mechanism exposed by the service delivery client of that new service type. This requirement is derived from the research question in Section 1.2 because it enables recording of the service experience and context of end user service sessions.
3. **Collate and deliver end user service experience and context measurements to a management system in near real time.** End user service delivery is inherently distributed. The SECCO framework must collect the measurements recorded on each end service delivery client running on a terminal, collate these measurements, and send them to Aesop in near real time. This requirement is derived from the research question in Section 1.2 because it enables delivery of end user service experience and context to Aesop in near real time.

5.3 Design

A conceptual overview of SECCO is shown in Figure 5-1. SECCO runs on each terminal on which end user service sessions are to be executed. SECCO is used to launch each service delivery client with a specification of the end user session scenario to run. SECCO records the service experience and context information and measurements of end user service sessions. SECCO collects the information and measurements from all service delivery clients running on a terminal and collates them into a single report, which it delivers to the management system.

The UML deployment diagram in Figure 5-2 shows the run time entities that exist when SECCO is deployed on a terminal. A single *SECCOReporter* process on each terminal sends reports to a *ManagementSystem* entity on a management host. Each *SECCOReporter* process interacts with multiple *SECCOClient* processes on the terminal, each one of which interacts with a single *ServiceDeliveryClient* such as a web browser or media player that is running end user service sessions.

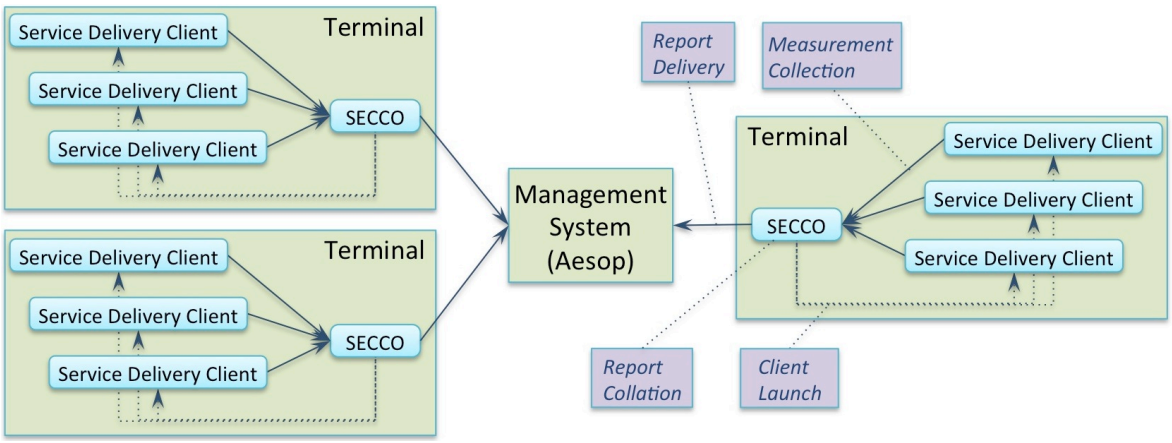


Figure 5-1: Conceptual Overview of SECCO

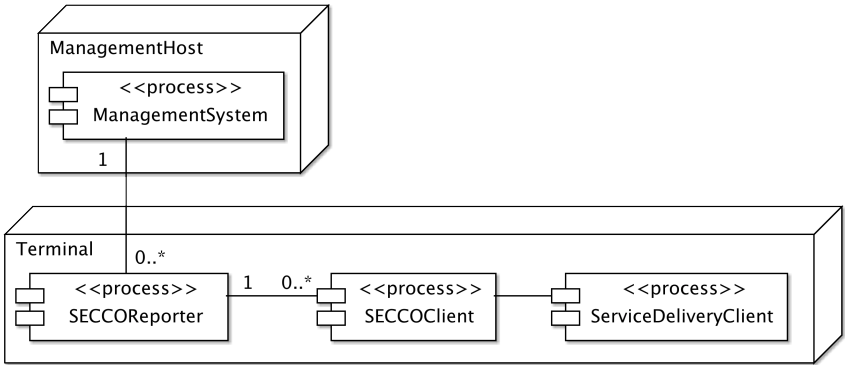


Figure 5-2: SECCO UML Deployment Diagram

5.3.1 The *SECCOReporter*

The UML class diagram of a *SECCOReporter* is shown in Figure 5-3. The *SECCO Reporter* periodically calls a *Collator* to collate the latest measurements into a report. The *collator* passes that report to the *Dispatcher*, which sends it to the *ManagementSystem*. The *Collator* is composed of a number of *Collector* instances, each of which contributes sections to a report for a period. The *LocationInformationCollector* contributes the current location of the terminal; the *TerminalInformationCollector* contributes information such as the terminal make and model, and metrics such as the current processor load and memory usage on the terminal. The *ConnectionExperienceCollector* contributes information and metrics on each active connection on the terminal that is carrying traffic and the *ServiceExperienceCollector* contributes information and metrics on the end user services that are running on the terminal.

The *getExperience()* method on *Collector* instances returns a generic type such as a string or an XML element. The *Collator* calls the *getExperience()* method on each *Collector* instance in turn and concatenates the returned values to compose the

report. The *Collator* is, therefore, unaware of the internal design and implementation details of each *Collector* specialization.

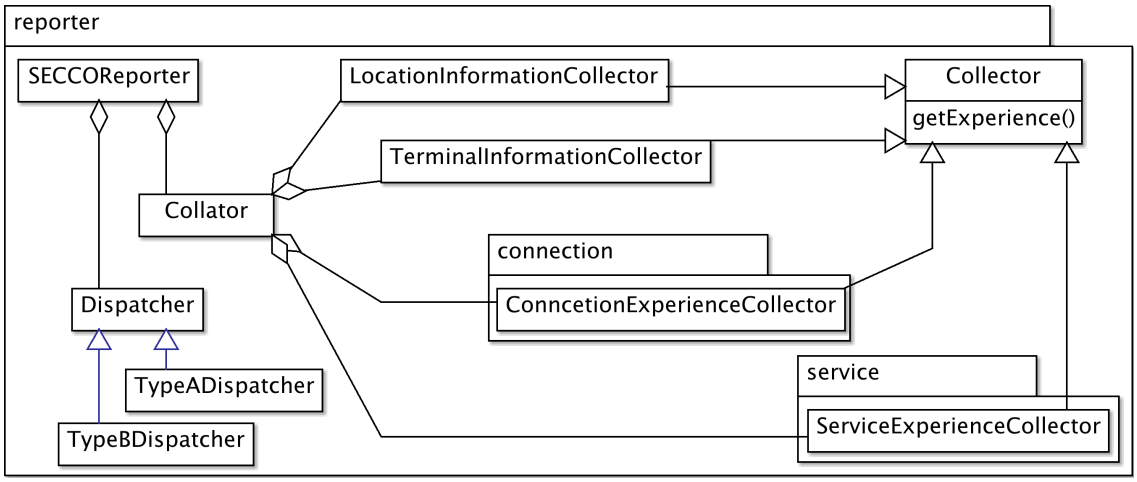


Figure 5-3: *SECCOReporter* UML Class Diagram

The *connection* package UML class diagram is shown in Figure 5-4. The *ConnectionExperienceCollector* instance collects information and metrics on each active connection, modelled as a *Connection* instance. The *Address* instance contributes addressing information such as the IP address and host name of the connection. The *ConnectionInterface* instance contributes interface information such as the MAC address and interface name. The *ConnectionMetrics* instance contributes metrics on the connection such as bytes and packets transferred which may be read from the interface hardware using a mechanism such as SNMP [IETF, 2002]. The *LatencyMeasurement* instance contributes metrics on the round trip time being experienced on the connection, which may be measured using an approach such as executing a periodic *ping* command to a predefined IP address.

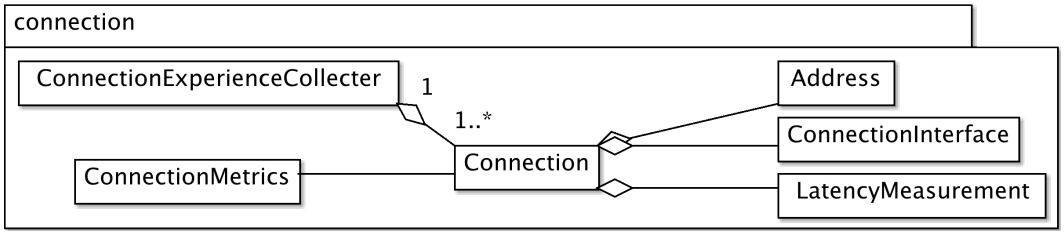


Figure 5-4: *connection* Package UML Class Diagram

The UML class diagram in Figure 5-5 shows the classes in the *service* package of a *SECCOReporter* and the *ServiceExperienceClient* class in a *SECCOClient* component. As shown in Figure 5-2, a *ServiceExperienceServer* instance in a *SECCOReporter* and a *ServiceExperienceClient* instance in a *SECCOClient* are in separate processes. When a *SECCOClient* starts, its *ServiceExperienceClient*

instance registers with the *ServiceExperienceServer* instance on the *SECCOReporter* running on the terminal using an inter-process communication mechanism. On shutdown of a *SECCOClient*, the *ServiceExperienceClient* instance removes its registration from the *ServiceExperienceServer* instance.

When a *Collator* calls the *getExperience()* method on a *ServiceExperience Collector* instance, it uses its *ServiceExperienceServer* instance to collect information and metrics on each active session on each *SECCOClient* running on the terminal. It uses inter-process communication to call the *getServiceExperience()* method on each *ServiceExperienceClient* instance that has registered.

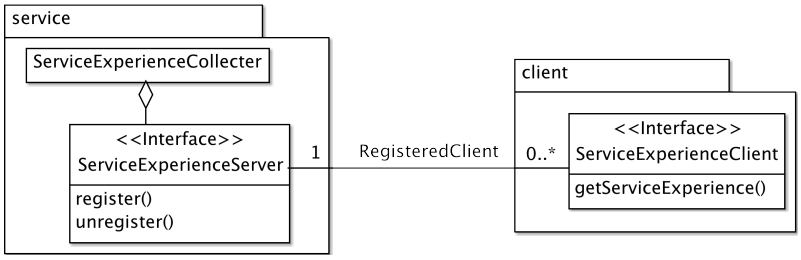


Figure 5-5: *service* Package UML Class Diagram

The *SECCOReporter* is designed to be agnostic to the mechanisms used to create sections of the reports it collates. The use of specialized instances of the *Collector* class ensures that the mechanisms to read information such as connection and session metrics are encapsulated. This *SECCOReporter* design can be applied to collect any type of end user service experience and context information. Dispatching of reports to a management system is also agnostic to the mechanism used for report transfer; the *Dispatcher* class can also be specialized to encapsulate different methods for transmission of reports.

5.3.2 The *SECCOClient*

The UML class diagram of a *SECCOClient* is shown in Figure 5-6. A *SECCOClient* is composed of three packages. The *client* package contains the main classes for controlling execution of an end user service session. The classes in the *session* package collect service experience and context on ongoing sessions. The *service deliveryclient* package encapsulates service delivery clients execute end user service sessions. In Figure 5-6, classes with class names in italics are abstract classes; the *ServiceX* and *ServiceY* specializations of each abstract class represent concrete classes for two arbitrary services X and Y. A *SECCOClient* for service X instantiates

the *ServiceX* specializations of each abstract class, whereas a *SECCOClient* for service Y instantiates the *ServiceY* specializations.

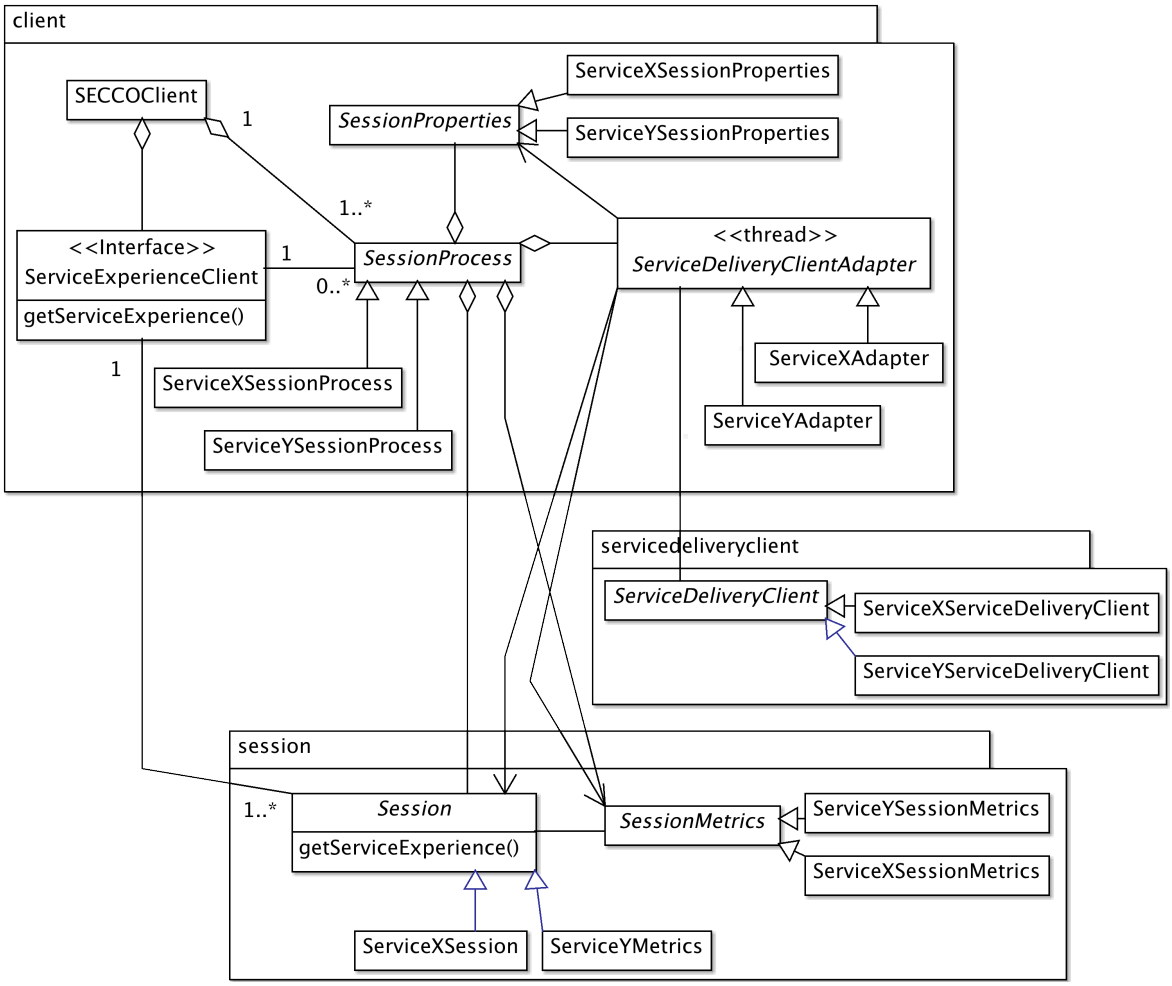


Figure 5-6: *SECCOClient* Class Diagram

When a *SECCOClient* starts, it creates a *ServiceExperienceClient* instance, which registers with the *SECCOReporter* running on the terminal. The *ServiceExperience Client* instance is called by the *SECCOReporter* to read end user service experience and context, as explained in Section 5.3.1.

The *SECCOClient* reads its configuration information to determine the number of sessions that should be started and to set the properties for those sessions. It creates a *SessionProcess* instance for each session and passes a *SessionProperties* instance to each *SessionProcess* instance with the configuration parameters for that session. The *SessionProcess* instance creates a *Session* instance to hold information on the session such as the user running the session and a *SessionMetrics* instance to hold metrics on the session such as the duration of the session and the number of bytes transferred in the session. The instance then creates a *ServiceDeliveryClient*

Adapter instance, passing a reference to the *SessionParameters*, *Session* and *SessionMetrics* individuals for the session.

The *ServiceDeliveryClientAdapter* specialized instance for a given service type starts a service delivery client such as a web browser or a media player, passing the session parameters specified in the *SessionProperties* instance using the method supported by the service delivery client in question for session specification. It then runs a thread that monitors the service delivery client, collecting end user service experience and context for the session by using whatever mechanism is exposed in the service delivery client. Session information such as the user running the session or the interface being used by the session is recorded in the *Session* instance. Session metrics such as the duration of the session or the number of bytes transferred by the session are recorded in the *SessionMetrics* instance.

When the *SECCOReporter* calls the *getServiceExperience()* method on the *ServiceExperienceClient* instance using inter-process communication, the *Service ExperienceClient* instance iterates over the list of *Session* instances in the *SECCOClient*, and calls the *getServiceExperience()* method on each one. Each *Session* instance returns its session information together with the metrics from its associated *SessionMetrics* instance. The session information and metrics from each session is concatenated in the *ServiceExperienceClient* instance and returned to the *SECCOReporter*.

The design of the *SECCOClient* is generic; it can be applied to launch, control and monitor sessions in service delivery clients of any type. Launch and configuration of service delivery clients is encapsulated in an adapter, as is monitoring and recording of end user service delivery session information and metrics. The *SECCOClient* reports session information and metrics to a *SECCOReporter* using an inter-process communication mechanism that is agnostic of the service types of the end user sessions being monitored.

5.4 Implementation

The SECCO implementation developed for use in the evaluation of Aesop is described in this section. SECCO was implemented largely in Java, together with some scripting support used for launching of *SECCOReporter* instances, *SECCOClient* instances, and end user service delivery clients.

In a SECCO deployment, the processes and instances shown in Figure 5-2 are instantiated as shown in Figure 5-7. The *SECCOReporter* on each terminal runs as a JVM (Java Virtual Machine). It uses Java RMI (Remote Method Invocation) [Oracle, 2013] to communicate with *SECCOClient* instances, each of which runs in a JVM. The *SECCOReporter* sends end user service experience and context to Aesop as terminal reports in GSQR [Fallon & Huang, 2011] format over XMPP [IETF, 2011]. GSQR was chosen because it provides a structure for terminal reports that captures the connection and session information on terminals and because the XML schemas for the protocol were available from previous work by the author of this thesis. XMPP was used as a transmission mechanism because it supports text-based messages such as XML, it provides the underlying addressing, routing, and transmission support for message delivery, and it was the messaging protocol of choice in the FAME project [Etzioni et al., 2011].

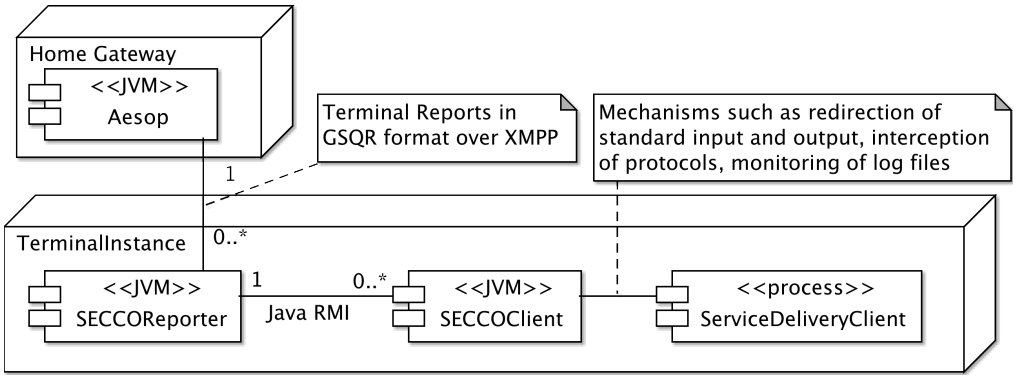


Figure 5-7: SECCO Instance UML Deployment Diagram

The interface used between *SECCOClient* instances and service delivery clients is dependent on the particular service type being delivered and the service delivery client being used. Implementations of this interface are discussed in Section 5.4.2.

5.4.1 The *SECCOReporter*

When it starts, a *SECCOReporter* instance reads a configuration file that defines its properties. Properties such as the interval between each report, the address of the XMPP server to which to send terminal reports, and the RMI port to open for connections from *SECCOClient* instances are specified.

Transmission of terminal reports over XMPP is implemented in an *XMPPSender* class, an extension of a *Dispatcher*, as shown in Figure 5-3. The *XMPPSender* instance opens a connection to the XMPP server specified in the configuration properties of

the *SECCOReporter* and sends terminal reports to it as they are forwarded by the *Collator* instance.

The *Collator* instance shown in Figure 5-3 creates a report using the reporting interval specified as a configuration property. The *getExeprience()* method of the *Collector* base class returns an XML element or a list of XML elements. Each sub-class of *Collector* returns XML elements that form a part of a terminal report in GSQR format. The *Collator* instance collates a terminal report by calling its instances of *Collector* specializations in the correct order to get all the elements of a GSQR terminal report and brackets those elements in a header and footer.

The *connection* package, shown in Figure 5-4, is used to determine the information and metrics on interfaces on a terminal. In the SECCO implementation, the identities of the interfaces on which information and metrics are to be collected is configured in the *SECCOReporter* configuration file. The *ConnectionExperience Collector* creates a *Connection* instance for each of those interfaces. The *Connection* instance instantiates a *ConnectionInterface* instance and an *Address* instance. The *ConnectionInterface* instance uses SNMP to read interface information such as the MAC address and description; the *Address* instance uses SNMP to read address information for the interface such as IP address and host name. All SNMP requests are executed using the SNMP4J library [SNMP4J.org, 2013].

The *Connection* instance creates a *ConnectionMetrics* instance to hold metrics on the connection such as the number of packets and bytes transferred and the latency of the connection. It also creates a *LatencyMeasurement* instance, which starts a thread that periodically pings an address specified in the *SECCOReporter* configuration file. The latency value returned by the *Latency Measurement* instance is stored in the *ConnectionMetrics* instance.

When the *getExperience()* method is called by the *Collator* instance on the *ConnectionExperienceCollector* instance, it calls the *Connection* instance for each interface. That triggers a SNMP request, which reads the most recent metrics on the interface into the *SessionMetrics* instance. Those metrics, together with the latency measurement returned by the *LatencyMeasurement* instance, are returned as an XML element to the *ConnectionExperienceCollector* instance via the *Connection* instance. The *ConnectionExperienceCollector* instance returns a list of XML elements to the *Collator* instance, one XML element for each connection.

In this implementation of SECCO, the *LocationInformationCollector* instance returns a location for the terminal configured in the SECCOReporter configuration file. In another implementation, a *LocationInformationCollector* instance for a GPS device or mobile network positioning could be used to provide dynamic location information. Information such as the terminal model and serial number returned by the *TerminalInformationCollector* instance is also read from the configuration file, but could also be determined programmatically in another implementation.

5.4.2 The *SECCOClient*

The *SECCOClient* implementation supports three types of end user services: file transfer, video on demand, and web browsing. The type of service to be executed is specified to a *SECCOClient* instance when it starts together with a configuration file that defines the properties of the service session to be executed. Properties such as the user name under which to execute the service session and the specification of the service scenario to run are specified.

When it starts, a *SECCOClient* instantiates a *ServiceExperienceClient* instance to connect to the *SECCOReporter* instance running on the terminal using the approach described in Section 5.3.2. It also instantiates a *SessionProcess* instance for the type of service to be executed, and passes the input parameters as a *SessionProperties* instance to the *SessionProcess* instance. Although the design of a *SECCOClient* supports multiple *SessionProcess* instances in a *SECCOClient*, in this SECCO implementation each *SECCOClient* has a single *SessionProcess* instance. The implementation of *SessionProcess* and associated specialized classes for the file transfer, video on demand, and web browsing services are described in Sections 5.4.2.1, 5.4.2.2, and 5.4.2.3 respectively.

When the *SECCOReporter* instance calls the *getServiceExperience()* method on a *SECCOClient*, the *ServiceExeprienceClient* instance reads the session information and metrics for each session as described in in Section 5.3.2. The *Session* and *SessionMetrics* specializations for each service class override the *getServiceExperience()* method, so their internal implementation for recording metrics is transparent to the service experience and context collection mechanism.

5.4.2.1 File Transfer

File transfer sessions are specified as a list of *get* and *put* operations specified in a configuration file. For each FTP operation, a start delay, the local file location, and a

remote host, username, password, and file location is read from the configuration file. The number of times the list of file transfer operations is executed can also be specified in the configuration file.

The File Transfer `SessionProcess` instance implements file transfer sessions over FTP [IETF, 1985] using the Apache Commons Net library [Apache, 2013a]. When a file transfer session starts, a new `FileTransferSession` and `FileTransferMetrics` object is instantiated. The session input parameters are recorded in the `FileTransferSession` instance. The Apache Commons Net library provides a listening mechanism on FTP sessions, which is used to update the `FileTransferMetrics` object with the number of bytes being uploaded and downloaded as the session progresses.

5.4.2.2 Video on Demand

SECCO supports RTSP [IETF, 1998a] video on demand sessions executed in the VLC media player [VideoLAN, 2013]. Video on demand sessions over RTSP were chosen because control messages of this protocol are relayed in text format and are easy to intercept and because the protocol is supported by the VLC media player. VLC was chosen because it is the most popular open source video player and its service experience and context logging mechanisms are well documented. Another reason for selecting VLC with RTSP is that VLC can be run in server mode and can serve videos over RTSP. VLC was used in this configuration as a video server in the experimental test bed, as explained in Section 6.2.5.

The specification of the sequence of video on demand sessions to execute is specified in a file as a list of RTSP URLs that specify the server, port, and location of the video to execute. As with file transfer sessions, the number of times the list of sequence of video on demand sessions is executed can also be specified in the file.

The Video on Demand `SessionProcess` instance uses an adapter to start the VLC media player for each RTSP session specified in the configuration file. When a video on demand session starts, a new `VideoOnDemandSession` and `VideoOnDemandMetrics` object is instantiated. The session input parameters are recorded in the `VideoOn DemandSession` instance. VLC is started with its network caching buffer size set at 1 second, which emulates real time streaming by minimizing buffering in the client.

The adapter monitors and parses the VLC media player log file during session execution to get the metrics of each video on demand session. VLC logs all events that

degrade playout quality. Each VLC event gives the source of the event, the event category, and whether the event is an error, a warning, or for information. The adapter counts the number of video freezes, errors and warnings logged by VLC and stores those values to the *VideoOnDemandMetrics* object.

5.4.2.3 Web Browsing

SECCO uses the Firefox web browser [Mozilla, 2013b] as a service delivery client for web browsing sessions because it is the most commonly used open source browser it has a rich ecosystem of open source plugins that are available to adapt.

Firefox does not expose a direct mechanism for recording end user service experience and context. The HttpFox Firefox plugin [Martin Theimer, 2013], an open source plugin written in JavaScript, records, analyses and presents the performance of a web page load graphically, showing the parallelism in the loading and also the degree of caching used. This plugin was enhanced to write its page loading analysis to file, providing a source of service experience and context information on end user sessions running in Firefox that can be recorded by an adapter.

The iMacros Firefox plugin [iOpus, 2013] allows Firefox to be scripted and to run under the control of a macro. A sequence of web pages can be specified for loading, delays can also be introduced between web page loads, and loops can be specified. Plugins like iMacros are commonly used for web scraping.

In SECCO, Firefox is deployed as shown in Figure 5-8. The specification of web browsing sessions to execute is specified as an iMacros macro. The macro specifies the web pages to load and the delay between each page load. The number of times the sequence of web pages is loaded can also be specified in the macro.

The Web Browsing *SessionProcess* instance uses an adapter to run Firefox with the web browsing session specified in the iMacros macro. When a session starts, a new *WebBrowsingSession* and *WebBrowsingMetrics* object is instantiated. The session input parameters are recorded in the *WebBrowsingSession* instance.

The adapter parses HttpFox log file to get web browsing session metrics as shown in Figure 5-8. The log contains load times for HTTP requests, volumes of data downloaded and uploaded by each request, and details on the status of completed requests. The adapter summarises request information for each session, calculating the number of requests, the average and standard deviation of request load time and

request size, and counts the number of requests that finished, had errors and were aborted. This summary information is stored to the `WebBrowsingMetrics` object.

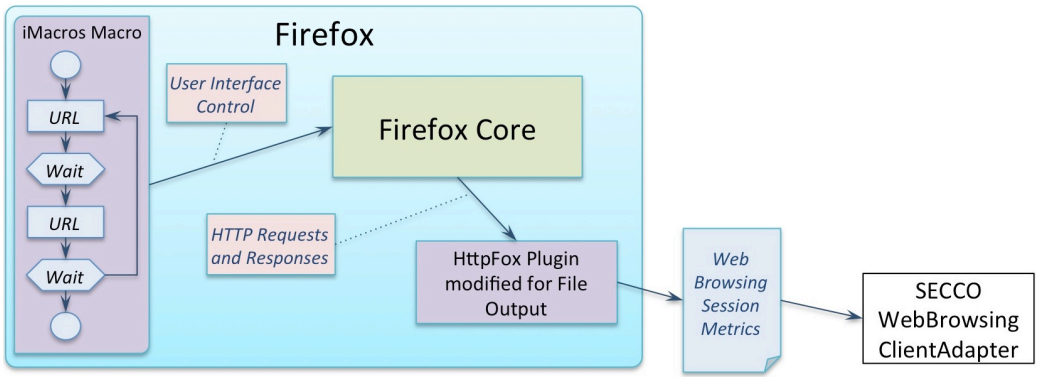


Figure 5-8: Firefox Deployment for Web Browsing end User Session Recording

There is no inherent session concept in web browsing. The adapter implements a timeout mechanism that marks a web browsing session as terminated if no activity is detected on a web browsing session for a configurable time period. Subsequently, if new activity is detected, a new web browsing session is deemed to have started.

5.5 Summary

SECCO can be used to automate execution of end user service sessions of arbitrary types. It provides a generic mechanism for launch and automation that can be adapted to use the launch and automation methods supported by end user service delivery clients. This meets the first requirement specified in Section 5.2.

SECCO provides a framework for recording session information and metrics that can be extended to record any type of end user session. Extensions of the framework encapsulate mechanisms exposed by the service delivery client for a particular service type. Recording of end user service experience and context is opaque to the information and metric collection mechanism in SECCO. Thus, SECCO complies with the second requirement specified in Section 5.2.

The collation mechanism in SECCO collects information and metrics in near real time. Collection of terminal, location, connection, and session information and metrics is supported for all connections and sessions running on a terminal. The delivery mechanism for forwarding reports is also encapsulated and can be extended to support various delivery mechanisms. Therefore, SECCO complies with the third requirement specified in Section 5.2.

Chapter 6
Experimentation

6.1 Introduction

In previous chapters, the artefacts that comprise the proposed approach in support of answering the research question have been described. Chapter 3 describes the EUSAO ontology, which holds the end user service expectation, experience, and context knowledge in a partitioned model structure. The Aesop autonomic end user service management system described in Chapter 4 monitors and analyses end user service sessions and plans and executes optimization traffic controls on those service sessions semantically with a Knowledge Base that uses the EUSAO ontology. The SECCO framework, explained in Chapter 5, is used to support the experimental scenarios and to gather and transport terminal reports to Aesop.

The evaluation of the proposed approach was carried out as a series of three experiments and a case study. This chapter describes those experiments. The case study, which examines the applicability of the approach in the mobile broadband access domain, was undertaken to show that the proposed approach was applicable in a second networking domain and is documented in Chapter 7.

The EUSAO ontology was evaluated through the evaluation of the Aesop approach (where it is a central part of the approach) and through the case study. The goal of the experimentation was thus to evaluate the proposed Aesop approach in the experimental domain, specifically addressing the *"to what extent…"* and *"…in near real time"* facets of the research question. To do this, it was decided to have a series of experiments, progressively evaluating the functionality of the implemented Aesop system. Starting with focus on the Monitoring functionality only, then on Monitoring and Analysis functionality, and finally the full functionality of Monitoring, Analysis, Planning, and Execution.

Each experiment assessed one or more of the four properties of the Aesop approach described in Table 1-1: *effectiveness*, *consistency*, *accuracy*, and *performance*. As explained in Section 1.3.3, those four properties above were selected

to address the *"to what extent"* and *"in near real time"* aspects of the research question. *Effectiveness* and *consistency* are measured on the entire system and measure to what extent the system as a whole optimizes end user service session delivery. *Accuracy* and *performance* are measured separately on each of the MAPE functions, assessing how well each function works and the amount of time taken for each function to execute. Table 6-1 gives a synopsis of the context for each experiment that was performed. For example, in the table it shows that Experiment 2 relates to research objective 3 in evaluating the accuracy and performance of the monitoring and analysis MAPE functions of Aesop in a test bed environment using Scenarios 2 and 3.

The domain, scenarios, and experimental methodology were common for all experiments to allow comparison of results from each experiment. Experiment 1 was executed in a simulated environment because, in addition to its primary goal of evaluating the performance of semantic monitoring using the Aesop approach, it was used to get an initial indication of the applicability of the approach before carrying out further research and experimentation on test beds. Experiments 2 and 3 were executed in a full Home Area Network test bed, with Aesop running on the Home Gateway and SECCO being used to control end user sessions, collecting end user service experience and context and for terminal reporting.

Table 6-1: Synopsis of Experimentation

Experiment Number	Research Objective Addressed	MAPE Functions of Aesop Evaluated	Evaluation Properties Assessed	Experimental Environment	Scenario(s) Used
Experiment 1	Objective 3	Monitoring	Performance	Simulation	Scenario 1
Experiment 2	Objective 3	Monitoring Analysis	Accuracy Performance	Test Bed	Scenario 2 Scenario 3
Experiment 3	Objective 4	Monitoring Analysis Planning Execution	Effectiveness Consistency Accuracy Performance	Test Bed	Scenario 2 Scenario 3

This chapter is structured as follows. Section 6.2 describes the framework for experimentation, motivating the experimental domain, explaining the two experimental scenarios, describing how timing was handled in experimentation, explaining the simulation and test bed environments, and laying out the experimental methodology. Section 6.3, Section 6.4, and Section 6.5 describe Experiments 1, 2 and 3

respectively. Section 6.6 documents some conclusions drawn from insights gained during experimentation.

6.2 The Experimental Framework

In order to ensure that experimentation was realistic, relevant, and measurable, an experimental framework, shown in Figure 6-1, was established which specified the experimental networking domain, end user services and scenarios. The framework also specified the structure of the simulation and test environments to ensure experimentation was conducted in a realistic setting where external influences were minimized and controlled, as well as the timings and run counts to use for experiments in order to gather sufficient measurements for analysis. Finally, the framework established an experimental methodology to ensure that experimental execution was rigorous, traceable and repeatable. Each subsection describes each facet of that experimental framework. The experimental framework was based on software testing and configuration and change management workflows [Kruchten, 2000] from the IBM Rational Unified Process (RUP) [IBM, 2012a] software development process framework.

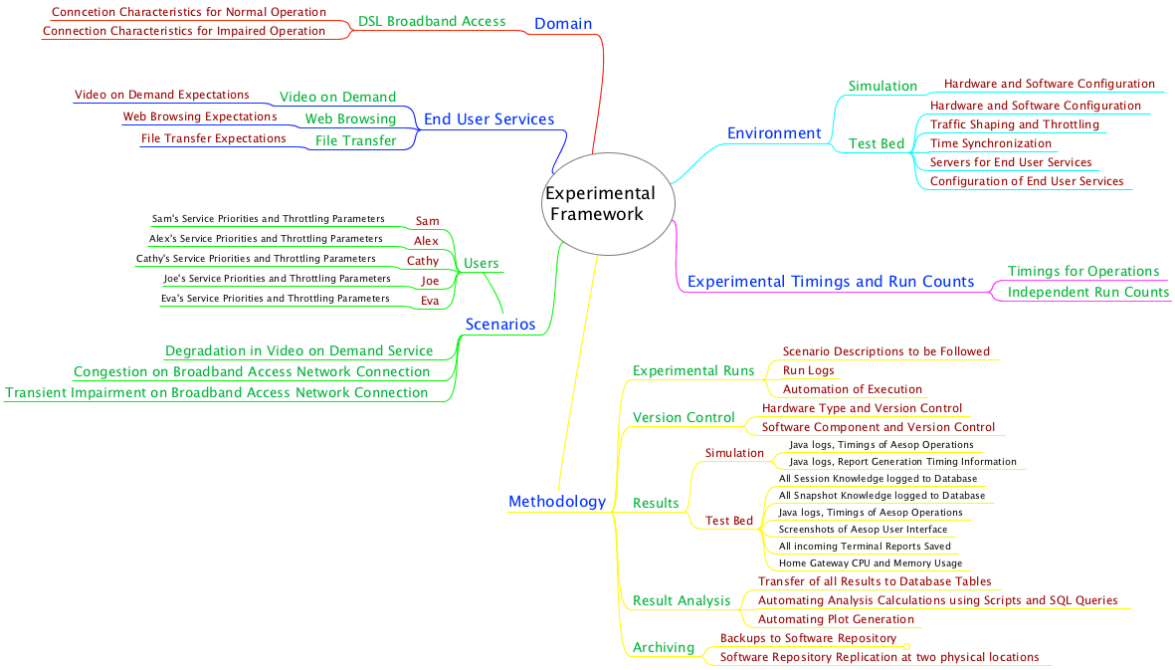


Figure 6-1: Facets of the Experimental Framework

The diagram in Figure 6-1 is a mind map [Buzan & Buzan, 2006]. Mind maps are a technique for visualizing hierarchical relationships between concepts. In the case of the environment of the experimental framework, Figure 6-1 shows that there is a simulation environment and a test bed environment. Both the simulation and test bed

environments have hardware and software configuration; the test environment has extra facets such as traffic shaping and throttling and servers for end user services.

6.2.1 Domain

The domain for experimentation that was selected is a Home Area Network in a FAME [Etzioni et al., 2011] HAN federation with a single connection to the Internet over a DSL broadband access network. This domain is discussed in detail in Section A.4. It was chosen as being realistic and relevant because such access connections are heavily used [OECD, 2011a], the characteristics and limitations of end user service delivery in HANs are well understood [OFCOM, 2012] [Sundaresan et al., 2011] [Dobrian et al., 2011] and because it is feasible and inexpensive to set up a realistic HAN test bed. End user service sessions run on terminals connected to the HAN. The terminals use terminal reporting to report their service experience and context to an Aesop system running on the Home Gateway. The Aesop system analyses the services running on the terminals and optimizes the use of the broadband access link resources to maximize the compliance of the set of running service sessions with service expectations.

A connection with an advertised speed of 20 Mbit/s is used in experimentation because that advertised speed is a popular choice with subscribers [OECD, 2011b]. The IP packet transfer and availability performance parameters throughput, packet loss, delay (Round Trip Time), and jitter (Packet Delay Variation) described by the ITU in [ITU-T, 2011] are controlled. Typical parameter values encountered on DSL connections advertised as being 20 Mbit/s were selected for the broadband access connection used in experimentation [OFCOM, 2012]. Those values are detailed in Table 6-2 and Table 6-3.

Table 6-2: Emulated Connection Characteristics, Congestion Experiments

Condition	Hold Time	Download Speed	Upload Speed	Packet Loss Rate	Delay	Jitter
Normal Operation	-	7 Mbit/s	1 Mbit/s	0.2%	19ms	0.5ms

Each row in Table 6-2 and Table 6-3 represents a network connection condition used in experimentation. Experiments use either a fixed connection speed, as shown in Table 6-2 or progressive throughput impairment as shown in Table 6-3. When a fixed connection speed is used, the speed of the connection stays at 7Mbit/s download and 1Mbit/s upload for the duration of the experiment. When progressive

throughput impairment is used, the throughput decreases from 18Mbit/s download and 1Mbit/s upload to 3Mbit/s download and 0.1Mbit/s upload over a period of 12 minutes, the impairment is held for 6 minutes, and returns to normal levels over a further period of 12 minutes.

Table 6-3: Emulated Connection Characteristics, Network Impairment Experiments

Condition	Hold Time	Download Speed	Upload Speed	Packet Loss Rate	Delay	Jitter
Normal Operation	-	18 Mbit/s	1.0 Mbit/s	0.2%	19ms	0.5ms
15 Mbit/s down	180 s	15 Mbit/s	0.5 Mbit/s	0.2%	19ms	0.5ms
12 Mbit/s down	180 s	12Mbit/s	0.4 Mbit/s	0.2%	19ms	0.5ms
9 Mbit/s down	180 s	9Mbit/s	0.3 Mbit/s	0.2%	19ms	0.5ms
6 Mbit/s down	180 s	6 Mbit/s	0.2 Mbit/s	0.2%	19ms	0.5ms
3 Mbit/s down	360 s	3 Mbit/s	0.1 Mbit/s	0.2%	19ms	0.5ms
6 Mbit/s down	180 s	6 Mbit/s	0.2 Mbit/s	0.2%	19ms	0.5ms
9 Mbit/s down	180 s	9 Mbit/s	0.3Mbit/s	0.2%	19ms	0.5ms
12 Mbit/s down	180 s	12 Mbit/s	0.4Mbit/s	0.2%	19ms	0.5ms
15 Mbit/s down	180 s	15Mbit/s	0.5 Mbit/s	0.2%	19ms	0.5ms
Normal Operation	-	18 Mbit/s	1 .0Mbit/s	0.2%	19ms	0.5ms

6.2.2 End User Services

The web browsing, video on demand, and file transfer end user services were selected as the services to be used in experimentation because web browsing, video on demand, and peer to peer file transfer services are among the most popular end-user services on fixed broadband access networks [Maier et al., 2009][Sandvine, 2012][Akamai, 2012], fulfilling the criteria of being realistic and relevant. Those services are discussed in Section 2.2.

For this experimentation, the expectations on sessions of each service type must be set so that the compliance of running sessions can be calculated. Service expectations must be set by service type, as described in Section 2.3.1.2. The service expectation values listed in Table 6-4 were selected using the guidance of the analysis of studies of real experienced service sessions presented in Section A.3, ensuring that they were realistic in the context of the network domain of DSL connected Home Area networks discussed in Section A.4. The expectation of each service was set statically and did not change during experimentation because they are configuration parameters that must be solicited from a user and are not likely to vary more frequently than daily. In a full system, such parameters would be configured using an

in-built user interface such as that described by Sventek et al. (2011) or by some other configuration system using a machine to machine interface.

Table 6-4: Service Expectation Parameters per Service

Service	Parameter	Value	Comment
Video on Demand	*High Quality*		The parameters selected for Video on Demand service experience are freezes and MOS, a measure of the quality of video playback. These parameters are discussed most frequently in the literature; see Section A.3.1.
	Freezes/Min	<=0.05	
	MOS	>=4.0	
	Standard Quality		
	Freezes/Min	<=0.1	
	MOS	>=3.0	
Web Browsing	*Request Size MBytes*	*Request Load Time (ms)*	Web page request load time is time to completion of load of a request in a web page and is the parameter used to assess web browsing in the literature as discussed in Section A.3.2. Typical encountered values for web request load times are used here to set expectations on the web browsing service.
	<=0.1	800	
	>0.1, <=1	2000	
	>1	60000	
File Transfer	Download Speed	4.2Mbit/s	The service expectation is set as 60% of the expected available bandwidth, 60% of 7Mbit/s

6.2.3 Scenarios

In order to reinforce the relevance of experimentation, three scenarios were designed based on the author's personal experience with running a Home Area network with limited connectivity and with a number of users. The scenarios were structured to highlight common problems that arise when using end user services in DSL broadband access connected HANs as discussed in Section A.4.

Sam subscribes to a broadband access network. The subscription is used to connect to the Internet. Sam, Alex (Sam's partner), Cathy (Teenager), Joe (Teenager), and Eva (Child) use Web Browsing, Video on Demand, and File Transfer services.

Sam, as the network owner, is using the Aesop system running on the Home Gateway to monitor and optimize usage of the Broadband Access Network connection. Sam sets the service expectations, priority, and throttling parameters for each service for each user as shown in Table 6-5. The throttling parameters are the limits applied to sessions by type and user when throttling is applied on a session for optimization. The priority and throttling parameters are set statically and do not

change during experimentation. Priority values were set based on the role of each user in the scenarios, and throttling parameters were set based on initial testing to determine what parameter values gave a minimum level of service. As with service expectations, in a full system these values would be set with a user interface or by another configuration system.

Table 6-5: Service Priorities and Throttling for User Service Sessions

User	Service	Priority	Throttling Parameters	
			Download Limit	Upload Limit
Sam	Web Browsing	1	N/A	N/A
	Video on Demand	2	2000Kbit/s	50Kbit/s
	File Transfer	5	500Kbit/s	100Kbit/s
Alex	Web Browsing	1	N/A	N/A
	Video on Demand	2	2000Kbit/s	50Kbit/s
	File Transfer	5	500Kbit/s	100Kbit/s
Cathy	Web Browsing	3	500Kbit/s	30Kbit/s
	Video on Demand	5	1000Kbit/s	35Kbit/s
	File Transfer	7	300Kbit/s	30Kbit/s
Joe	Web Browsing	3	500Kbit/s	20Kbit/s
	Video on Demand	6	800Kbit/s	30Kbit/s
	File Transfer	10	100Kbit/s	10Kbit/s
Eva	Web Browsing	7	200Kbit/s	10Kbit/s
	Video on Demand	3	1500Kbit/s	40Kbit/s
	File Transfer	10	100Kbit/s	10Kbit/s

The Aesop system monitors service usage in the HAN and checks if any services are not meeting expectations. If some services are not meeting expectations, the Aesop system optimizes bandwidth usage by selecting throttling actions that limit the bandwidth usage by lower priority service sessions and asks the Home Gateway to apply those throttling actions in the Home Area Network.

The Aesop system tracks how well optimizations perform over time. The level of throttling applied is incrementally increased or decreased as the compliance with expectations of services that have not been throttled decreases and increases.

6.2.3.1 Degradation in Video on Demand Session

Sam decides to watch the latest high definition movie from her broadband supplier's Video on Demand service. Initially, the quality of the video is excellent. However, the quality of the video deteriorates progressively due to packet loss in the

DSL broadband access network until, after 10 minutes, the picture quality is so bad that the picture is unwatchable and Sam switches the movie off. Sam checks her Aesop system, which confirms that the session quality did indeed deteriorate over its lifetime. Sam makes a note to call her broadband supplier next day and complain.

The timeline of Scenario 1 is shown in Figure 6-2.

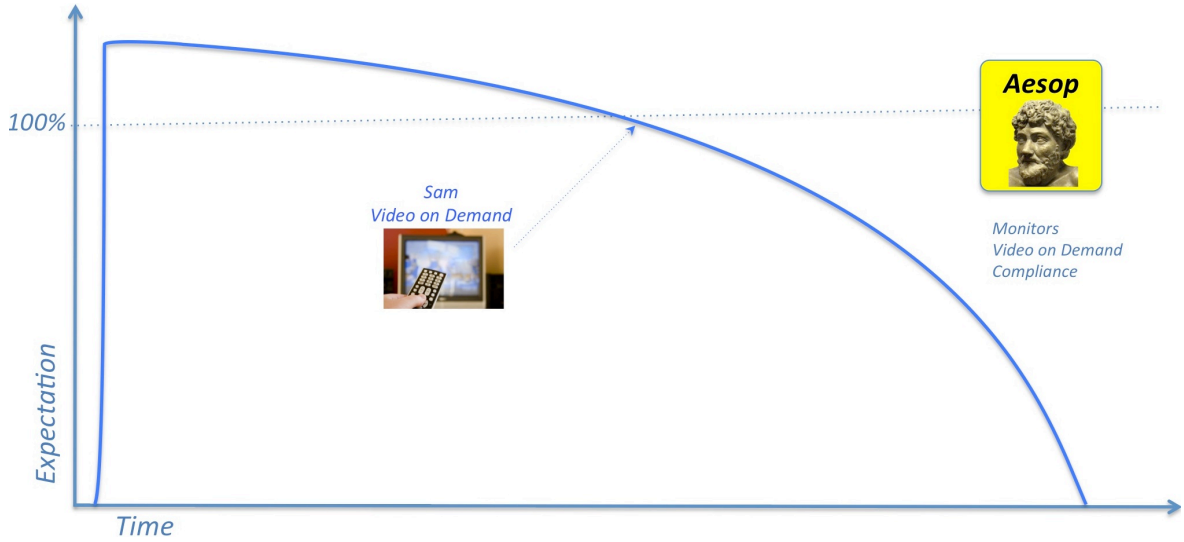


Figure 6-2: Timeline of Scenario 1

6.2.3.2 Congestion on Broadband Access Network Connection

It is evening time. Joe is a computer enthusiast and decides to download the DVD ISO of the latest version of his favourite Linux distribution from the Internet. He knows the download will take some time so he starts the download prior to leaving for football practice.

Cathy returns from a visit to the library and decides to update her social networking page. Her friends have suggested a number of interesting YouTube videos. Cathy decides to watch the videos while using her social networking site.

Alex has finished work for the day and decides to watch the latest episode of his favourite TV programme in High Definition using the Video on Demand service.

Sam receives a call from work to say that a presentation she is making the next day must be updated and some figures on the associated spreadsheet are incorrect. She logs onto her employer's documentation environment to update the documents.

All service sessions fail to meet expectations. The Aesop system sees that the compliance of services to expectations is not being met and limits the bandwidth of the file transfer session, which has the lowest priority. Aesop continues to monitor

the running service sessions and continues to limit the bandwidth of the lowest priority sessions until the remaining high priority sessions meet expectations.

Sam completes her work, and more bandwidth becomes available, the limits on low priority sessions are reduced. Alex's TV programme finishes and yet more bandwidth becomes available. The limit on the file transfer session is removed.

The timeline of Scenario 2 is shown in Figure 6-3.

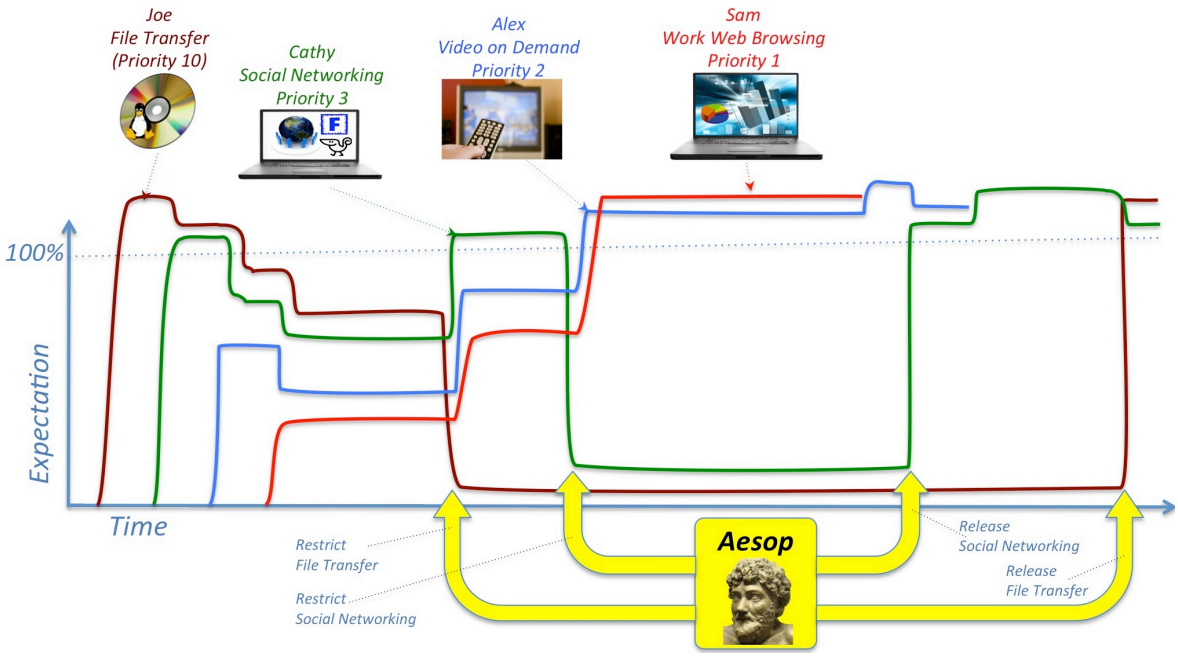


Figure 6-3: Timeline of Scenario 2

6.2.3.3 Transient Impairment on Broadband Access Network Connection

It is the weekend. Sam and Eva decide to watch a high definition children's movie on the family TV. Cathy also decides to watch a high definition move on her computer. Joe is chatting with some friends on a social networking portal and is browsing videos of computer equipment on an auction site. Alex is using his company's documentation environment to write a report on the previous week's project progress.

All service sessions meet expectations initially.

After some minutes, both Video-on-Demand sessions begin to fail to meet expectations because of a reduction in bandwidth on the broadband access network connection. The Aesop system determines that Cathy's Video on Demand session has lower priority and limits it. The remaining sessions meet expectations again.

After some more minutes the bandwidth available reduces further, the video on demand session fails to meet expectations again and the two web browsing sessions also fail to meet expectations. The Aesop system limits Joe's web browsing session

because it has the next lowest priority. The Aesop system continues to limit lower priority sessions until all sessions except Alex's highest priority session is limited.

The Aesop system continues to monitor the sessions and the network connection. As bandwidth availability increases, Aesop progressively removes the limitations on sessions. Finally, when the connection bandwidth is fully restored, the limitations on all sessions are removed and all sessions again comply with expectations.

The timeline of Scenario 3 is shown in Figure 6-4.

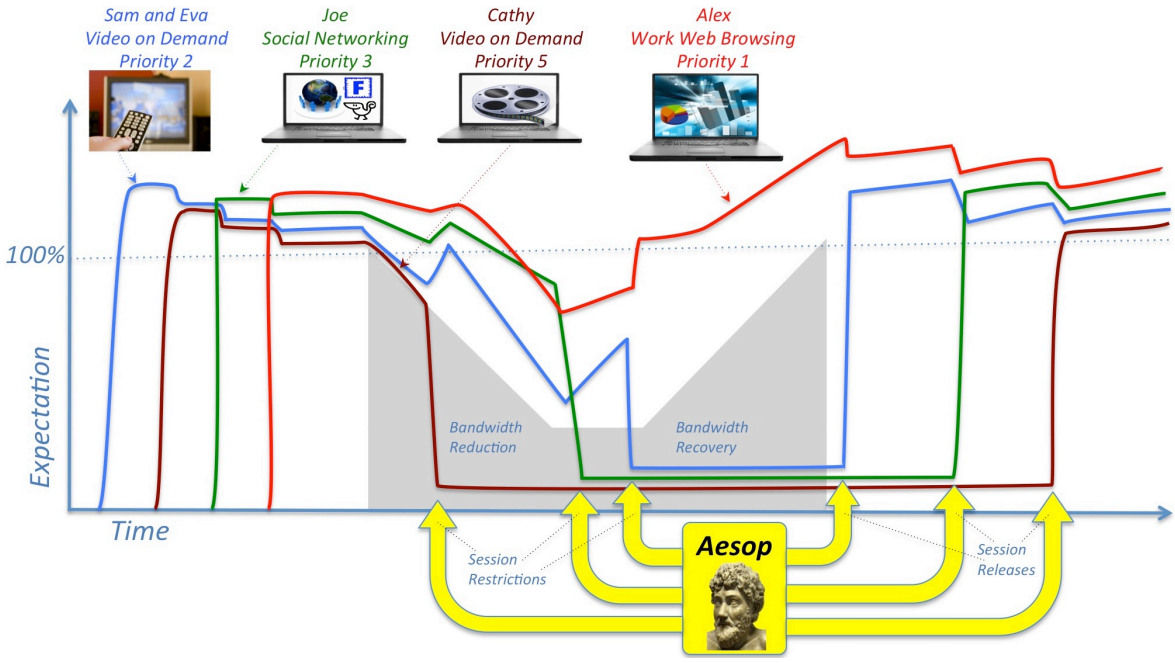


Figure 6-4: Timeline of Scenario 3

6.2.4 Simulation Environment

Experiment 1 was executed in a simulated environment, designed to produce realistic and measurable results for Video on Demand sessions. A simulator generates terminal reports that contain the metrics produced by a Video on Demand client where the session experiences progressive packet loss over a 10 minute time period. The client runs an end user session that follows Scenario 1 in Section 6.2.3.1.

The metrics that vary in the terminal reports are packet loss, a quantitative Quality of Service (QoS) metric, and the Mean Opinion Score (MOS), a qualitative Quality of Experience (QoE) metric calculated by the client. At session start, packet loss in the session is zero. As the session continues, packet loss increases and MOS values deteriorate. When packet loss reaches a level of over 2%, the picture becomes unwatchable [Handurukande et al., 2011] and the session is terminated. In the scenario, MOS values vary from 5.0 (Perfect) to 0.0 (worst). For the purposes of

simulating MOS values, an exponential relationship between packet loss and MOS is assumed [Fiedler et al., 2010], and is approximated using Equation 6-1 below. Figure 6-5 shows the approximated relationship graphically. It is important to note that the actual relationship between packet loss and MOS falls much more steeply at low levels of packet loss and levels out as packet loss values increase [Fiedler et al., 2010].

Equation 6-1: MOS approximation for Video on Demand Session Simulation

$$MOS = P - (e^l - 1)\ if\ (P - (e^l - 1) > 0, 0\ otherwise$$

P is a perfect MOS score of 5.0 and l is packet loss expressed as a percentage.

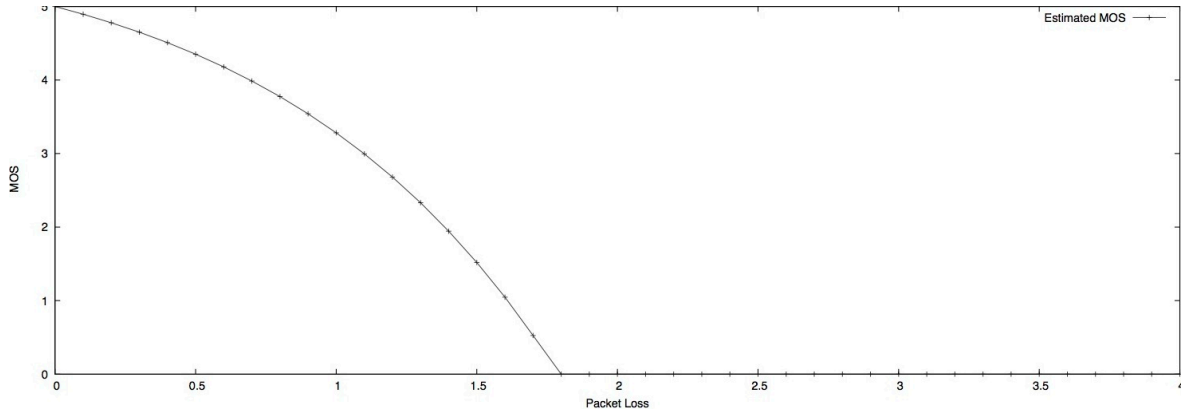


Figure 6-5: Approximated Relationship between Packet Loss and MOS

Each simulated Video on Demand session was designed to run for 10 minutes with packet loss increasing from 0% to 0.2% during that period, simulating a gradual progressive impairment. Terminal reporting was set to report every minute on each session, representing near real time reporting [NGMN, 2007b]. Therefore, in a single simulated Video on Demand session, 11 terminal reports per session are generated at 1 minute intervals to simulate 10 minutes of operation: an initial terminal report at session start, 9 periodic terminal reports at one minute intervals, and a terminating terminal report. Packet loss is varied from 0% to 2% over the lifetime of the Video on Demand session and Equation 6-1 is used to calculate MOS values from packet loss.

The simulation environment created for experimentation is shown in Figure 6-6, and ran on a single computer, the "saor" host. The simulator produced terminal reports, which were written to the hard disk of the computer. The Knowledge Monitoring and Knowledge Persistence components of Aesop (see Figure 4-1) monitored service experience and context knowledge and stored it to a semantic store, also running on the same computer. The host was performing no other tasks besides routine operating system housekeeping. The details of the "saor" host and

software components used in simulation appear in Table E-1 of Appendix E and Table G-1 of Appendix G respectively.

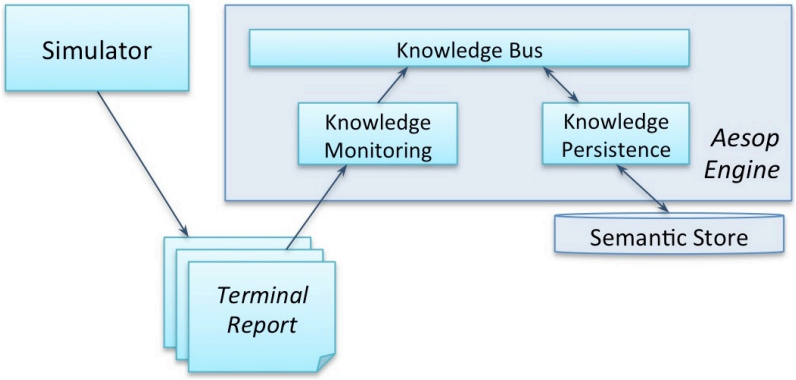


Figure 6-6: Simulation Environment

6.2.5 Test Bed Environment

A full Home Area Network test bed, shown in Figure 6-7, was designed, built, and commissioned for the execution of Experiments 2 and 3. It consists of a real Home Area Network with four laptops connected to a Home Gateway using an Ethernet switch. SECCO was used to run end user sessions that were measurable. The details of the test bed equipment appear in in Table E-1 of Appendix E; photographs of the test bed appear in Appendix F.

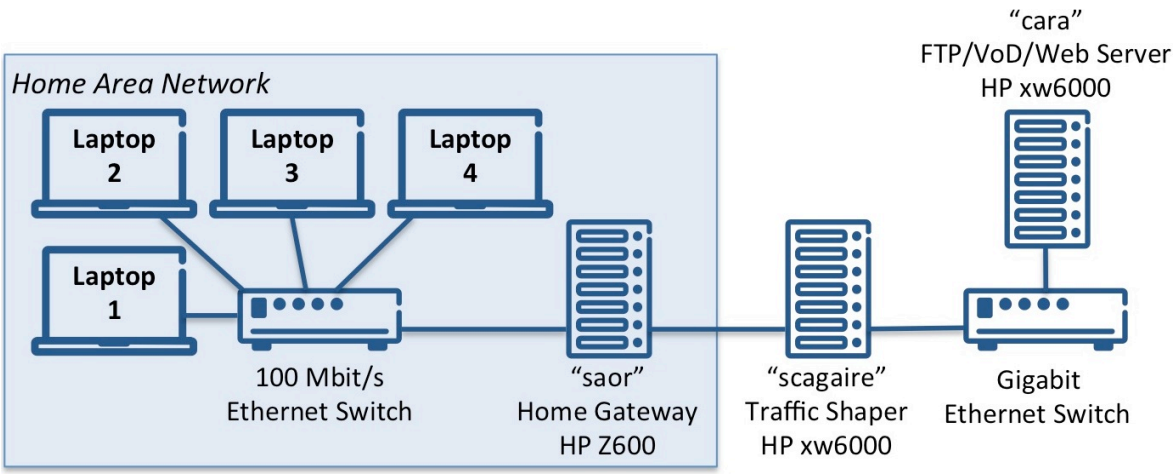


Figure 6-7: Test Bed for Experimentation

The test bed is designed so that service degradation can be controlled at a single point, in the traffic-shaping node. Cabled network connections are used in the HAN and to connect servers so that network performance in those network segments is an order of magnitude greater than on the emulated broadband access network and can be ignored in experimentation.

The Home Gateway node and the Traffic Shaper node both have two Network Interface Cards (NICs) installed in them. Traffic between those NICs is bridged using Linux bridging support [Linux Bridging, 2013]. The Linux netfilter framework [Netfilter Framework, 2012] is used to apply traffic controls to the outgoing and incoming NICs on the traffic shaper node to simulate the performance and impairments in a broadband access network using the values from Table 6-2 and Table 6-3. Aesop optimizes service delivery by restricting traffic carrying low priority services by applying traffic controls on the outgoing and incoming NICs.

Aesop uses the time stamps on terminal reports to group terminal reports into snapshots so it was important that the system clock on all hosts were synchronized in the Home Area Network. A NTP [IETF, 2010b] daemon was used on each laptop and on the Home Gateway to ensure the system clocks were synchronized.

A logical view of the test bed is shown in Figure 6-8. The server machine ran a FTP [IETF, 1985] server as a target for file transfer sessions. A number of test videos were available on a RTSP [IETF, 1998a] server for video on demand sessions and were served by a VLC instance in server mode [VideoLAN, 2013] running on the server machine. The server machine also ran an Apache Web server [Apache, 2013b]. Two web sites ran on the web server, a workplace web site with web pages typically less than 1MB in size, made up of text and small images. Pages on the social networking site averaged 4MB in size, made up of large images and embedded flash videos.

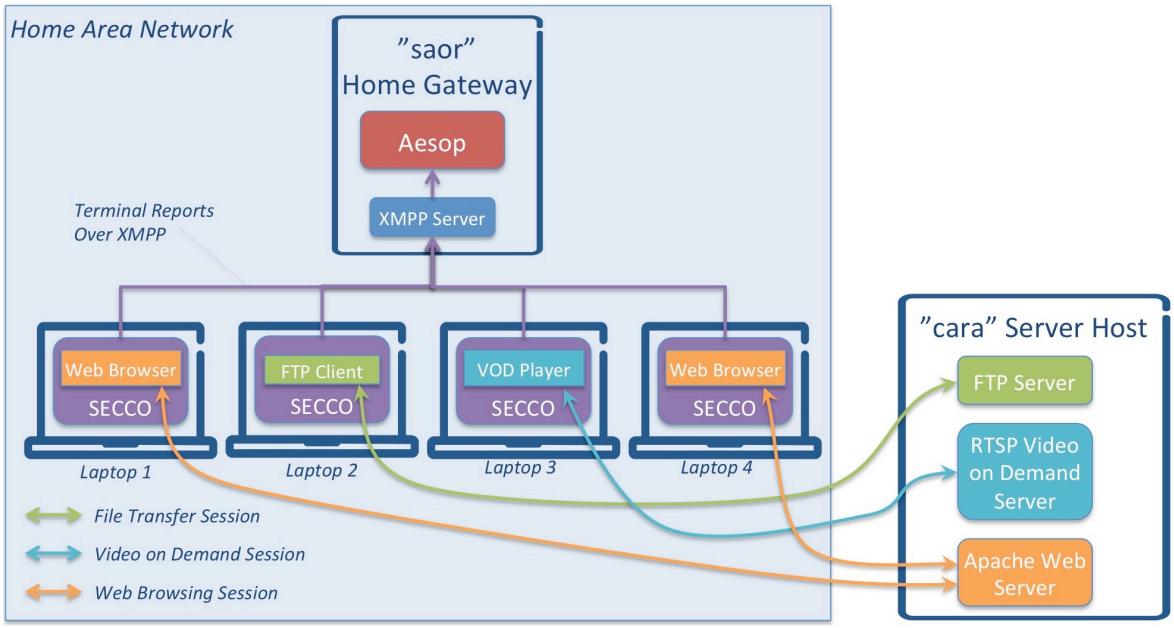


Figure 6-8: Logical View of Test Bed with Running Sessions and Terminal Reporting

In experiments, end user clients ran under the control of SECCO, as shown in Figure 6-8. An XMPP server [IETF, 2011] ran on the home gateway, to which Aesop subscribed. SECCO monitored the sessions running one each client and sent terminal reports as XMPP messages to the XMPP server, which forwarded them to Aesop.

6.2.6 Experimental Timings and Runs

In each experimentation run, a time budget of 10s is allocated for each of the four phases of knowledge processing in Aesop: semantic monitoring, analysis, optimization planning, and throttling action execution. A 10s interval is realistic because it is sufficiently long to mitigate momentary network fluctuations, and a combined time budget of 40s for a full autonomic loop is short enough to be considered near real time [NGMN, 2007b].

Statistical analysis was selected as an analysis method for measurements of metrics used to determine the effectiveness and performance of Aesop's autonomic functions, those metrics appear in Table 6-6. Measurements were grouped by minute for statistical analysis. In Experiment 1, Monitoring time is the only measurement used. A total of six measurements are collected in the test with the least number of terminals reporting. In Experiment 2, each of four clients and the Home Gateway send a terminal report to Aesop every minute, giving a total of 30 measurements for Monitoring time. All other metrics are recorded during periodic operations that run every 10 seconds, giving 6 measurements per minute.

A Normal distribution of the population was assumed for the metrics in Table 6-6 because an examination of those metrics performed during initial testing indicated that they were not skewed and did not exhibit a large number of outliers [Moore et al., 2009]. When the number of measurements is more than 15, confidence levels in mean, standard deviation, and margin of error in normal populations do not vary greatly [Moore et al., 2009].

In order to obtain enough measurements to have stable confidence levels in the margin of error of results, 10 experimental runs were executed in Experiment 1 and six experimental runs were executed for each scenario in Experiments 2 and 3. As shown in Table 6-6, this gave at least 60 Monitoring time measurements for each minute in Experiment 1 and at least 36 measurements for all metrics for each minute in Experiments 2 and 3.

Following experimentation, a strong indication of Normal distribution was confirmed for all the metrics in Table 6-6 by demonstrating a very strong linear tendency on Normal quantile plots generated using all the measurements collected for those metrics in experimentation [Moore et al., 2009]. Those plots appear in Appendix H. In the statistical analysis performed on the measurements, the mean, standard deviation, and margin of error at a 99% confidence level was calculated for each metric for each minute of an experiment. The results of the statistical analysis for each experiment are presented and interpreted in the individual sub-sections for each experiment in this chapter.

Table 6-6: Metrics Examined using Statistical Analysis

Metric	Evaluation Property	Measurements Per Run Per Min		No. Experimental Runs Per Min		Total Measurements Per Min	
		Exp. 1	Exp. 2, 3	Exp. 1	Exp. 2, 3	Exp. 1	Exp. 2,3
Compliance	Effectiveness	-	6	10	6	-	36
Monitoring Time	Performance	6	30	10	6	60	180
Analysis Time	Performance	-	6	10	6	-	36
Optimization Time	Performance	-	6	10	6	-	36
CPU Usage	Performance	-	6	10	6	-	36
Memory Usage	Performance	-	6	10	6	-	36

6.2.7 Experimental Methodology

The experimental methodology, based on software testing and configuration and change management workflows [Kruchten, 2000] from the IBM Rational Unified Process (RUP) [IBM, 2012a] software development process framework, ensured that experimentation was executed in a rigorous manner, and that all experiments were measured, traceable and repeatable, and all analysis of results was traceable and repeatable.

The simulated and test hardware and software used in experiments were controlled. The specifications of all computers were recorded and the same Linux OS version and patch level was installed on all machines, as shown in Table E-1 of Appendix E. Software artefacts developed in the course of this research were stored and controlled in a software repository. The versions of all third party software were recorded, as shown in Table G-1 of Appendix G. Where a third party software

component was available in the Linux repository, that version was used, ensuring all computers had the same software version. Where software was manually installed, the software was downloaded and stored in the software repository. That version was then installed from the repository to ensure the same version was used on all computers. The software repository was regularly backed up to two separate remote servers.

Experimental runs always followed one of the scenarios described in Section 6.2.3. An experimental scenario specification was prepared for each scenario, which specified a sequence of actions to be performed in an experimental run to execute that scenario. The scenario specifications appear in Appendix I. Run log tables based on those specifications were printed and used during experimentation to record the execution time of each action in each scenario in experimental runs. Those run log tables appear in Appendix J.

Experiment 1 used scenario 1, Experiment 2 and Experiment 3 used both scenario 2 and scenario 3. A number of runs of each experiment were executed in order to gather enough measurements to conduct analysis, as explained in Section 6.2.6. The run log tables show that each action in each scenario in each experimental run was executed and was executed at the correct time. This confirms that each separate run of all experiments were executed as specified in their scenario specifications, so comparison of the results of a run with all other runs of that experiment using the same scenario specification is appropriate.

Experimental execution was automated with scripts where possible. Experimental runs in the simulated environment were entirely automated using scripts. In runs on the test bed, each action in the scenario specification was automated using a script; the scripts were started manually at the correct time during the experiment.

The following results and information was collected during experimental runs in the simulated environment, samples of which appear in Section K.1 of Appendix K:

1. Java logs from Aesop containing results on the performance of monitoring, giving timings on monitoring, processing, and storing operations for terminal reports.
2. Java logs from the simulator giving timing information on terminal report generation.

The following results and information was collected during experimental runs in the test bed environment, samples of which appear in Section K.2 of Appendix K:

1. Recordings of all session knowledge and session metric knowledge reported by Aesop from its Knowledge Base were stored in a relational database using the Aesop database client. Session knowledge gives details of the sessions that executed in experimental runs as well as optimization actions taken by Aesop on sessions. Session metric knowledge contains the service experience and context from terminal reports and service expectations and compliance added by Aesop analysis operations. A dump of the result database was taken and stored to disk at the end of each experimental run.
2. Java logs from the Aesop server containing results on the performance of Aesop monitoring, analysis, and optimization operations.
3. A screenshot of the Aesop client user interface showing a plot the compliance of all sessions during the experimental run.
4. All incoming terminal reports received by Aesop during each experimental run.
5. Home gateway CPU and Memory performance metrics logged every 10 seconds during the run using Linux system statistics utilities [Linux Sysstat, 2013].

All raw results from experimental runs in both the simulated and test bed environments were archived in the software repository, and are available on the optical disk accompanying this thesis.

Post experimental analysis was carried out in a rigorous manner. All experimental analysis calculation and production of plots was automated for repeatability and traceability. The analysis of experimental results used a combination of scripts and SQL queries to perform calculations and generate tables and plots. The result logs from simulated experimental runs were inserted into database tables after experimental runs. In test bed experimental runs, session data and metrics for each run were inserted into database tables during experimental runs, all other result data for each run was inserted into the run database using scripts after experimental runs. Analysis calculations on individual run data and across runs were implemented as SQL queries. All analysis scripts and queries were archived in the software repository, and are available on the optical disk accompanying this thesis. Example analysis scripts and SQL queries appear in Appendix L.

All experiments performed as part of this research are repeatable. Simulated experiments are fully scripted; the script can be re-run by re-executing the script. Experiments from the test bed are also repeatable. Aesop can run in “file” mode as

well as “XMPP” mode (see Section 4.4.4.1), it can read the saved terminal reports from experimental runs to re-execute those runs.

6.3 Experiment 1: Semantic Monitoring of Terminal Reports

The motivation for this experiment was to evaluate the semantic monitoring autonomic function of Aesop. The experimental hypothesis was cast so as to provide evidence that service experience and context could be monitored using semantic lifting and stored persistently in near real time using the Aesop approach.

Terminal reporting was active on six terminals in a Home Area Network. A total of six terminals were selected as a reporting load because having six terminals active at once in a home network was considered to be a realistically heavy level of HAN usage. Each terminal reported every minute, frequently enough to provide service experience and context to Aesop so as to allow monitoring in near real time [NGMN, 2007b].

6.3.1 Hypothesis

That it is possible to extract semantic information which references a set of ontologies from terminal reports in a time window of a minute when terminal reporting is set up so that each of six terminals report every minute.

6.3.2 Experimental Setup and Execution

In order to support the hypothesis of the experiment, it must be shown that it is possible to lift and store semantic information at a reporting load of six terminals reporting every minute. The experimental setup was chosen to provide evidence to test the hypothesis; it was designed to assess the performance of semantic lifting and storage in the Knowledge Monitoring and Knowledge Persistence components of the Aesop system shown in Figure 4-1.

In the experiment Scenario 1, the Video on Demand degradation scenario described in Section 6.2.3.1, was run in the simulation environment described in Section 6.2.4. Each run was executed as a series of tests, with the number of simultaneous simulated Video on Demand sessions reported by terminals being varied for each test, as shown in the experimental scenario specification for Scenario 1 in Table I-1 in Appendix I. The first test of each experimental run was run with six terminals reporting, the specification required to test the experimental hypothesis. In order to gather a richer result set on the performance of semantic lifting and

persistence to the semantic store, further tests were conducted where the number of terminals reporting was varied up to a level of 300 terminals.

All experimental runs had a run time of 10 minutes, as specified in the scenario specification in Table I-1 in Appendix I. In order to ensure that all sessions in an experimental run did not start at the same instant, the simulator was set to start terminal report generation for individual sessions in a test at some random time between 0 and 1 minutes.

Ten runs of the experiment were executed in order to obtain enough measurements for statistical analysis as discussed in 6.2.6. The run log for the experiment appears in Table J-1 in Section J.1 of Appendix J.

6.3.3 Experimental Results

The parameters that were measured during the processing of each terminal report are listed in Section M.1 of Appendix M. The three most important parameters are *Generation to Store Time*, *Processing Time*, and *Intercept Time*. *Generation to Store Time* was the time taken from generation of a terminal report in the simulator to its semantic information being stored in the RDF store: t_1 to t_4 in Figure 6-9. The *Processing Time* was the time taken to process a terminal report from reading the report from the file system to complete storing its semantic information in the RDF store: t_2 to t_4. The *Intercept Time* was the portion of the Processing Time taken from reading a report from the file system to being ready to store the semantic information in the RDF store; Processing Time without the RDF store contribution: t_2 to t_3. Intercept time is the time taken to lift a terminal report into semantic form.

The *File System Wait Time* was the amount of time that a terminal report remains on disk after creation by the simulator until processing of it by Aesop begins. At low reporting loads, this time period was always less than one second because Aesop slept for one second when no terminal reports existed on disk.

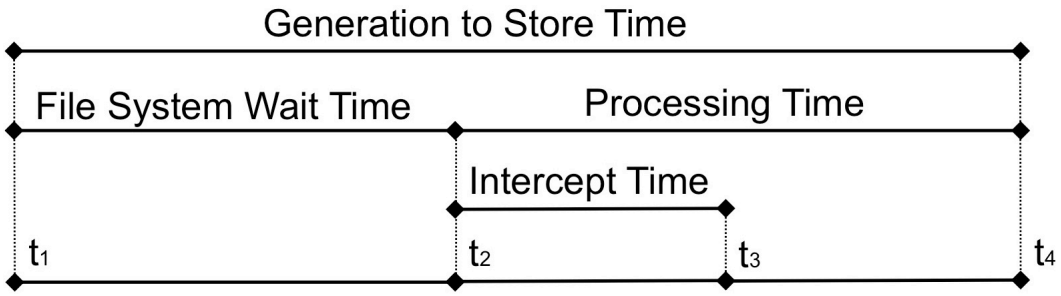


Figure 6-9: Relationship between Time Parameters in Experiment 1

A statistical analysis of the results of Experiment 1 appears in Table N-2 in Section N.1 of Appendix N.

6.3.4 Analysis of Results

Figure 6-10 shows the average and standard deviation of Processing Time and Intercept Time for tests where six terminals are reporting over the ten runs of Experiment 1. Examining the plots, one can observe that the average Processing Time and Intercept Time is less than 0.5 seconds and 20 milliseconds respectively for all runs at that reporting rate. These results bear out the hypothesis that it is possible to populate a Knowledge Base with knowledge that references a set of ontologies from terminal reports in the time window of a minute when terminal reporting is set up so that each of six terminals report every minute. Aesop can lift knowledge from terminal reports, populate instances of the EUSAO ontology with that knowledge, and store that knowledge persistently at that reporting rate.

The results show that, at the relatively low reporting rate of 6 terminal reports per second, in-memory semantic lifting operations take tens of milliseconds and operations to the semantic store take fractions of seconds.

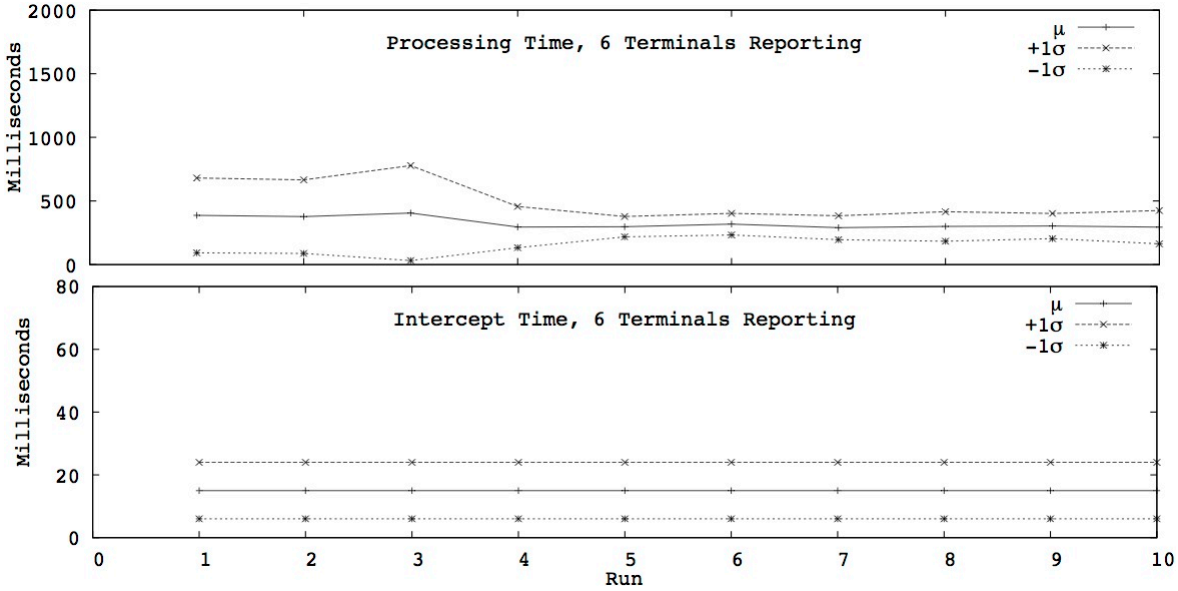


Figure 6-10: Statistical Analysis of Results, 6 Terminals Reporting

Examination of results from other tests with higher reporting loads gives further insight into the performance of semantic lifting and persistence operations.

6.3.4.1 Reporting Limit on Aesop

As explained in Section 6.3.2, each test should take between 10 and 11 minutes to complete because each simulated terminal starts reporting after a delay of 0 to 1

minutes and reports for 10 minutes. Elapsed time per run, as shown in Figure 6-11, shows that the observed length of each test is highly correlated for all runs. With reporting loads up to 175 terminals, test times are very close to 11 minutes, indicating that Aesop is processing terminal reports as they are generated. Elapsed time per run increases well beyond 11 minutes for higher reporting loads because Aesop reached its maximum speed for processing terminal reports. A backlog of terminal reports built up and Aesop continued to run after simulation completed to clear that backlog. Therefore, the observed results indicate that Aesop can lift and semantically store terminal reports at loads of up to 150 terminals reporting every minute, a rate of 2.5 terminal reports processed per second.

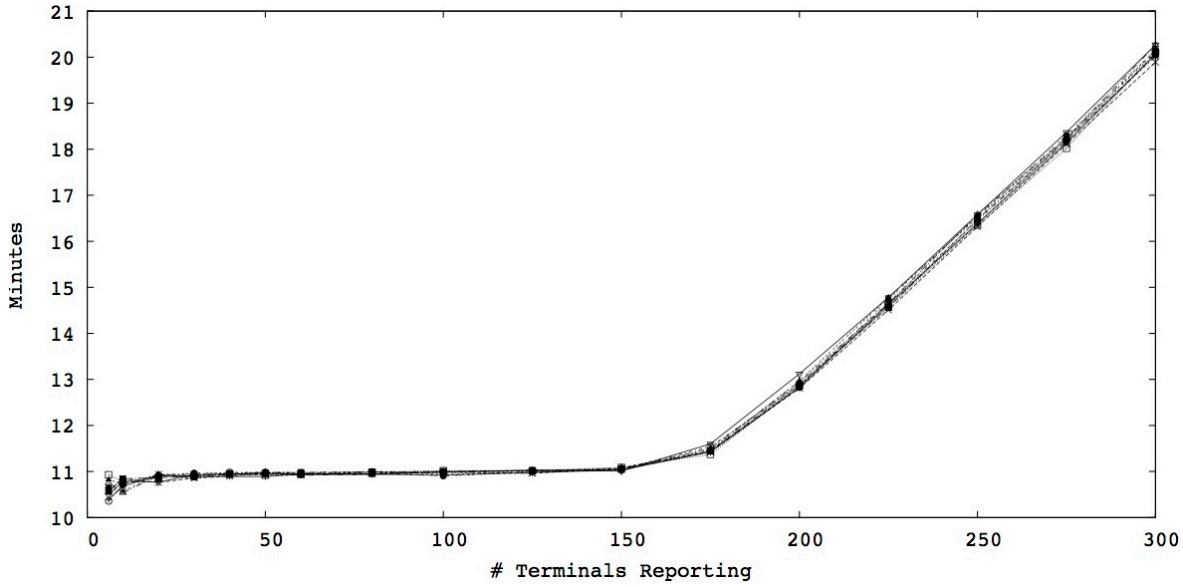


Figure 6-11: Elapsed time per Run, 10 Runs

6.3.4.2 Generation to Store, Processing, and Intercept Times

Figure 6-12 shows a plot of the average Generation to Store Time for the 10 independent experimental runs. This plot shows the same highly correlated trend across the runs as in the elapsed time per run plot in Figure 6-11; small numbers of seconds up to reporting loads of 150 terminals per minute, and an increase in the Generation to Store Time taken per terminal report beyond that reporting level as the terminal report backlog was cleared.

The plot for average Processing Time over the 10 runs is shown in Figure 6-13. Average Processing Time is not well correlated across the runs at very low reporting loads but does become well correlated across runs at reporting loads above 30 terminals reporting per second. It exhibits an increasing trend at reporting loads from 60 terminals.

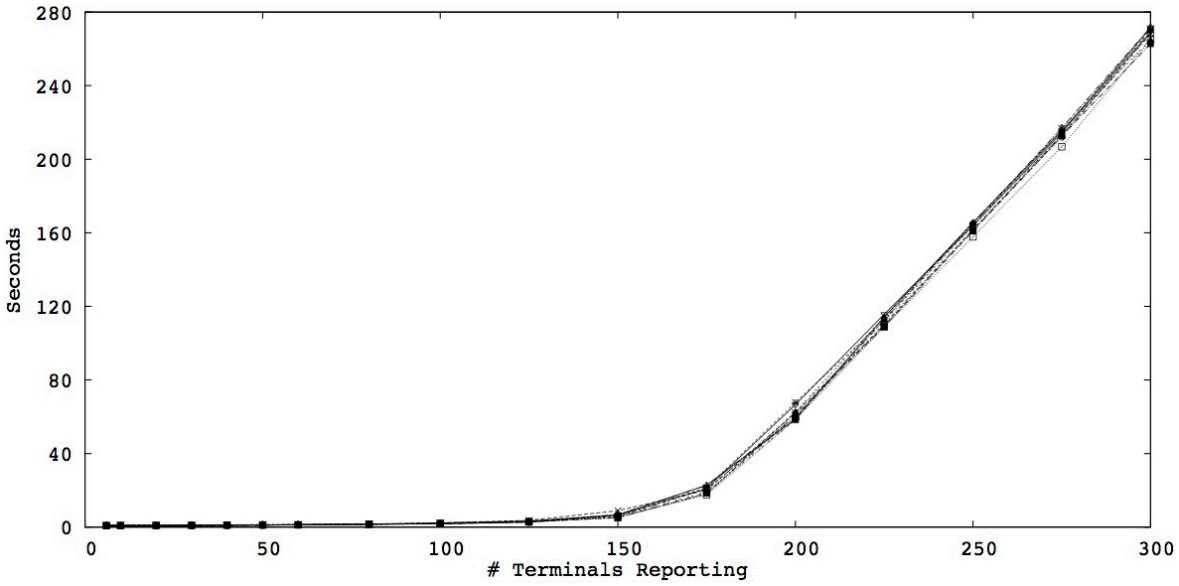


Figure 6-12: Average Generate to Store Time: 10 Runs

The plot of average Intercept Time in Figure 6-14 is also highly correlated across runs. It shows that the time taken lift the knowledge from a terminal report into semantic form is always less than 20ms and decreases to 4ms as reporting loads increases. At 4ms per terminal report, Aesop can lift 250 terminal reports per second. The higher Intercept Time values at low terminal loads occur because Java XML handling classes become inactive at such load levels and must be re-instantiated.

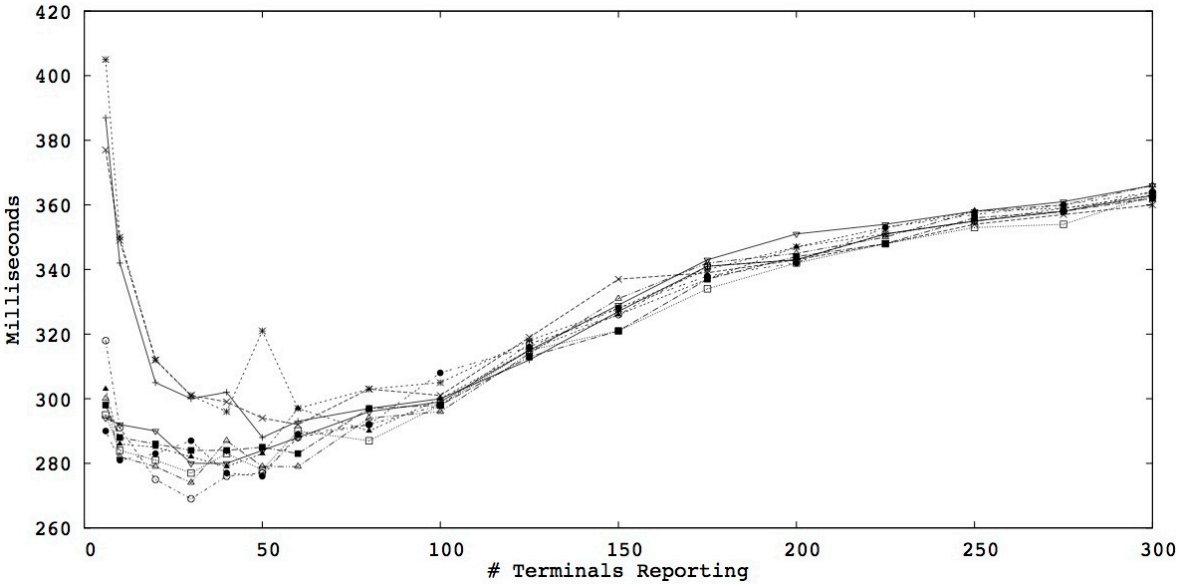


Figure 6-13: Average Processing Time, 10 Runs

6.3.4.3 Relationship Between Average Processing Time and Average Intercept Time

The plot of average Processing Time and average Intercept Time in Figure 6-13 and Figure 6-14 show that Intercept time represents a very small proportion of the overall time to process a terminal report, 4ms of 360ms at high reporting loads. The

trends shown in Figure 6-13 are therefore almost entirely attributable to the performance of the semantic store.

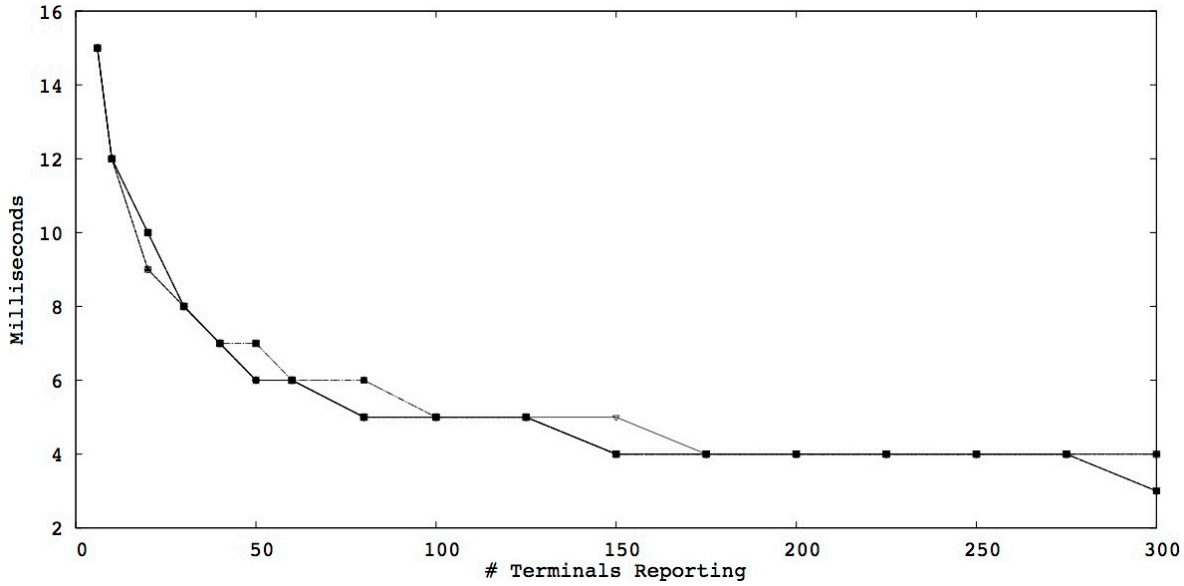


Figure 6-14: Average Intercept Time, 10 Runs

Average Processing time shows an increasing trend at reporting loads above 30 terminals reporting per second. This trend is expected because at higher loads, the RDF store performance deteriorates because the amount of links to be built as extra data is added increases. The low near-linear nature of this increase indicates that storage time to the RDF store is stable as storage load increases.

The high Average Processing Time observed in 4 runs at low reporting loads merits examination. The data for those runs showed that 2-5% of the database writes for the tests in question took longer than 1 second, much more that the average of 0.5% of slow writes across all run tests. The high rate of slow writes on tests with low loads occurred due to database connections becoming inactive due to low write rates.

The maximum number of terminal reports that can be handled per second by Aesop is the inverse of processing time. A processing time of 200ms per terminal report means that 5 reports can be handled per second. Table 6-7 takes the lowest and highest average processing times and sets out what the maximum number of terminal reports per second and maximum reporting load would be for those processing times, excluding all other factors. If each terminal reports every minute, Aesop should be able to handle and persist reporting loads of between 148 and 223 terminals.

Table 6-7: Lowest and Highest Processing Times

	Run	Test	Av. Processing Time (ms)	Max. Terminal Reports	Max. Reporting Load
Low	6	4	269	3.72	223
High	3	1	405	2.47	148

6.3.4.4 Spread of Processing Time and Intercept Time Data:

Using average values does not always give an accurate view of a set of measurements so the spread of Processing Time and Intercept Time measurements was examined, the two most interesting measurements in Experiment 1. The combined spread of Processing Time and Intercept Time data for all runs is presented for each test in Figure 6-15 and Figure 6-16 in the form of box and whisker plots. As an example, in the leftmost plot in Figure 6-15, the data spread for 660 tests is presented; 6 terminals produce 11 reports each in 10 runs. High outliers, data points 1.5 times the Inter Quartile Range (IQR) beyond the upper quartile were eliminated. There were no low outliers. The percentage of data points identified as outliers is indicated in parenthesis on the x-axis labels as a percentage.

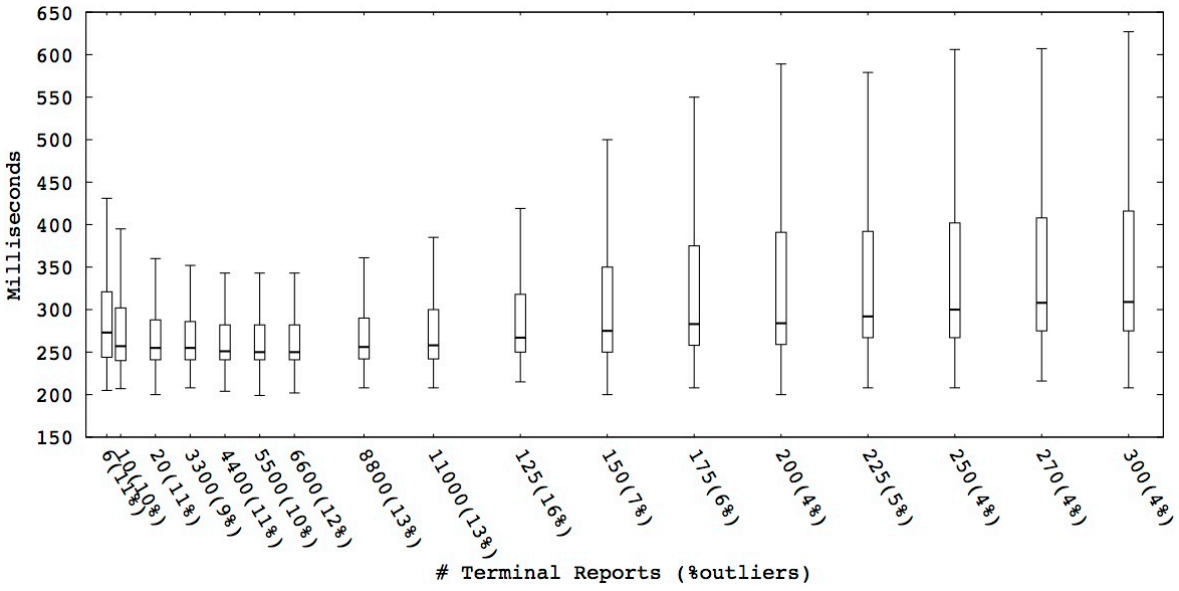


Figure 6-15: Spread of Processing Time Data, All Runs

The spread in Processing Time is skewed to the right, with spread and skew increasing as terminal load increases, with the increase closely tracking the trend of Average Processing Time shown in Figure 6-13. The spread in processing time is wide, and widens as processing load increases. Because Processing Time is almost entirely composed of the time taken to execute operations to the semantic store,

these measurements indicate that the spread in time taken to execute such operations becomes wider as load increases.

The spread in Intercept Time is also skewed, but spread and skew decreases as terminal report load increases, closely tracking the trend in Average Intercept Time in Figure 6-14. The measurements indicate that Intercept Time remains stable and predictable at least to the maximum measured load in the experiment.

The box and whisker plots identified a relatively high number of outliers. The plots in Figure 6-17 and Figure 6-18 show data points including outliers in Processing and Intercept Time results for tests with 300 terminals reporting for all runs, and were produced to identify patterns in outliers. There are a significant number of outliers in the early measurements, due to system initiation costs. The Processing Time plot shows that outliers occur frequently during the run, whereas the Intercept Time plot shows that outliers are extremely rare once the system has initiated.

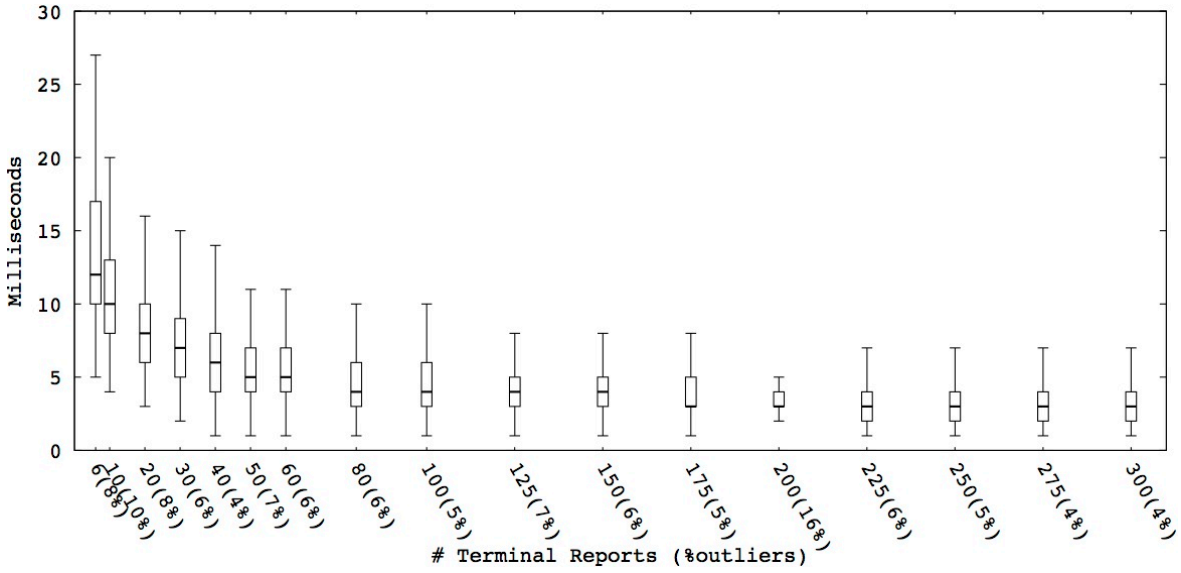


Figure 6-16: Spread of Intercept Time Data, All Runs

This analysis shows that Intercept time, the time taken to process a terminal report semantically, is stable and has low spread in the system. Processing Time, which includes the time taken to store the semantic information to the RDF store, is less stable and exhibits higher spread.

6.3.5 Summary

The goal of Experiment 1 was to evaluate the semantic monitoring function of Aesop. The experimental results showed that monitoring using semantic lifting and storage of service experience and context was demonstrable in near real time.

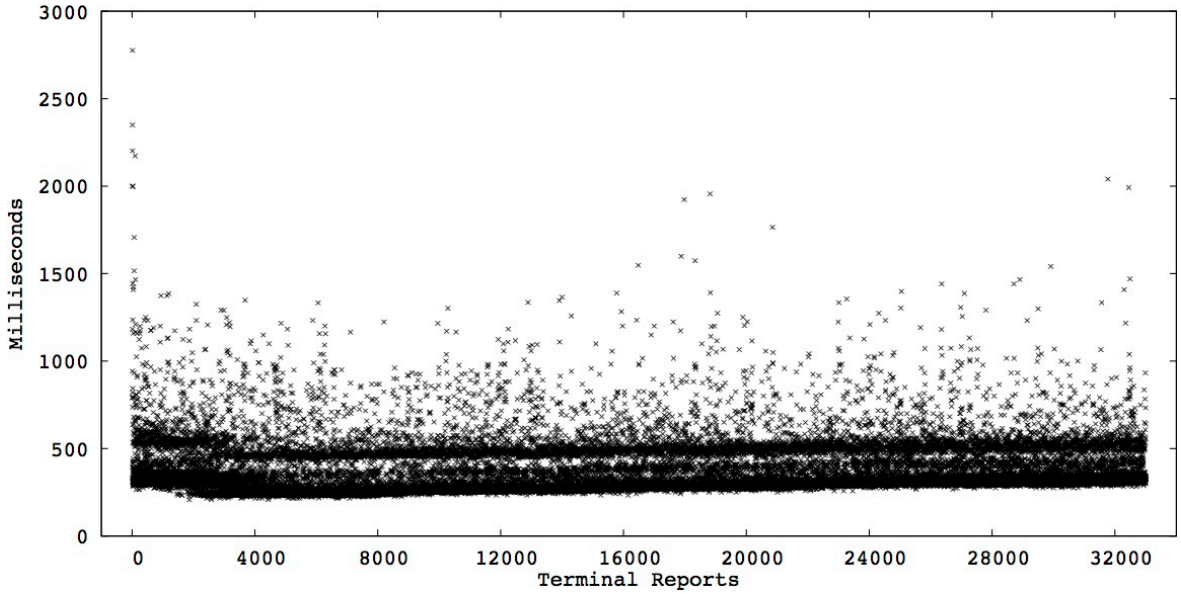


Figure 6-17: Processing Time Data Points (300 Terminals Reporting), All Runs

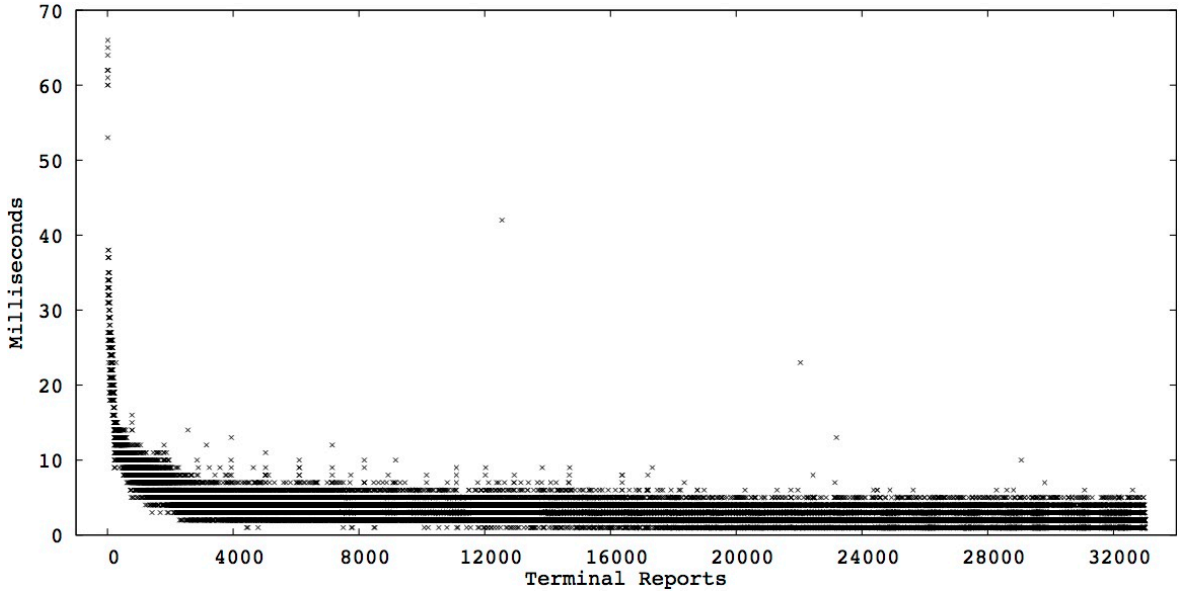


Figure 6-18: Intercept Time Data Points (300 Terminals Reporting), All Runs

The experimental results showed that, when six terminals are reporting, Aesop lifted knowledge from terminal reports, populated instances of the EUSAO ontology with that knowledge, and stored that knowledge persistently at an average speed of 0.5 seconds per terminal report. They also demonstrated that the in-memory operation of lifting knowledge and populating that knowledge took, on average, less than 20ms. The operation of persistently storing that knowledge to the semantic store was much slower.

Experimental results from tests with higher reporting loads showed that, as reporting loads increase, the time taken to semantically lift knowledge from a terminal report into the EUSAO ontology stabilizes at an average of 4ms per terminal

report, with a very tight spread on that average value. Aesop can therefore lift terminal reports at a rate of 250 reports per second. This indicates that Aesop can scale to handle the lifting loads one might reasonably expect in a Home Area network and is likely to execute efficiently on a machine with much lower capabilities than that used in this experimentation such as a Home Gateway or Set Top Box.

The total time to process a terminal report increases to a maximum average rate of 405ms with a wide spread as load increases because of the overhead introduced by the semantic store. This represents a rate of about 2.5 terminal reports per second. In-memory operations are 100 times faster than operations that use the semantic store and have a more predictable execution time.

6.4 Experiment 2: Semantic Analysis of Service Experience

The overall purpose of this experiment was to evaluate the semantic monitoring and analysis autonomic functions of Aesop using realistic scenarios and a realistic test bed. The goal of the experiment was to measure the extent to which service experience of a set of services could be monitored and semantically analysed to deduce compliance with service expectations in near real time.

Experimental results were analysed to determine the *accuracy* and *performance* of Aesop's monitoring and analysis operations. The accuracy of semantic monitoring was assessed by comparing the individuals of concepts read from Aesop's Knowledge Base with data read from the source elements in the terminal reports input into Aesop. The accuracy of the Semantic Analysis algorithm was assessed by comparing the compliance results calculated by Aesop with results produced by an independent off-line calculation performed by a Java program running over the terminal reports input into Aesop. Timing information on Aesop monitoring and analysis operations were analysed statistically to assess their performance.

6.4.1 Hypothesis

That service experience of a set of services can be semantically analysed accurately and periodically to deduce their compliance with previously set service expectations in a 20 second time window, composed of a 10 second window for monitoring and a 10 second window for analysis.

6.4.2 Experimental Setup and Execution

The experiment was designed to provide evidence to show that Semantic Analysis of the compliance of services with expectations was accurate. In addition, measurements were required to show that the performance of the system was adequate to meet the time constraints specified in the hypothesis. The experiment exercised all components of the Aesop system as shown in Figure 4-1 except those concerned with optimization planning and execution, thus providing measurements on the monitoring and analysis autonomic functions of Aesop.

All experimentation was carried out on the HAN test bed environment described in Section 6.2.5. In order to provide measurements on how the system performed in two different network situations, two scenarios were used in the experiment; the congestion scenario described in Section 6.2.3.2 and the impairment scenario described in Section 6.2.3.3. Six independent experimental runs were conducted for each scenario in order to obtain enough measurements for statistical analysis as discussed in Section 6.2.6, giving a total of 12 experimental runs.

In experimental runs using Scenario 2, the experimental scenario specification for Scenario 2 in Table I-2 in Appendix I was followed. Scenario 3 runs used the Scenario 3 experimental scenario specification in Table I-3. The run logs for all experimental runs appear in Table J-2 and Table J-3 in Section J.2 of Appendix J.

6.4.3 Experimental Results

The parameters measured during the processing of each terminal report are listed in Section M.2 of Appendix M. Parameters measuring optimization performance and throttling operations are not applicable to Experiment 2 and were not collected.

In order to assess the *accuracy* of semantic service analysis in Aesop-MA, the two most important parameters to observe are *Session Priority* and *Snapshot Compliance*. When Aesop observes a new session, it sets the *Session Priority* using semantic rules. Session priority is important because it is used during analysis to find service expectation knowledge in the Knowledge Base. The *Snapshot Compliance* is the output parameter of the analysis operation for each snapshot and is determined semantically for each session metric reading using the queries as explained in Section 4.3.4.2.

The correctness of the session priority as deduced by Aesop is straightforward to check visually. The correct priorities by user and service are listed in Table 6-5, so the

priority set by Aesop can be verified by looking up the user and service for each test session on that table.

The accuracy of Aesop in determining the session compliance is somewhat more difficult to determine. Aesop monitors end user sessions by creating a session snapshot for each terminal report it receives and placing knowledge into the snapshot from that terminal report. It then analyses that snapshot to determine the compliance of the session with expectations for that snapshot. Therefore, the accuracy of both Aesop's monitoring operations and analysis operations must be assessed to determine how well compliance is calculated. A Java program was written which read fields from incoming terminal reports saved during experimental runs. To assess monitoring, the values of those fields were compared with concept individual properties on snapshots read from Aesop's Knowledge Base. The accuracy of the compliance value calculated semantically by Aesop for each snapshot in its analysis operation was assessed by comparing that value with a value calculated independently in the Java program using hard-coded service expectation values from the particular scenario for each experimental run.

To assess the *performance* of Aesop, *Monitoring Time*, *Analysis Time*, *CPU Usage* and *Memory Usage* are important. The *Monitoring Time* is the amount of time taken by Aesop to read knowledge from a single terminal report and place that knowledge into the Knowledge Base, and the *Analysis Time* is the time taken for an analysis operation to run over any amended snapshot buckets. Operation timings were parsed from Java logs recorded during experimental runs. The performance of those operations was analysed by applying statistical techniques to that timing information. *CPU usage* and *memory usage* were analysed using results collected by the Linux system statistics utilities.

The results of Experiment 2 appear in Section N.2 of Appendix N. Section N.2.1 summarises the session parameters for each session including the Session Priority. The results of an analysis of the accuracy of Aesop in monitoring terminal reports and analysing end user service compliance is presented in Section N.2.2. Section N.2.3 shows a statistical analysis of Aesop's monitoring time, analysis time, CPU usage and memory usage over the six independent runs of each scenario. Finally, Section N.2.4 contains a statistical analysis of the compliance of each service to expectations, again over the six independent runs of each scenario.

6.4.4 Analysis of Results

This section presents an analysis of the experimental results for Experiment 2. It examines the results that measured the accuracy of Aesop's monitoring and analysis operations and looks at the time and resource usage performance of those Aesop operations. It goes on to present and interpret the compliance of each session with expectations as calculated by Aesop for experimental runs using each scenario.

6.4.4.1 Accuracy of Aesop Monitoring and Analysis Operations

Aesop deduced the session priority correctly for all experimental sessions. The results shown in Table N-3 and Table N-4, when compared with the values in Table 6-5, show that the semantic rule shown in Listing C-1 of Appendix C derived session priority from the session user and type correctly in all sessions.

The *monitoring accuracy* of Aesop, as calculated using the off-line program, summarised over all experimental runs is shown in Table 6-8. The detailed results appear in Table N-5. The results show that Aesop demonstrated a perfect monitoring accuracy of 100% in experimental runs; Aesop correctly read the information from terminal reports into snapshot individuals for all snapshots. These results are evidence that Aesop can build and hold a Knowledge Base that accurately represents the service experience and context of running service sessions.

Table 6-8: Aesop Monitoring Accuracy

Terminal Reports	Missed	Monitoring Incorrect	Monitoring Accuracy
10777	3	0	100.00%

Aesop did not read 3 terminal reports into snapshots during monitoring. An analysis of the Aesop logs for the runs where terminal reports were missed was carried out. All missed terminal reports were caused by an `ArrayIndexOutOfBounds` exception thrown by Jena while adding terminal report knowledge to a snapshot model.

The *analysis accuracy* of Aesop, as calculated using the off-line program, summarised over all experimental runs is shown in Table 6-9. The detailed results appear in Table N-6. The results show that the analysis accuracy of Aesop was 100% in experimental runs; Aesop looked up the service expectations correctly for each snapshot and calculated the compliance of that snapshot with expectations in all cases where it performed an analysis. These results are evidence that Aesop can accurately determine the compliance of running sessions with expectations.

Table 6-9: Aesop Analysis Accuracy

Snapshots	Not Analysed	Analysis Incorrect	Analysis Accuracy
10466	27	0	100.00%

The Aesop analysis operation did not analyse snapshots in 27 cases because of a bug where Aesop's routine maintenance operation ran in parallel with the analysis operation, causing a concurrency exception. The zero values for those snapshots were not removed or corrected prior to analysis of session compliance because that analysis should use the compliance as calculated by Aesop. In any event, the absolute number of incorrect compliance values is too small to significantly affect statistical analysis of compliance levels.

In summary, Aesop demonstrated perfect accuracy on monitoring and analysis autonomic functions, indicating that the semantic lifting, queries, reasoning, and rules used to implement those functions perform correctly.

6.4.4.2 Performance of Aesop Monitoring and Analysis Operations

The performance of Aesop's monitoring and analysis operations were analysed statistically, the results of that analysis appear in Section N.2.3; see Table N-7 and Table N-8 for Scenario 2, Table N-11 and Table N-12 for Scenario 3. Plots of those results appear in Figure 6-19 and Figure 6-20. The results show that, for experimental runs in both scenarios, the observed performance for monitoring and analysis was well within the time window of 10 seconds each as specified in the experimental hypothesis.

Figure 6-19 shows plots of monitoring time averages and standard deviations for each minute of the six runs of Scenarios 2 and 3. Average monitoring times show a slow linear increase from 0.6 seconds when all sessions are started (minute 12 for Scenario 2, minute 9 for Scenario 3) to 1.2 seconds at the point at which sessions begin to terminate (minute 45 for Scenario 2, minute 48 for Scenario 3). Over this period, the standard deviation of monitoring time for both scenarios is low and stable, increasing from 0.1 seconds to 0.25 seconds. A small increase in standard deviation can be observed due to session initiation (minutes 9 to 12 for Scenario 2 and minutes 6 to 9 for Scenario 3) and session termination (minutes 45-51 for Scenario 2 and minutes 48 to 54 for Scenario 2). These values are very comfortably within the monitoring window time allocation of 10 seconds.

This slight increase in monitoring time average and standard deviation during experimental runs occurs because the monitoring process adds knowledge and references to the Dynamic Global Knowledge model of Aesop each time a terminal report is read. As explained in Section 4.4.4.4, the current Aesop Knowledge Base maintenance implementation does not carry out housekeeping on the Dynamic Global Knowledge model; as experimental runs progress, that model grows and the time taken to perform operations to the model increase. Such housekeeping, essential in a production system expected to stay up for long periods, is not critical to implement in an experimental system.

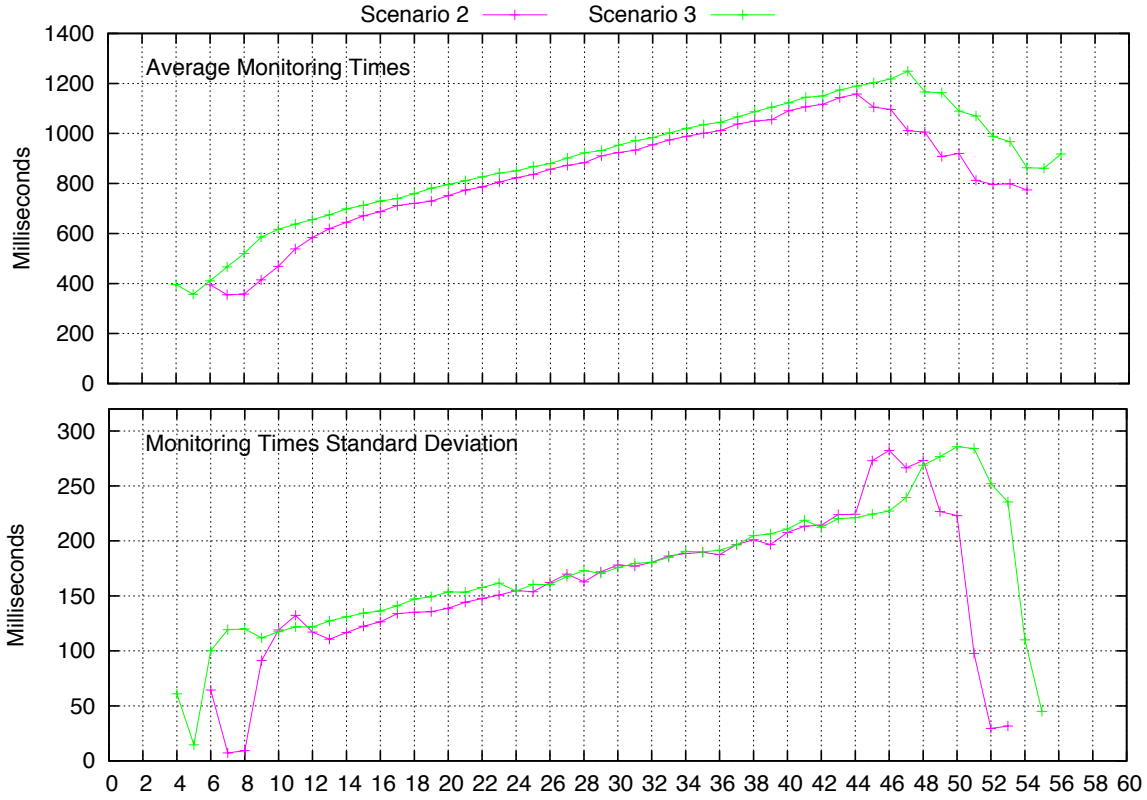


Figure 6-19: Monitoring Times Average and Standard Deviation

Figure 6-20 shows plots of analysis time averages and standard deviations for each minute of the six runs of Scenarios 2 and 3. The analysis times for both scenarios go from 0.2 seconds when all sessions are started to 0.4 seconds at 28 minutes. The analysis time holds at 0.4 seconds until sessions being to terminate where a small increase to 0.5 seconds is observed. The standard deviation is low, showing a slight increase from 40 milliseconds at the point when all sessions are running to 80 milliseconds at the point when sessions begin to terminate. In contrast to the smooth curve observed on monitoring times standard deviation, analysis times standard deviation shows fluctuations of 10 to 20 milliseconds. This pattern occurs because

monitoring operations are triggered on demand as terminal reports appear whereas analysis operations run periodically every 10 seconds.

The reason for the plateau trend observed on analysis times is because analysis primarily uses snapshot models and Aesop housekeeping clears snapshots, snapshot buckets, and snapshot models that are older than 10 minutes. After 10 minutes of operation with all sessions running, the number and size of snapshot models in the Knowledge Base begins to stabilize (minute 22 for Scenario 2, minute 19 for Scenario 3). By minute 28, analysis times for both scenarios has stabilized.

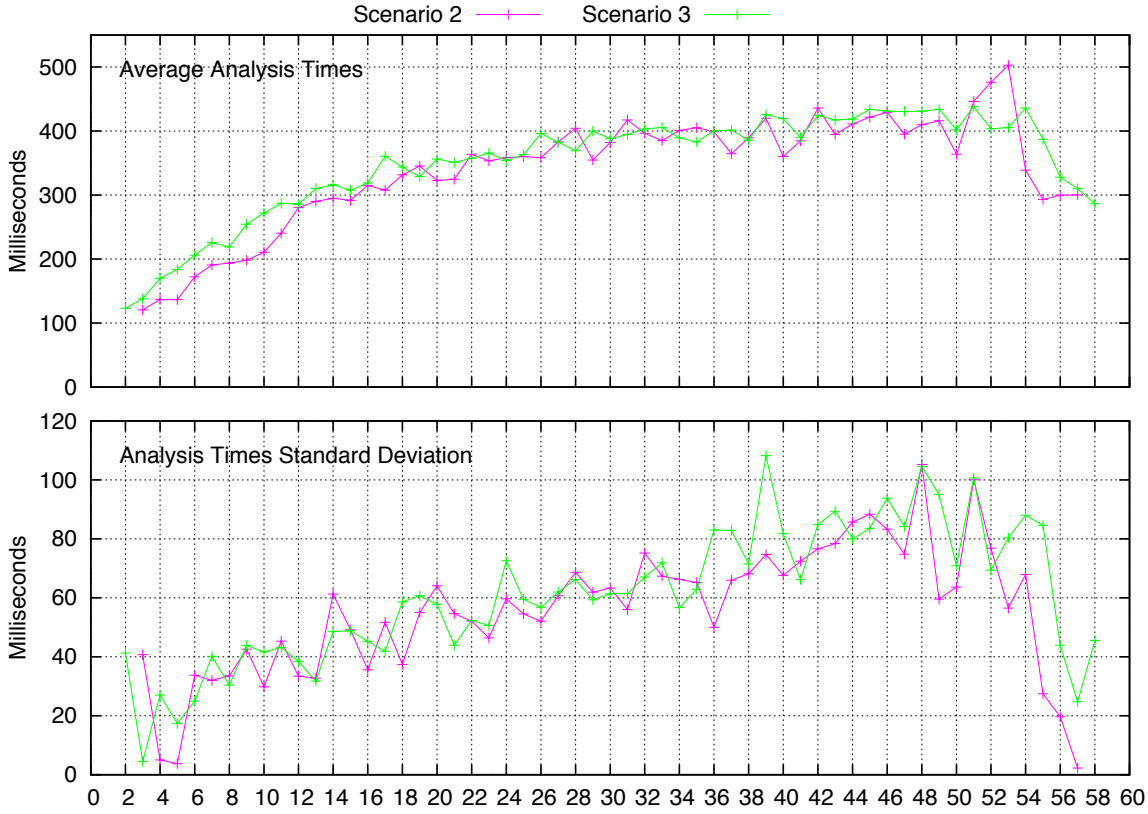


Figure 6-20: Analysis Times Average and Standard Deviation

CPU and memory usage were also analysed statistically, the results of that analysis also appear in Section N.2.3; see Table N-9 and Table N-10 for Scenario 2, Table N-13 and Table N-14 for Scenario 3. The results are plotted in Figure 6-21 and Figure 6-22. An examination of the results confirms that, for the experimental runs, Aesop did not use system resources excessively. CPU usage did not increase above 12% and memory usage remained below 5GB on the Home Gateway running Aesop.

The CPU usage plots for both scenarios are very similar. After some initial spikes when Aesop is started, CPU usage increases from 6% to 12% over the course of the experimental runs. The standard deviation on CPU usage is very low. CPU usage

shows the same slight increasing trend observed in monitoring time, with the increase in CPU usage also attributable to extra processing required to read and store the growing Dynamic Global Knowledge model in the Knowledge Base. The average CPU usage for Scenario 3 runs is marginally higher than for Scenario 2 because Scenario 3 runs start earlier and last longer, they start at minute 6 and continue to minute 54 whereas Scenario 2 runs start at minute 9 and continue to minute 51.

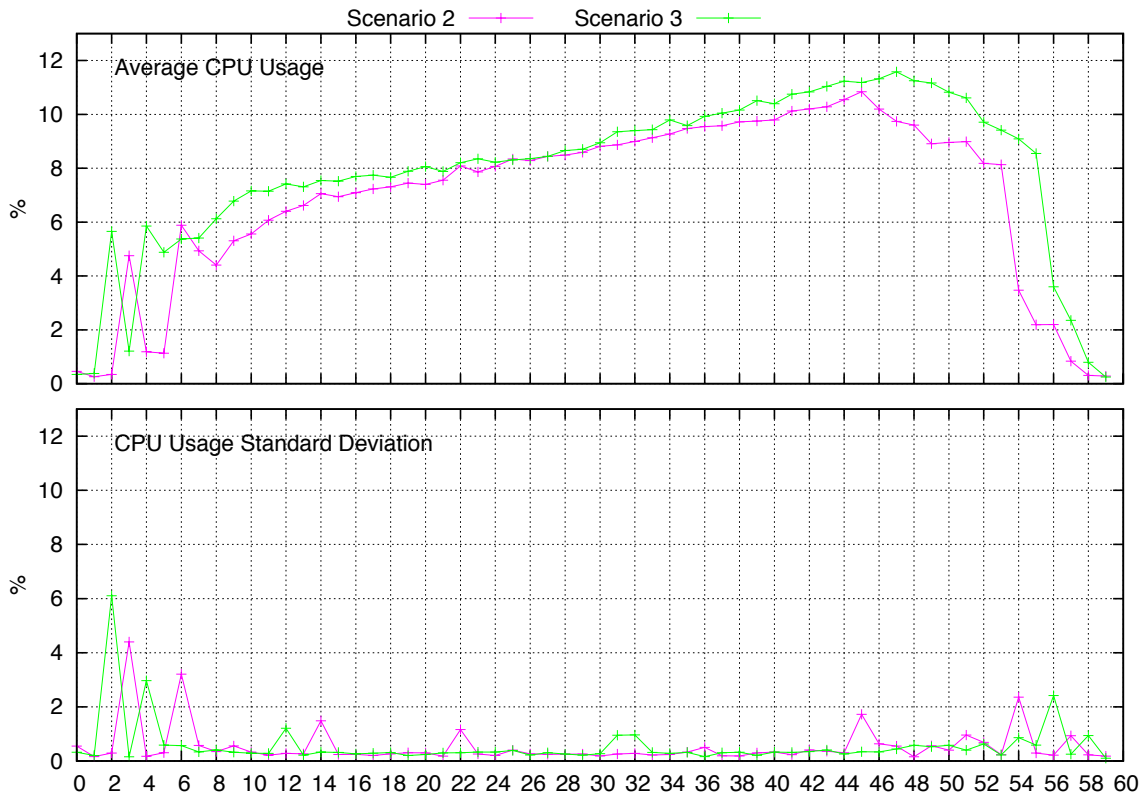


Figure 6-21: CPU Usage Average and Standard Deviation

Memory usage increases from 2GB at the beginning of experimental runs to 5GB at the end of runs. Memory usage starts to plateau as it approaches 5GB. The memory limit on the Java virtual machine was set at 6GB; as memory usage approaches that value garbage collection operates more aggressively in the JVM [Oracle, 2012]. Average memory usage, like average CPU usage is marginally higher for Scenario 3 runs than Scenario 2 runs because Scenario 3 runs are longer.

To summarise, the observed performance for monitoring and analysis autonomic functions in Aesop was fast enough to be considered near real time, well within the allocated time window of 10 seconds each. CPU and memory usage was well behaved during experimental runs.

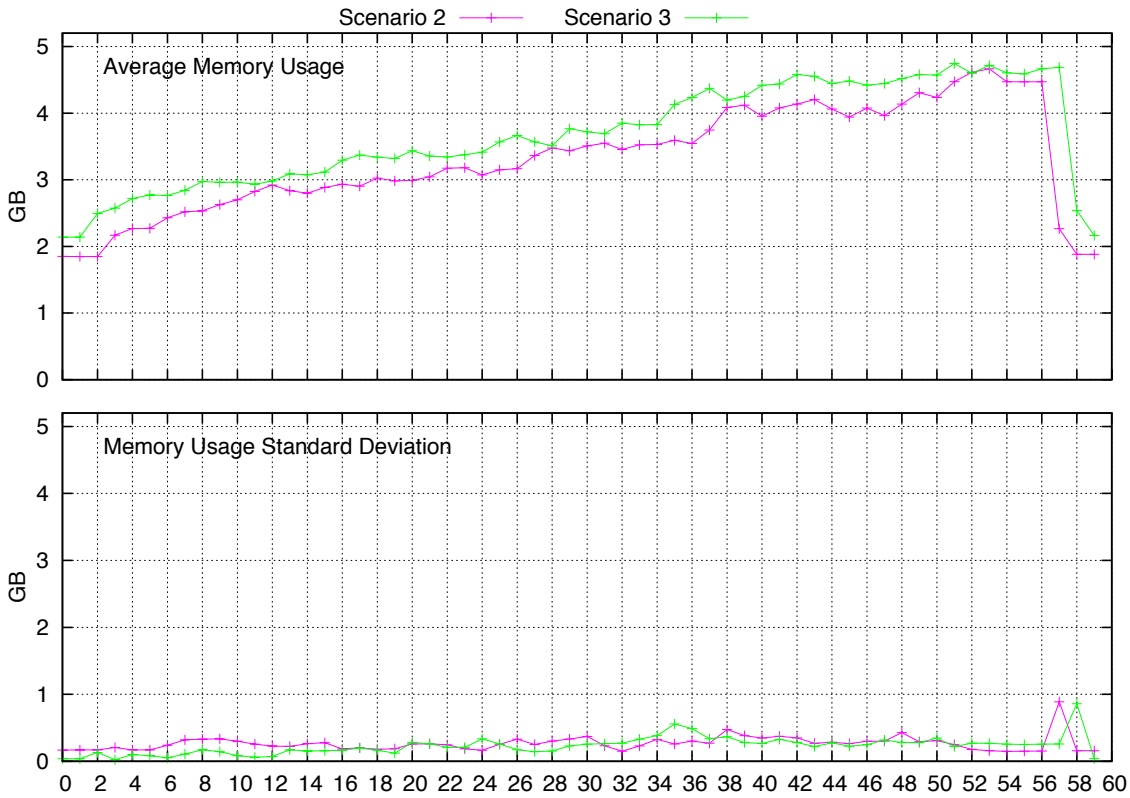


Figure 6-22: Memory Usage Average and Standard Deviation

6.4.4.3 Compliance of Sessions with Expectations

The compliance of each end user session with expectations as calculated by Aesop for each experimental run of Scenario 2 and Scenario 3 was analysed statistically. The results appear in Section N.2.4; see Table N-15 to Table N-18 for Scenario 2, Table N-19 to Table N-22 for Scenario 3. Plots of the results appear in Figure 6-23 and Figure 6-24.

From the perspective of addressing the experimental hypothesis, the actual value of compliance on each end user session is unimportant; what is important is that Aesop determines the compliance value for each snapshot accurately and with the required performance. However, it is interesting to examine how the compliance level of service sessions varied over the course of the experimental runs.

Figure 6-23 shows the average percentage compliance for each session in Scenario 2, the scenario where the Home Area Network broadband access network connection throughput is limited at 7Mbit/s down and 1Mbit/s up, see Table 6-2. The connection becomes congested as the number of end user services using it increase. The green vertical lines show the times at which sessions start and stop and are useful to show the influence of the start or stop of a session on other sessions.

In general, once all sessions are running, sessions executed in a constrained but stable manner. The compliance of each session with expectations did not vary greatly during the period when all sessions were running.

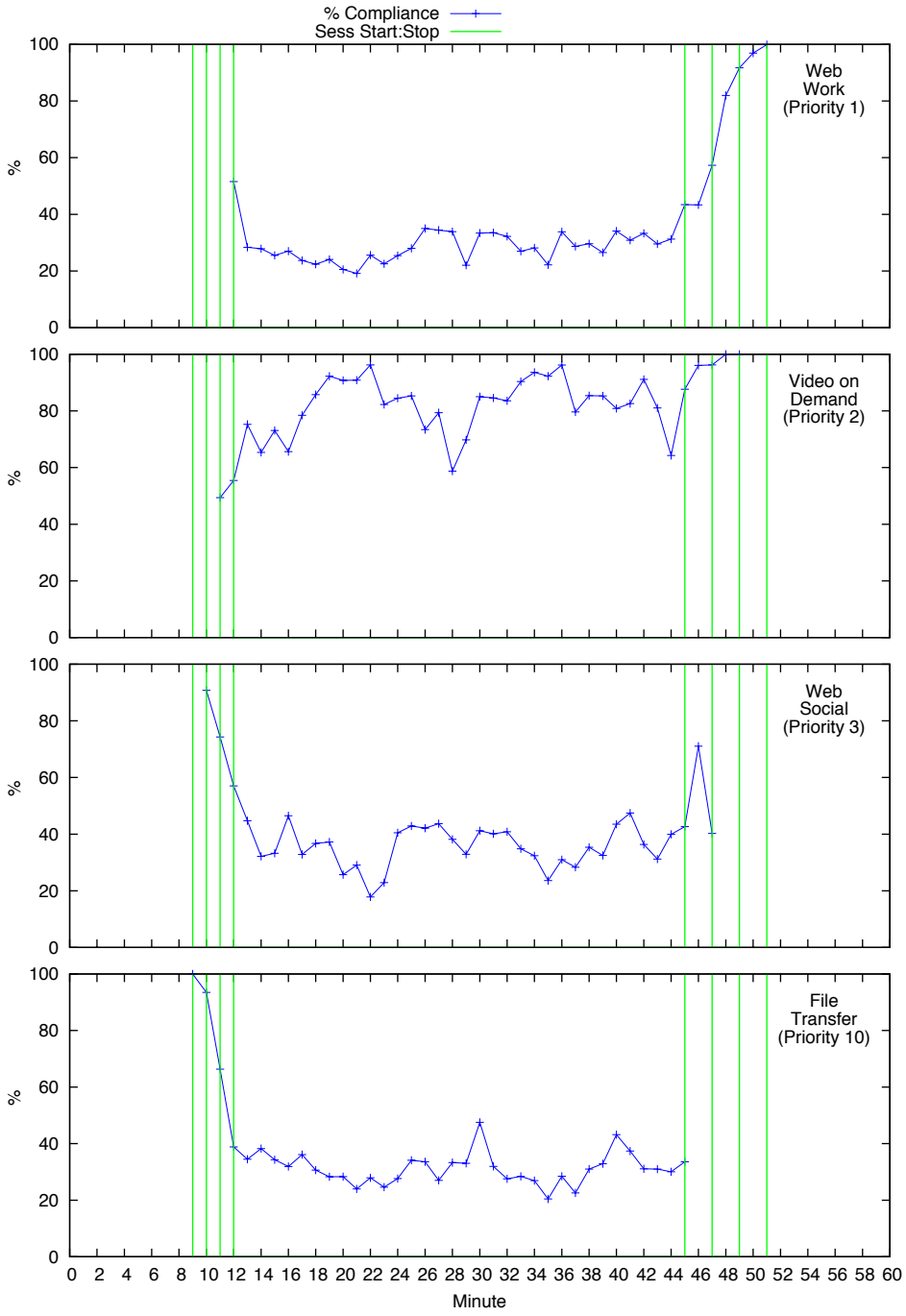


Figure 6-23 Average Session Compliance, Scenario 2

The highest priority Work Web Browsing session exhibits the lowest level of compliance, with compliance drops from an initial value of 50% to a value of approximately 30% in the first minutes. After that, it maintains the level of 30% until the other sessions begin to terminate, after which compliance increases to 100%.

The Video on Demand session with a priority of 2 shows the highest level of compliance, having an increasing compliance score during the first minutes of the session. Compliance stabilizes at a level of approximately 80% until the File Transfer and Social Networking Web Browsing sessions terminate, at which point compliance increases to 100%. The increasing compliance at session start occurs because, as explained in Section 5.4.2.2, the Video on Demand client caches incoming video for 1 second, which alleviates degradation due to congestion once the cache has filled.

The social networking web browsing session starts with a compliance level of 90%, which drops to approximately 30% as other sessions start. It fluctuates around 30% until the file transfer session terminates, after which an increase to 70% occurs, followed by a drop back to 40%. The standard deviation on the last point in that session is 39% (see Table N-17), indicating a large spread in the final readings for those sessions, probably caused by poor throughput on web requests being shut down by the browser.

The file transfer session initially has high compliance. As each other session starts, its compliance progressively drops to a level of around 30%. It remains at around that level until the session terminates.

Looking at the set of sessions from Scenario 2 on Figure 6-23 as a whole, it is clear that the highest priority Work Web Browsing session has the worst compliance level with expectations; such web sessions made up of large numbers of small requests are highly susceptible to congestion. The increase in compliance towards the end of Work Web Browsing sessions due to termination of other sessions indicates that allocating a more of the available bandwidth to these high priority sessions would increase their compliance.

The average percentage compliance of each session in Scenario 3 is shown in Figure 6-24. In Scenario 3, a progressive impairment on the throughput of the Home Area Network broadband access network connection occurs as shown in Table 6-3. The throughput on the connection decreases by 80% and returns to its normal value over the course of experimental runs. The green stepped line on the plots in Figure 6-24 shows the progression of the impairment during runs and illustrates the influence of the impairment on the compliance of each session with expectations.

The plots of compliance on all four sessions track the impairment increase and decrease quite closely. All four sessions show near perfect compliance until the

impairment begins at 12 minutes into experimental runs, and show recovery to near perfect compliance after the impairment finishes at 42 minutes. It is interesting to note that all four sessions cope well with the first two reductions in bandwidth, the compliance level of all sessions recovers with three sessions recovering to a compliance level of near 100%. Once the third reduction in bandwidth occurs 18 minutes into the runs, the compliance level of all sessions trend downwards until the minimum bandwidth level is reached at 24 minutes.

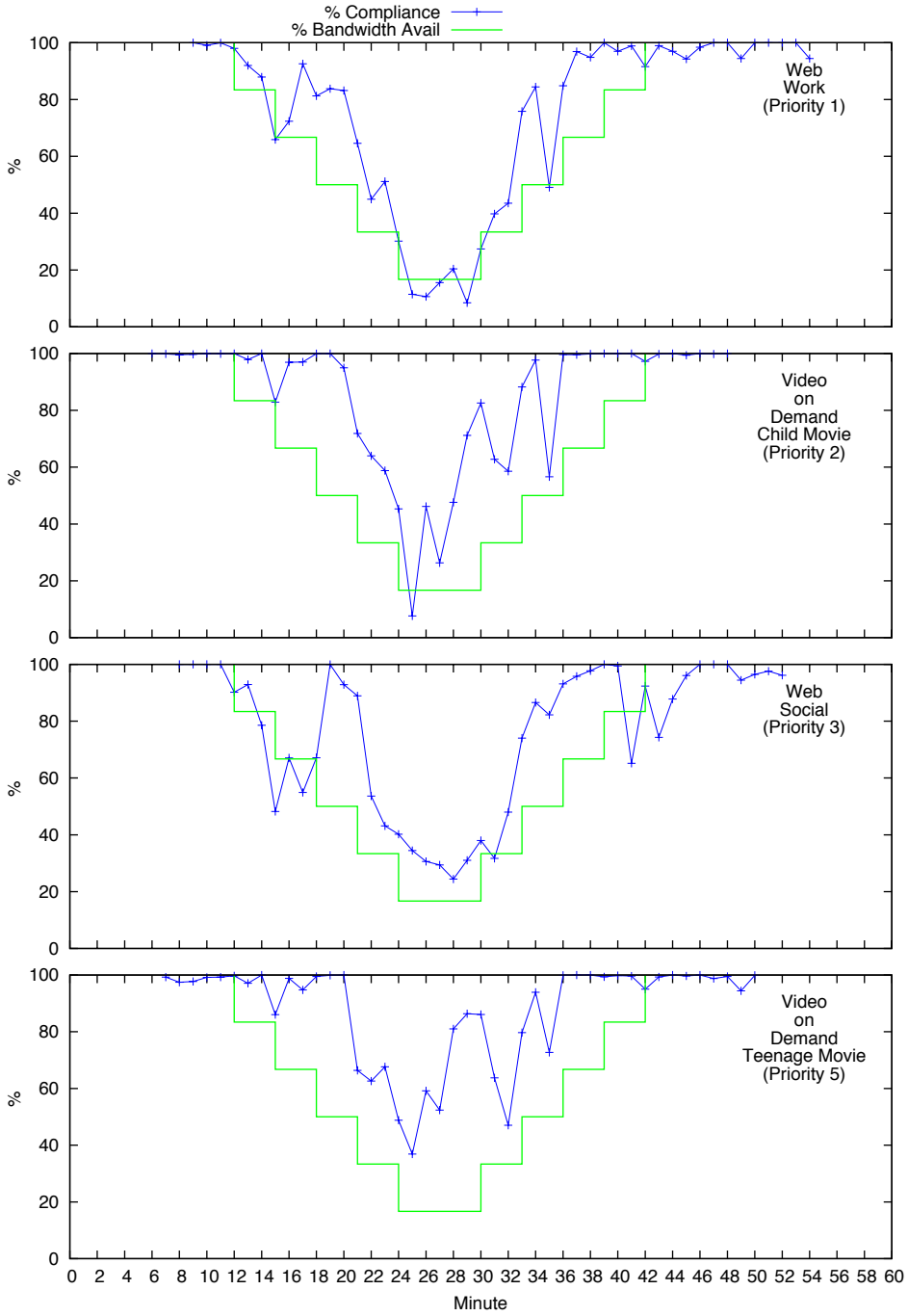


Figure 6-24: Average Session Compliance, Scenario 3

The stepwise recovery observable on Video on Demand sessions is due to the algorithm used by SECCO to calculate the MOS values based on the log from a Video on Demand client. When a freeze occurs, the Video on Demand client issues a freeze report, followed by a number of errors and warnings as it tries to recover the session over the next number of seconds. SECCO weights the freeze report very heavily and the subsequent error and warning messages less heavily, which causes an initial low MOS value followed by better MOS values. Subsequent freezes cause the MOS value to drop again followed by an improvement.

In this scenario, as in Scenario 2, the highest priority Work Web Browsing session is the session with the worst compliance level, further demonstrating that such web browsing sessions are sensitive to throughput restrictions in the network.

6.4.5 Summary

Evaluation of the semantic monitoring and analysis functions of Aesop was the primary focus of Experiment 2, to assess the extent to which monitoring and analysis could be undertaken in near real time.

The experiment was carried out using Scenario 2, the congestion scenario and Scenario 3, the impairment scenario. Experimental runs were executed on a Home Area Network test bed with Aesop running on the Home Gateway, and SECCO executing sessions on terminals and forwarding terminal reports to Aesop. The results demonstrated that Aesop accurately analyses service experience against service expectations and does so in a time period of less than 20 seconds from reception of a terminal report, supporting the experimental hypothesis.

The accuracy of Aesop was assessed by checking how well Aesop deduced the priority of each session, monitored end user sessions by reading information into snapshots in the Knowledge Base, and analysed those snapshots to determine their compliance with expectations. The results show that Aesop deduced the session priority correctly for all sessions. Aesop monitoring read the information from terminal reports into the Knowledge Base with 100% accuracy. Aesop calculated compliance correctly on snapshots it processed with 100% accuracy.

The results from experimental runs show that Aesop performs well. The average monitoring times remain below 1.3 seconds with a standard deviation of 0.2 seconds, and average analysis times remain below 0.5 seconds with a standard deviation of 0.1 seconds. These values are very comfortably under the 10-second time budgets

allocated for report monitoring and analysis. Average CPU usage did not increase above 12%, and average memory usage did not exceed 5GB during runs. A slight increasing trend in CPU usage is due to the increasing size of the Dynamic Global Knowledge Model in experimental runs.

The main purpose of Experiment 2 was to evaluate the accuracy and performance of Aesop in determining end user session compliance. Although not central to that purpose, the actual observed compliance of sessions with expectations in experimental runs is interesting to examine, as it is a major feature that Aesop brings to users. In experimental runs of the Scenario 2 congestion scenario, the average compliance of each session with expectations remained stable once all sessions had commenced. The Work Web sessions exhibited the lowest compliance with expectations, and the Video on Demand sessions exhibited the highest compliance. In the experimental runs of the Scenario 3 impairment scenario, compliance with expectations mirrored the progression of the impairment closely. All sessions recovered well once the network impairment disappeared. Again, the Work Web sessions exhibited the lowest level of compliance with expectations and the Video on Demand sessions exhibited higher compliance with expectations.

6.5 Experiment 3: Semantic Optimization of Services towards Expectations

The purpose of Experiment 3 was to evaluate Aesop as a full autonomic system with all four autonomic functions running. The goal of the experimentation was to evaluate the extent to which the usage of network resources could be tuned to optimize service delivery of a set of services in near real time. The experiment was designed as an extension of Experiment 2 for optimization. During experimentation, the deduced optimizations and the performance of optimization planning and execution operations were recorded in addition to the compliance metrics, monitoring and analysis times, and original terminal reports which were also collected in Experiment 2 experimental runs. All experimental conditions for Experiment 3 were set up to be identical for equivalent runs in Experiment 2 to allow comparison of the results of Experiment 2 and Experiment 3 when assessing how well Aesop optimized end user service sessions.

The analysis of results assessed the *effectiveness* and *consistency* of Aesop in optimizing service delivery of a set of services. Effectiveness is the evaluation

parameter that shows how well Aesop is optimizing service sessions; it measures the degree of change in compliance with expectations of high priority services due to optimization. *Consistency* is an indication of the level of stability of Aesop's optimization decisions; it measures the uniformity with which Aesop optimized a set of sessions in different experimental runs. The *accuracy* and *performance* of Aesop's autonomic functions were examined in order to assess those evaluation parameters for each function in the full system.

The *effectiveness* of optimization planning and execution in Aesop was determined by comparing the average compliance level observed on non-throttled sessions from Experiment 3 with the average compliance level observed on the equivalent sessions from Experiment 2 for each minute in which optimization actions were applied in the network. Results from the two experiments are comparable because experimental setup, scenarios, and conditions were identical for equivalent runs in both experiments. The only difference between two equivalent runs was that optimization was inactive in Experiment 2 and was active in Experiment 3. Consistency was analysed by comparing the throttling operations that were applied and the time at which Aesop applied them in each of the six experimental runs for each scenario.

The *accuracy* of Aesop's optimization planning and execution algorithm was examined by checking the correctness of application and release of throttling actions in the log produced by Aesop. The off-line Java program used in Experiment 2 was extended for this purpose; its monitoring and analysis accuracy calculations were also run on the results of Experiment 3. Statistical techniques were used to analyse the *performance* of Aesop's monitoring, analysis, and optimization planning and execution operations, in the same manner as for Experiment 2. The analysis and performance results from Experiment 3 were compared with those obtained in Experiment 2.

6.5.1 Hypothesis

That semantic service optimization can prioritize the most important service sessions in a Home Area Network at the expense of less important service sessions in a time window of 30 seconds, composed of a 10 second window for monitoring*, a 10 second window for analysis, and a 10 second window for optimization planning and execution.*

6.5.2 Experimental Setup and Execution

The experiment was set up to provide evidence to test the experimental hypothesis; that Aesop can prioritize the most important sessions correctly and can perform the prioritization in the 30 second time window. As experimental runs used a full Aesop system as shown in Figure 4-1, the experimental results provide measurements on the effectiveness, consistency, accuracy, and performance of Aesop as a full End User Service Management system.

As in Experiment 2, all experimentation was carried out on the HAN test bed environment described in Section 6.2.5. The two scenarios used in Experiment 2 were re-used; the congestion scenario described in Section 6.2.3.2 and the impairment scenario described in Section 6.2.3.3. Six independent experimental runs were conducted for each scenario in order to obtain enough measurements for statistical analysis as discussed in Section 6.2.6, giving a total of 12 experimental runs.

In experimental runs using Scenario 2, the experimental scenario specification for Scenario 2 in Table I-2 in Appendix I was followed, as in experimental runs for Scenario 2 in Experiment 2. Scenario 3 runs used the Scenario 3 experimental scenario specification in Table I-3, the same scenario specification as was used for experimental runs of Scenario 3 in Experiment 2. The run logs for all experimental runs appear in Table J-4 and Table J-5 in Section J.3 of Appendix J.

6.5.3 Experimental Results

The parameters measured during the processing of each terminal report are listed in Section M.2 of Appendix M.

The *effectiveness* of Aesop in optimizing the compliance of the set of end user service sessions was determined by comparing the results of Scenario 1 and Scenario 2 runs in Experiment 3 with runs of the same scenario in Experiment 2 for each minute of experimental runs. The algorithm described in pseudo code in Listing 6-1 was designed to determine effectiveness of optimization by performing that comparison. The algorithm is implemented as a SQL query, shown in Listing L-3 of Appendix L.

```
double  avg_sess_comp_before[#minutes][#sessions]
double  avg_sess_comp_after [#minutes][#sessions]
boolean sess_throttled      [#minutes][#sessions]

for each minute
  for each session
    avg_sess_comp_before = avg(Experiment 2 compliance values: minute, session)
    avg_sess_comp_after  = avg(Experiment 2 compliance values: minute, session)
```

```
    session_throttled    = (#metrics with session throttled) >= (#metrics / 2)
  end for
end for

double avg_comp_before[#minutes]
double avg_comp_after [#minutes]

for each minute
  int    session_count = 0
  double total_comp_before
  double total_comp_after

  for each session
    if (!session_throttled[minute][session])
      total_comp_before = avg_sess_comp_before[minute][session]
      total_comp_after  = avg_sess_comp_after [minute][session]
      session_count++
    end if
  end for

  avg_comp_before = total_comp_before / session_count
  avg_comp_after  = total_comp_after  / session_count
end for

effectiveness = (avg_comp_after - avg_comp_before) * 100 / avg_comp_before;
```

Listing 6-1: Effectiveness Algorithm Pseudo Code

The first loop of the algorithm in Listing 6-1 calculates average compliance for each session for each minute for Experiments 2 and 3, the "before" and "after" values. It determines if a session was throttled in a minute by checking how many metric readings for that minute reported throttling as active. If more than half of the metric readings reported throttling as active, the session was set as throttled in that minute. The second loop of the algorithm calculates the overall average compliance for all non-throttled sessions from Experiment 3 for each session and also the overall compliance of the equivalent sessions in Experiment 2. In this way, only optimized (non-throttled) sessions are considered in the effectiveness calculation. The final line of the algorithm calculates effectiveness as the percentage difference of compliance after optimization as compared to compliance before optimization.

The *consistency* of the optimization planning and execution operations of Aesop were analysed by visually examining the order in which Aesop applied and removed throttling actions in each experimental run of Scenario 2 and 3, and manually undertaking a five-number summary [Moore et al., 2009] of the spread of times at which Aesop applied and removed those throttling actions. Updating the off-line Java program to programmatically analyse consistency proved too complex because the semantic Optimization Planning and Execution algorithm described in Section 4.3.4.3 uses many variables during execution and has inherent delays built into it. For example, optimization backs off when new and deleted sessions are identified and when throttling actions are applied and removed.

The monitoring, analysis, and optimization *accuracy* of Aesop was assessed by comparing the results produced by Aesop with those produced by an independent Java program, as discussed in Section 6.4.3. Aesop implements optimization planning and execution autonomic functions in a single operation. The optimization accuracy is the accuracy of that operation in determining what throttling application or release action should be applied to the network at a given stage in either of the experimental scenarios used. The throttling actions selected and applied or released by Aesop at each scenario stage were compared with a hard-coded table, derived from the Scenario descriptions in Appendix I, of what optimization should occur at that stage of the scenario. The Java program used in Experiment 2 for analysing the accuracy of monitoring and analysis was extended to perform this optimization accuracy check.

In Experiment 3, the *Optimization Time*, the time taken for an optimization planning and execution operation to complete is an important *performance* parameter and was analysed. As in Experiment 2, *Monitoring Time*, *Analysis Time*, *CPU Usage* and *Memory Usage* are also important and were analysed, with their Experiment 3 results compared to the results from Experiment 2 to determine the effect of optimization planning and execution operations on those parameters.

The results of Experiment 3 appear in Section N.3 of Appendix N. The tables of Section N.3.1 show the optimization effectiveness of Aesop for Scenario 2 and 3 as determined by the algorithm in Listing 6-1, as well as a statistical analysis of the compliance of each session type from Scenario 2 and 3. The tables in Section N.3.2 show the consistency of Aesop, detailing the optimization throttling actions executed on sessions during experimental runs and showing the spread of throttling actions as applied and released by Aesop during optimization. Section N.3.3 summarises the session parameters for each session including the Session Priority. An analysis of the accuracy of Aesop in monitoring knowledge, analysing snapshot compliance, and optimizing end user sessions is presented in Section N.3.4. Section N.3.5 shows a statistical analysis of Aesop's monitoring, analysis, and optimization times, CPU usage and memory usage over the six independent runs of each scenario.

6.5.4 Analysis of Results

This section examines the experimental results for Experiment 3. It presents an analysis of the effectiveness of Aesop's optimization planning and execution operations and how well optimization works for each session type in each scenario,

the primary focus of Experiment 3. It goes on to examine the consistency with which Aesop applies optimization over the experimental runs of Experiment 3. The accuracy of Aesop's monitoring, analysis, and optimization planning and execution operations are analysed, as is the time and resource usage performance of those operations.

6.5.4.1 Effectiveness of Aesop in Optimizing the Running Service Session Set

The effectiveness of Aesop's optimization planning and execution operations was determined using the algorithm in Listing 6-1 to compare the compliance of unthrottled sessions in Experiment 3 with the equivalent sessions from Experiment 2 for each minute where optimization was applied. The detailed results appear in Table N-23 and Table N-24 of Section N.3.1 of Appendix N. The results are summarised in Table 6-10.

The results in Table 6-10 show that Aesop optimization is effective in improving the compliance of high priority sessions, particularly when such sessions would otherwise show high levels of non-compliance. Aesop improved the average compliance of unthrottled sessions in Experiment 3 over the same sessions in Experiment 2 in both scenarios. The effectiveness of Aesop in Scenario 2 experimental runs was much higher than in Scenario 3 runs; in Scenario 2 compliance was more than doubled whereas in Scenario 3, compliance increased by 13%. When sessions where the original compliance was below 50% are examined, the effectiveness of optimization is higher in both Scenario 2 and Scenario 3, showing a near tripling of compliance in Scenario 2 and improving to 60% for Scenario 3.

Table 6-10: Aesop Optimization Average Effectiveness

Scenario	Effectiveness %	
	All Unthrottled Sessions	Unthrottled Sessions, Compliance < 50%
Scenario 2	136.500	175.957
Scenario 3	13.201	60.428

The plots of effectiveness by minute in Figure 6-25 shows the effectiveness of Aesop during runs of Scenario 2 and 3. The red line is a plot of effectiveness per minute while throttling is active. The green vertical lines in the Scenario 2 plot indicate session start and stop times. The steeped green line in the Scenario 3 plot indicates the progression of network impairment in that scenario. The plots illustrate how optimization works better in minutes where resources are constrained the most.

In Scenario 2, the Home Area Network is constrained between minutes 12 and 48; in Scenario 3, the network impairment is at its height between minutes 18 and 36.

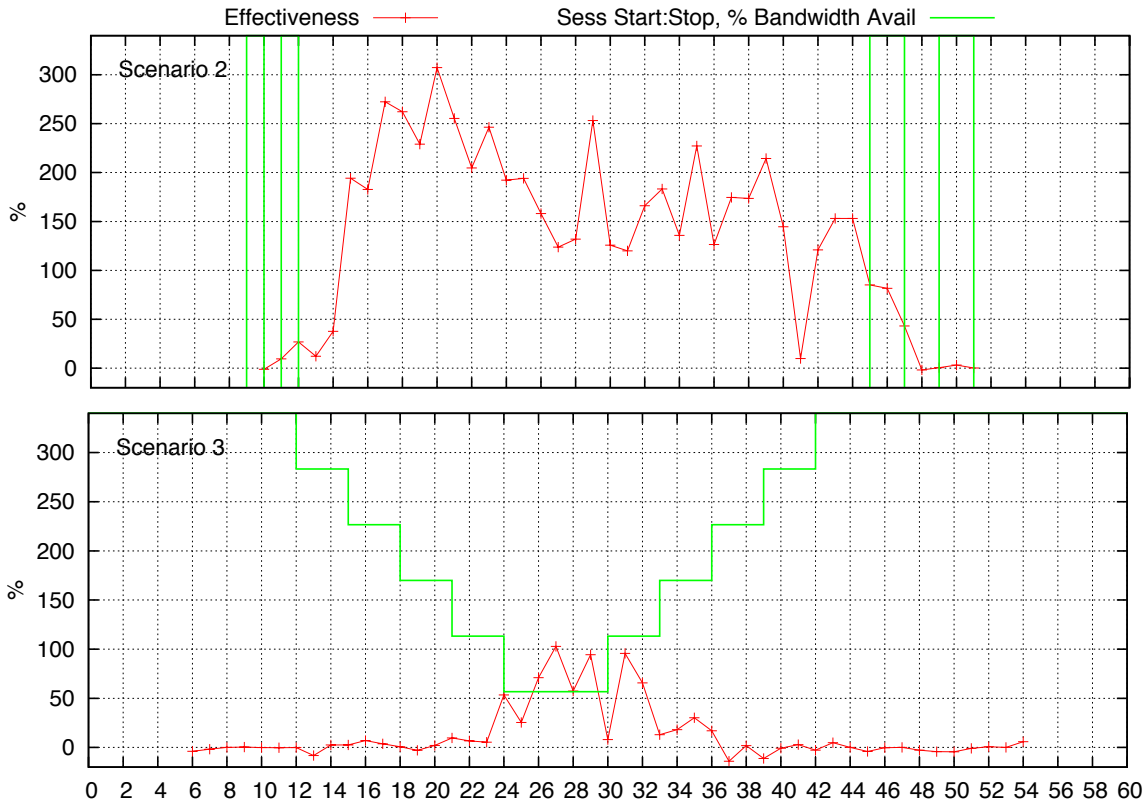


Figure 6-25: Effectiveness of Aesop Optimization, Scenarios 2 and 3

Figure 6-26 and Figure 6-27 show the difference in average compliance for each session type in Scenario 2 and Scenario 3 in Experiments 2 and 3, equivalent to Figure 6-23 and Figure 6-24 for Experiment 2.

In Figure 6-26, the blue line is the average compliance level for each session type before optimization (Experiment 2), and the red line is the average compliance level after optimization (Experiment 3). The green vertical lines indicate the point at which sessions started and stopped in the scenario. The pink box plots show the spread of throttling; the left box plot shows the spread of throttling application on each session, the right box plot shows the spread of throttling removal. Throttling consistency will be discussed further in Section 6.5.4.2.

The plots in Figure 6-26 show that Aesop is very effective as an optimization system in this scenario. The compliance level of the highest priority work web browsing session is improved from an average of 30% to an average of 80% by optimization, going from having the lowest level of compliance to the highest.

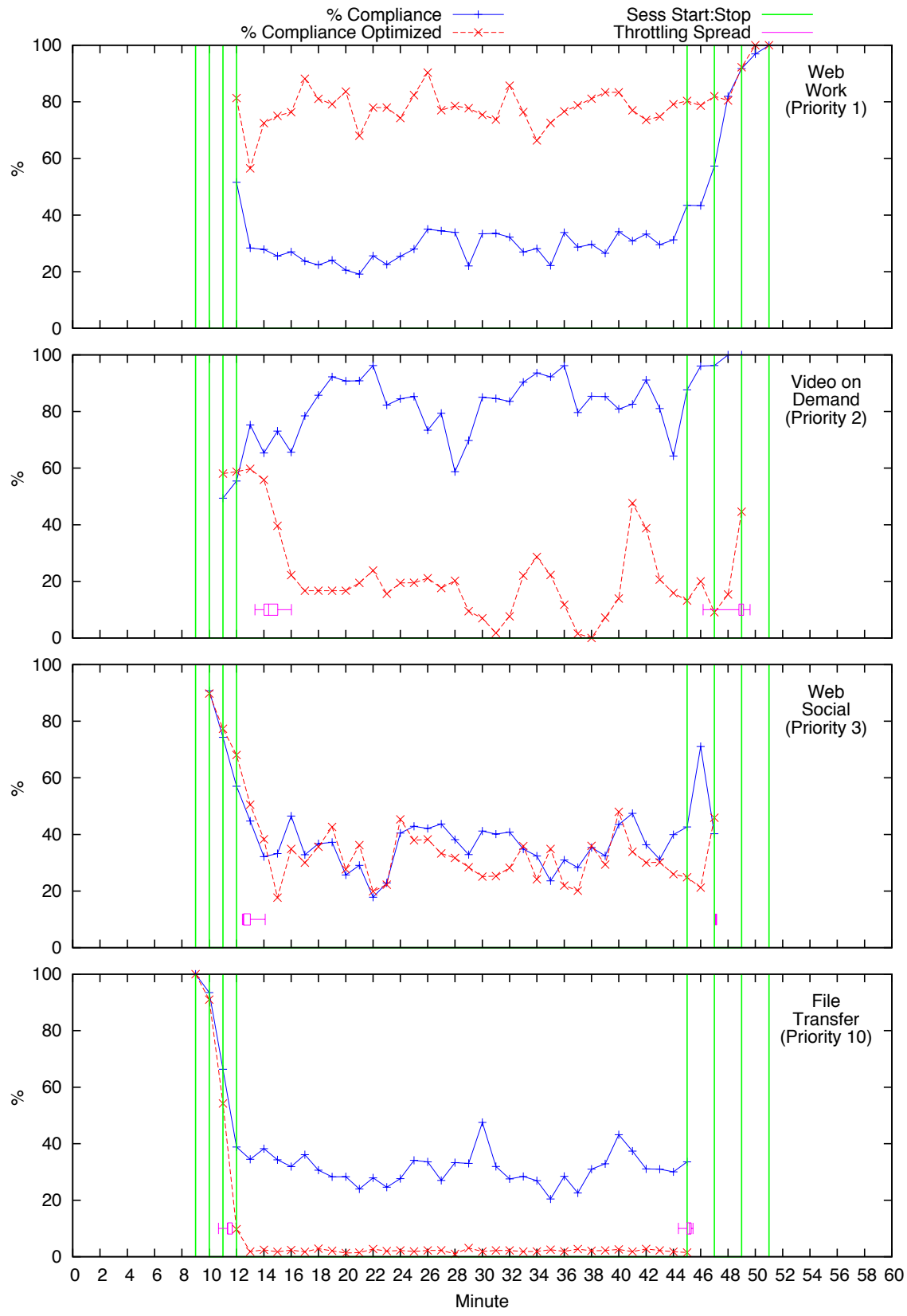


Figure 6-26: Average Session Compliance before and after Optimization, Scenario 2

Video on Demand sessions, which had the highest level of compliance in Experiment 2, exhibit the largest decrease in compliance from a level of 80% to 20% as a result of optimization. The peaks at minute 34 and 42 are caused by restarts of the Video on Demand client during some runs, which caused throttling to be released

and re-applied on those runs. Corresponding slight drops in compliance on the work web browsing session compliance for the same minutes can be observed.

Social networking web browsing session shows a drop in compliance of 5% to 10%, much less than the higher priority video on demand sessions, indicating that a more severe throttling value than the value in Table 6-5 should be set for such web browsing sessions to reduce their throttled bandwidth consumption. Compliance of file transfer sessions drops to near zero, indicating that throttling applied to those sessions is effective in releasing bandwidth for higher priority sessions.

In Figure 6-27 as in Figure 6-26, the blue line is the average compliance level for each session type before optimization (Experiment 2), and the red line is the average compliance level after optimization (Experiment 3). The green stepped line shows the progression of network impairment, with the line showing the percentage of the original 18Mbit/s of bandwidth available at each minute. The pink box plots show the spread of throttling; the left box plot shows the spread of throttling application on each session, the right box plot shows the spread of throttling removal. Throttling consistency will be discussed further in Section 6.5.4.2.

Figure 6-27 shows that Aesop was effective in improving the compliance of the highest priority work web browsing session, more than doubling compliance for some minutes at the peak of the network impairment and giving a significant improvement to compliance of this highest priority session during this period. However, because the overall level of bandwidth available on the access network connection at the peak of the network impairment is low, even this high priority session has compliance levels of between 20% and 40% during this period.

Compliance of priority 2 video on demand sessions drop below Experiment 2 values from the point at which throttling is applied (minutes 22 and 24) to the point at which throttling is restored (minutes 34 and 37). Compliance initially drops by 5% to 10%, dropping by 20% to 30% for later minutes. The stepwise recovery on these video on demand sessions caused by the manner in which SECCO calculates MOS values discussed in Section 6.4.4.3 is also observable in the Experiment 3 plot.

Social networking web browsing sessions also exhibit drops in compliance between throttling being applied at minutes 14 to 16 to being removed at minutes 36 to 39. Again, the drop in compliance is less than 10% initially, but the drop increases to levels of 50% at later minutes.

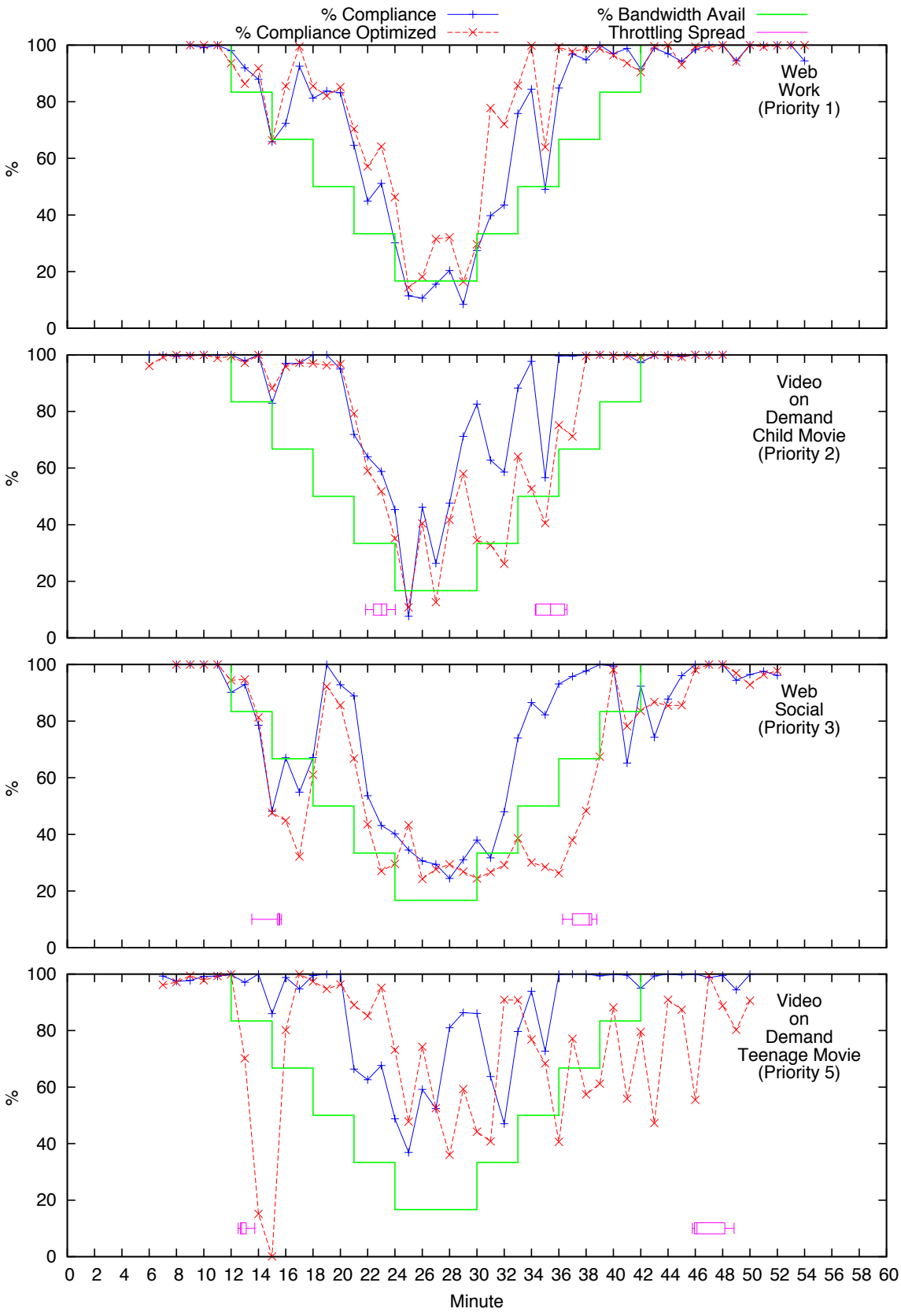


Figure 6-27: Average Session Compliance before and after Optimization, Scenario 3

The video on demand priority 5 sessions exhibit better average compliance in Experiment 3 than Experiment 2 for large parts of their runs. This is because the level of throttling of 1Mbit/s applied to such sessions as specified in Table 6-5 is not

enough to prevent these sessions continuing to consume significant bandwidth. After an initial large drop in compliance to near zero when throttling is applied between minutes 12 and 14, the sessions recover to near full compliance when bandwidth becomes available after minute 16 when throttling is applied to the priority 3 social networking web browsing sessions. From then compliance begins to drop as the network impairment progresses, but improves again when more bandwidth becomes available at minute 24 when throttling is applied to the priority 2 video on demand sessions. Between minute 28 and 31, at the period of most impairment, compliance does drop below levels of Experiment 2 as the highest priority unthrottled work web browsing session takes the available bandwidth. As more bandwidth becomes available, priority 5 Video on Demand session again consume that bandwidth and compliance improves above Experiment 2 levels. When throttling is released on priority 2 video on demand sessions at minute 34 and social networking web browsing sessions at minute 36, those sessions consume more of the available bandwidth, and the compliance of priority 5 video on demand sessions drops below Experiment 2 levels for the remainder of their runs. The stepwise recovery on the sessions is again caused by the manner in which SECCO calculates MOS values.

In summary, Aesop improved the compliance of high priority sessions, particularly when the sessions demonstrated low compliance levels when not optimized. When the issue of degrading compliance was local to the home area network Aesop resolved the issue; when the issue was external to the home area network, Aesop reduced the impact of the issue. The compliance of high priority Web Browsing sessions increases at the expense of File Transfer and Video on Demand sessions when throttling parameters are correctly tuned on those sessions. Web browsing sessions show smaller reductions in compliance when throttled.

6.5.4.2 Consistency of Application and Release of Optimization Throttling Actions

The consistency of Aesop's optimization algorithm was analysed by examining the uniformity with which throttling actions were applied and released in experimental runs and spread of throttling application and release in the experimental runs for the two scenarios in Experiment 3. The throttling actions applied and removed by Aesop appear in Table N-33 and Table N-34 in Section N.3.2 of Appendix N. The spread of throttling actions the start and end of Scenario 2 runs are detailed in Table N-35 and Table N-36. Scenario 3 throttling action spreads during the impairment increase

period are shown in Table N-37, and those during the impairment decrease period are shown in Table N-38.

Figure 6-28 and Figure 6-29 present plots of the first application and final removal of throttling actions for each session in each run of Scenario 2 and 3 respectively, the data that appears in Table N-33 and Table N-34. This type of plot was used because they illustrate the progression of throttling for each run as horizontal lines together with the external factor that is influencing throttling as vertical or stepped lines.

Figure 6-28 and Figure 6-29 illustrate the consistency of Aesop's optimization planning and execution operations. They show that throttling operations for optimization execution are applied and released in a uniform manner in each of the experimental runs. The throttling operations on low priority sessions in Scenario 2 are removed when those sessions terminate. The throttling operations for Scenario 3 in Figure 6-29 exhibit a skew to the right when compared to the progression of the network impairment as illustrated with the stepped green line. The reason for this skew is that Aesop waits a predefined interval when it detects changes in compliance before reacting, causing this skewing effect.

As can be seen in Figure 6-28, in Scenario 2 the web browser froze at the beginning of run 4 causing extra bandwidth to be available and delaying the start of throttling on the Video on Demand session. In the same run the Aesop terminated early due to an inconsistent ontology caused by a duplicate terminal report sent by SECCO. The results of Scenario 2 run 4 were eliminated from the data in Table N-35 and Table N-36 used to plot the box plots in Figure 6-26. The file transfer session was terminated one minute early in run 6 of Scenario 2.

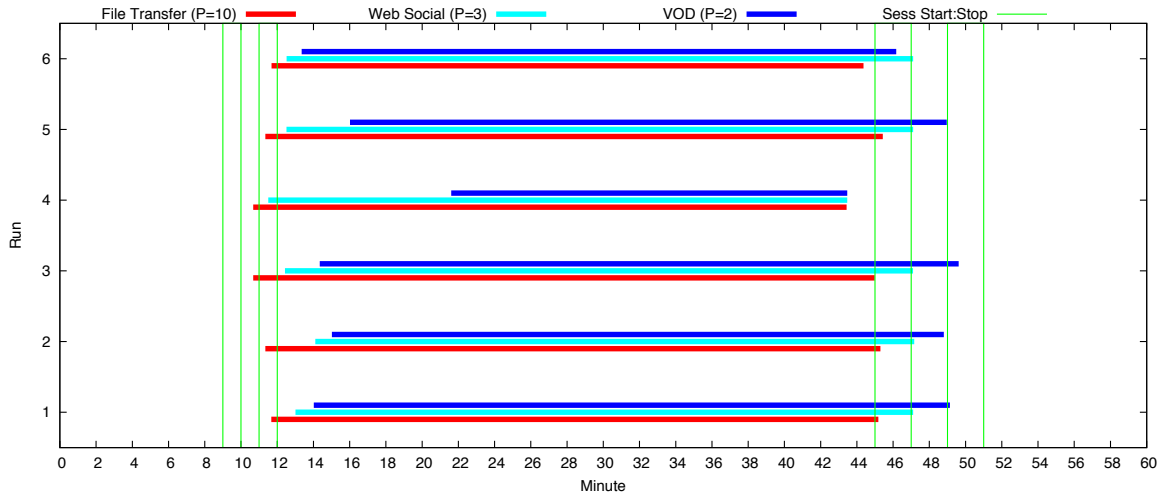


Figure 6-28: Throttling on Scenario 2 Sessions

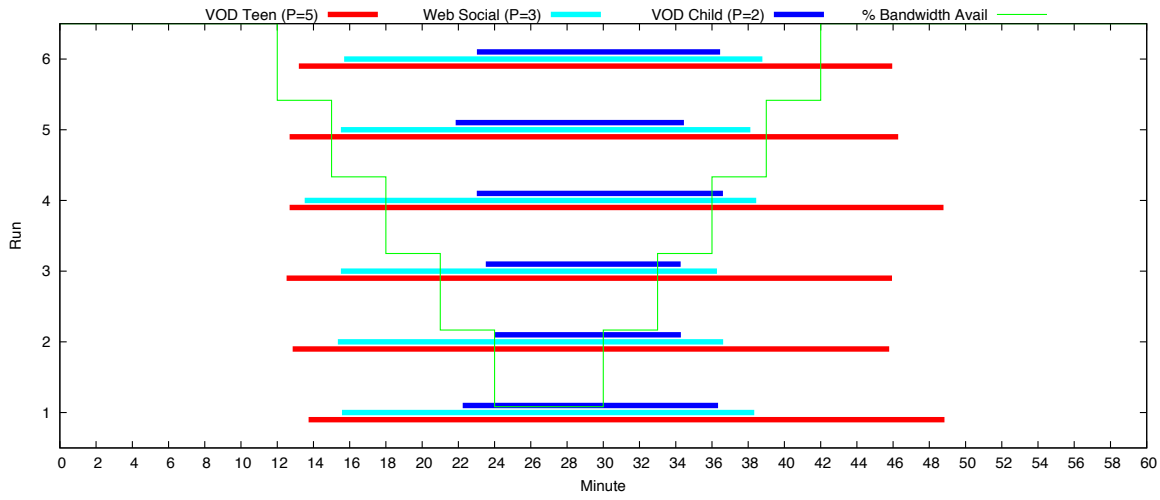


Figure 6-29: Throttling on Scenario 3 Sessions

Five-number summaries were used to examine the centre and spread of throttling actions on sessions during runs because the number of measurements was low. The throttling spreads for Scenario 2 are presented in Table N-35 and Table N-36; those for Scenario 3 are presented in Table N-37 and Table N-38. Table 6-11 and Table 6-12 show the range of throttling actions for Scenarios 2 and 3 respectively. The data shows that Aesop exhibits spreads of between 1 and 3.5 minutes in experimental runs which, in the opinion of the author, is a remarkable level of consistency given the timing variation inherent in the optimization algorithm and number of variables considered by it during execution. The spread on the initial throttling application operation is tighter at approximately 1 minute in both scenarios because the optimization algorithm performs less pre checks and back offs in this initial case.

Table 6-11: Throttling Spreads for Scenario 2

Type	Priority	Runs	Throttling Spread (Min)	
			Session Start	Session Stop
File Transfer	10	5	1.008	1.067
Web Social	3	5	1.675	0.067
Video on Demand	2	5	2.667	3.442

Table 6-12: Throttling Spreads for Scenario 3

Type	Priority	Runs	Throttling Spread (Min)	
			Impairment Increase	Impairment Decrease
Video on Demand	5	6	1.217	3.050
Web Social	3	6	2.175	2.508
Video on Demand	2	6	2.175	2.333

The compliance plots for sessions from each scenario in Figure 6-26 and Figure 6-27 of Section 6.5.4.1 show box plots in a pink colour for all sessions that were

throttled during experimental runs. Box plots were used because they visually represent five-number summaries and are very useful for comparing spreads [Moore et al., 2009]. Those box plots are plots of the information from Table N-35 to Table N-38, and show the spread of throttling application and release in the context of session compliance. The box plots highlight the consistent manner in which throttling is applied and released. In addition, they show how the compliance level of sessions is influenced by optimization throttling actions. In all the plots, the box plot for application of throttling correlates with a decrease in the average compliance of sessions, and the box plot for release of throttling correlates with an increase in the average compliance of sessions.

To summarise, throttling operations for optimization execution are applied and released in a uniform manner in all experimental runs, as indicated by the throttling spreads. Application of the first throttling action in a session has a tighter spread, and throttling actions are skewed to the right of the action that triggered their application or release. The application and release of throttling actions shows the expected tight correlation with reductions and increases in the average compliance of the session to which throttling was applied.

6.5.4.3 Accuracy of Aesop Monitoring, Analysis, and Optimization Planning and Execution Operations

Aesop deduced the session priority correctly for all experimental sessions, providing identical results to those seen in Experiment 2. The results are shown in Table N-39 and Table N-40 of Section N.3.3 in Appendix N.

The *monitoring accuracy* of Aesop, as calculated using the off-line program, summarised over all experimental runs is shown in Table 6-13. The detailed results appear in Table N-41. Aesop demonstrated a monitoring accuracy of 100%, correctly reading the information from terminal reports into snapshot individuals for all snapshots. These results, identical to the results obtained in Experiment 2, reinforce the evidence that Aesop can build and hold a Knowledge Base that accurately represents the service experience and context of running service sessions.

Table 6-13: Aesop Monitoring Accuracy

Terminal Reports	Missed	Monitoring Incorrect	Monitoring Accuracy
11571	72	0	100.00%

Aesop did not read 72 terminal reports into snapshots during loading. An analysis of the Aesop logs for the runs where terminal reports were missed was carried out. As in Experiment 2, all missed terminal reports were caused by an `ArrayIndexOutOfBounds` exception thrown by Jena while adding terminal report knowledge to a snapshot model. The optimization planning and execution operations running in Experiment 3 used the Knowledge Base heavily, causing a higher frequency of such exceptions.

The *analysis accuracy* of Aesop, again calculated using the off-line program, summarised over all experimental runs is shown in Table 6-14. The detailed results appear in Table N-42. The analysis accuracy of Aesop of 100% was identical to the analysis accuracy observed in Experiment 2, reinforcing the evidence that Aesop can accurately determine the compliance of running sessions with expectations.

Table 6-14: Aesop Analysis Accuracy

Snapshots	Not Analysed	Analysis Incorrect	Analysis Accuracy
11499	77	0	100.00%

The Aesop analysis operation did not analyse snapshots in 77 cases, again due to concurrency exceptions, the same reason that caused 27 snapshots not to be analysed in Experiment 2, see Section 6.4.4.1. More concurrency exceptions occurred in Experiment 3 because the optimization planning and execution operations that use the Knowledge Base more heavily were not running in Experiment 2. As in Experiment 2, the zero compliance values for those snapshots were not removed or corrected prior to analysis of session compliance because that analysis should use the compliance as observed and set by Aesop. Again, the absolute number of incorrect compliance values is too small to significantly affect statistical analysis of compliance levels. Resolving this concurrency issue would have significantly altered the behaviour of Aesop, making comparison of the results of Experiment 2 and Experiment 3 impossible.

The *optimization accuracy* of Aesop measured the percentage of optimization actions that were selected correctly and applied the correct throttling parameters as specified in Table 6-5. The optimization accuracy, as checked using the off-line program, summarised over all experimental runs is shown in Table 6-15. The detailed results for all runs are shown in Table N-43.

The optimization accuracy of Aesop was 100%, with Aesop applying and releasing throttling in the correct order for all normal experimental runs; Aesop always applied throttling to the unthrottled session with the lowest priority and always removed throttling from the throttled session with the highest priority. In addition, Aesop handled throttling actions correctly on terminated and restarted sessions, Aesop automatically removed throttling on 11 sessions that terminated, and re-applied throttling on one restarted session. Aesop applied the correct throttling parameters in all cases. These results provide evidence that Aesop can accurately apply and release throttling actions for session optimization.

Table 6-15: Aesop Optimization Accuracy

Throttling Actions	Correct Actions	Accuracy	Terminated Sessions	Restarted Sessions
124	124	100.00%	11	1

In summary, Aesop demonstrated perfect accuracy on monitoring, analysis and optimization planning and execution autonomic functions, indicating that the semantic lifting, queries, rules and reasoning used to implement those functions perform correctly.

6.5.4.4 Performance of Aesop Monitoring, Analysis, and Optimization Planning and Execution Operations

The performance of Aesop's monitoring, analysis, and optimization planning and execution operations were analysed statistically, the results of that analysis appear in Section N.3.5; see Table N-44, Table N-45, and Table N-46 for Scenario 2, Table N-49, Table N-50 and Table N-51 for Scenario 3. The results for both scenarios show that the observed performance for monitoring, analysis, and optimization planning and execution was within the time window of 10 seconds each as specified in the experimental hypothesis.

As optimization was active in Experiment 3, the performance of optimization planning and execution operations is examined first. Figure 6-30 shows plots of optimization time averages and standard deviations for each minute of the six runs of Scenarios 2 and 3. The results show that in Experiment 3, average optimization times remained within the time window of 10 seconds.

Optimization times increase from 0.5 seconds at the point at which sessions start (minute 9 to 12 for Scenario 2 and minute 6 to 9 for Scenario 3) to 3 seconds at a point 10 minutes later (minute 22 for Scenario 2 and minute 19 for Scenario 3).

Optimization times increase more slowly to 3.5 seconds at the point at which sessions begin to terminate (minute 45 for Scenario 2 and minute 48 for Scenario 3), at which point optimization times tail off. Although the average optimization time does not differ greatly between Scenarios 2 and 3, the spread for both scenarios differs significantly. The standard deviation on optimization times is stable at levels below 250 milliseconds for the initial and final period of Scenario 2 runs and for all of Scenario 3 runs excepting a peak at 33 minutes. However, optimization time standard deviation fluctuates greatly on Scenario 2 between minute 18 and minute 50.

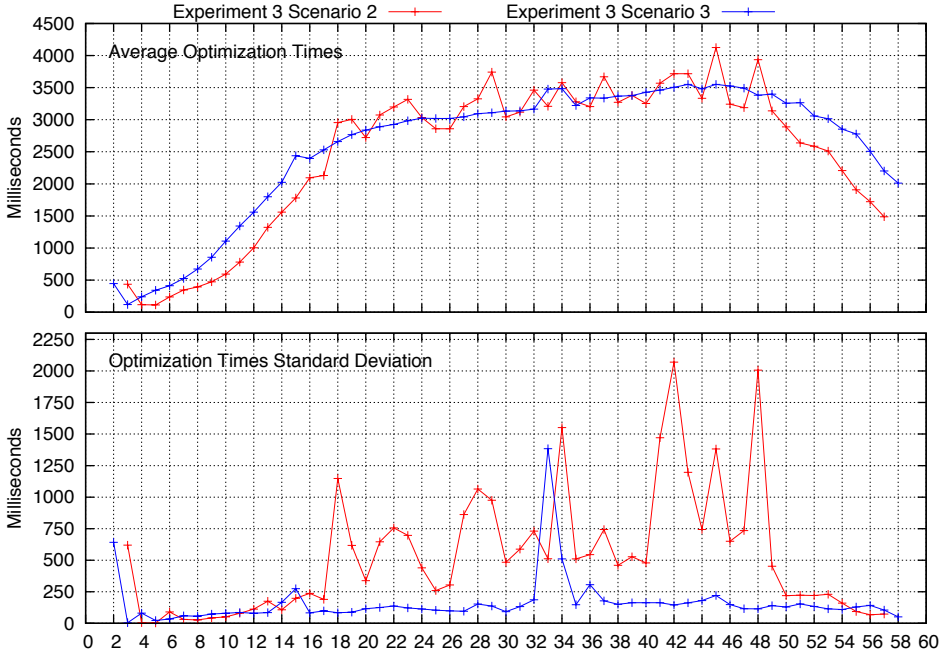


Figure 6-30: Optimization Times Average and Standard Deviation

The reason for the plateauing trend observed on optimization times 10 minutes after all sessions have started is because optimization planning and execution operations use snapshot models heavily and Aesop housekeeping clears snapshots, snapshot buckets, and snapshot models that are older than 10 minutes. After 10 minutes of operation with all sessions running, the number and size of snapshot models in the Knowledge Base has stabilized. The lower increasing trend observed from that point is due to the growth of the Dynamic Global Knowledge model of Aesop, see the discussion in Section 6.4.4.2.

The difference in the spread of optimization times between Scenarios 2 and 3 is due the behaviour of the semantic Optimization Planning and Execution algorithm (see Section 4.3.4.3) when applied to these different scenarios. The optimization algorithm uses a series of pre-checks before carrying out a full calculation on whether

to carry out an optimization action. Optimization planning and execution operations that continue after pre-checks use the Knowledge Base more lightly than optimization planning and execution operations that execute fully and are less likely to be blocked while waiting for monitoring and analysis operations to complete.

Scenario 2 is a constant congestion scenario; all sessions are non-compliant for most of the duration of experimental runs, which means that, once session establishment has completed, the optimization algorithm always passes pre checks and executes fully to determine if throttling should be applied to or released from a session. In contrast, Scenario 3 is a scenario with a progressive impairment. Sessions are fully compliant for large parts of experimental runs, which means the optimization algorithm completes after its initial compliance pre-checks.

The back off behaviour of the optimization algorithm also tends to increase the number of optimization runs that complete during pre checks in Scenario 3 as opposed to Scenario 2. The algorithm is designed to back off for a at least 40 seconds before applying a throttling action and 70 seconds before releasing a throttling action if another throttling action has been applied or released in that period. During this back off period, optimization planning and execution operations complete without carrying out any further checks. A glance at Figure 6-28 in Section 6.5.4.2 shows that all throttling actions are applied to Scenario 2 by minute 16. By minute 17 the back off period for the last throttling action has expired, and all subsequent optimization planning and execution operations must execute fully. In Scenario 3, as shown in Figure 6-29 in Section 6.5.4.2, throttling is progressively applied and released as the network impairment progresses and optimization planning and execution operations complete after pre-checks for much of the impairment period. The peak at minute 33 in Scenario 3 occurs because the network impairment improvement that begins at minute 30 results in full optimization actions in that minute to release throttling on the priority 2 Video on Demand session.

Figure 6-31 shows plots of monitoring time averages and standard deviations for each minute of the six runs of Scenarios 2 and 3 for both Experiments 2 and 3, the same plot as Figure 6-19 for Experiment 2. The plots show that average monitoring time remains within the allocated time window of 10 seconds for all experiments.

Average monitoring times and monitoring time standard deviation for Scenario 2 of Experiment 3 correlate closely with those of Experiment 2 up to minute 18 and

after minute 50. Between minutes 18 and 50, average monitoring time fluctuates irregularly between 1 second and 4 seconds. Standard deviation also fluctuates irregularly between 1 second and 5 seconds. The peaks and troughs in average and standard deviation of times mirror each other perfectly, when an increase in average time is observed, an increase in the spread is also observed. This time window from minute 18 to minute 50 is the same time window for which fluctuations in the standard deviation of optimization time were observed in Figure 6-30 and the amplitude of the fluctuations in average monitoring time for Scenario 2 match the fluctuations in optimization time for runs of the same scenario.

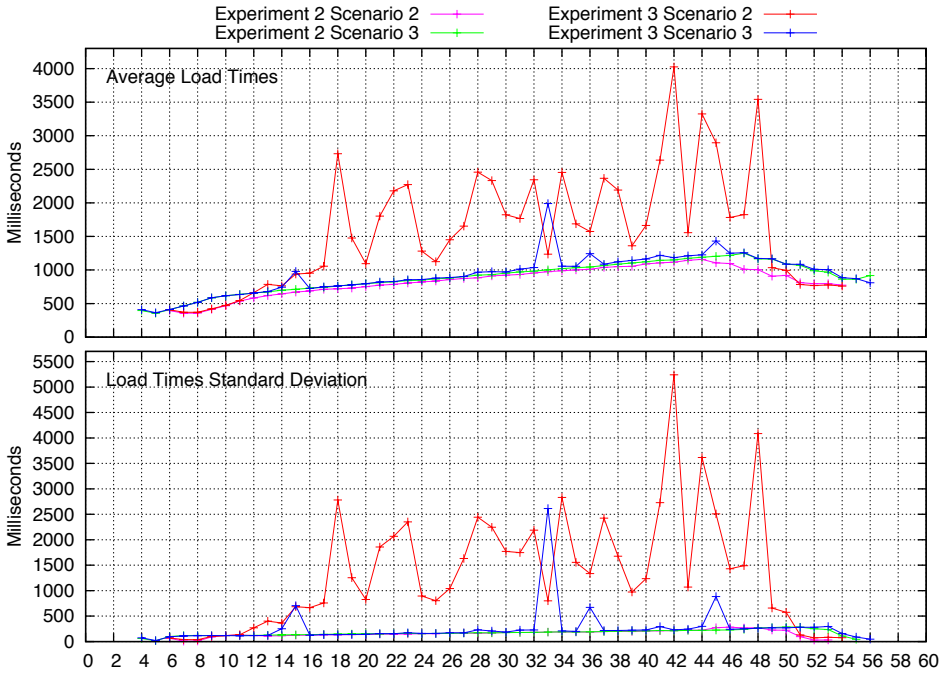


Figure 6-31: Monitoring Times Average and Standard Deviation

Average monitoring times for Scenario 3 of Experiment 3 do correlate closely with those of Experiment 2 with the exception of four peaks, showing the same linear increase from 0.6 seconds when all sessions are running to 1.2 seconds at 48 minutes when sessions begin to terminate. A large peak to 2 seconds occurs at minute 33 and three smaller peaks to 1 second, 1.2 seconds, and 1.5 seconds occur at minute 15, 36, and 45 respectively. The standard deviation curve for Scenario 3 of Experiment 3 also correlated with the Experiment 2 results very closely; except for the same peaks that were observed in the average monitoring time plot. In the case of standard deviation, the peaks are larger than on the average plot.

Figure 6-32 shows plots of analysis time averages and standard deviations for each minute of the six runs of Scenarios 2 and 3 for both Experiments 2 and 3, the same

plot as Figure 6-20 for Experiment 2. The plots show that average analysis time remains within the allocated time window of 10 seconds for all experiments.

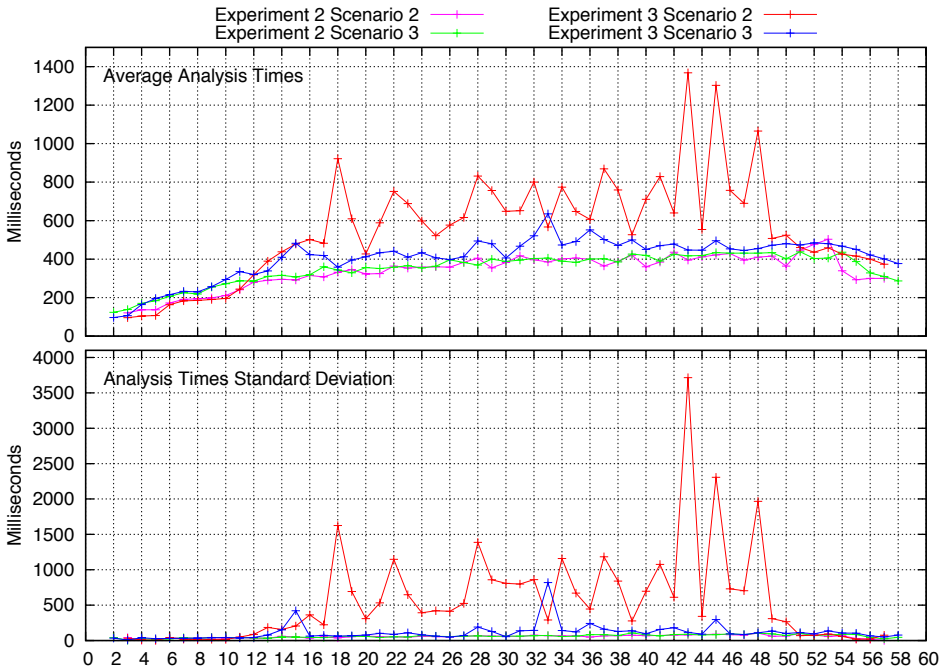


Figure 6-32: Analysis Times Average and Standard Deviation

As with monitoring times, average analysis times and analysis time standard deviation for Scenario 2 of Experiment 3 correlate closely with those of Experiment 2 up to minute 18 and after minute 50. Between minutes 18 and 50, average monitoring time fluctuates irregularly between 0.5 seconds and 1.4 seconds, a much smaller level of fluctuation than for average monitoring times. Standard deviation also fluctuates irregularly between 0.3 seconds and 4 seconds. As with monitoring times, the peaks and troughs in average and standard deviation of times mirror each other perfectly, when an increase in average time is observed, an increase in the spread is also observed. This time window is again from minute 18 to minute 50, and the amplitude of the fluctuations in average analysis time for Scenario 2 match the fluctuations in optimization time for runs of the same scenario.

As seen with monitoring times, the Scenario 3 average and standard deviation of analysis times do correlate closely with those of Experiment 2 with the exception of the same four peaks, showing the same linear increase from 0.2 seconds when all sessions are running to 0.5 seconds at 28 minutes, plateauing at this level until sessions begin to terminate. The standard deviation curve for Scenario 3 of Experiment 3 also correlated with the Experiment 2 results very closely, except for

the same peaks that were observed in the average analysis time plot. Again, the peaks on the standard deviation plot are larger than on the average plot.

To examine the relationship between optimization times and monitoring and analysis times, average monitoring, analysis, and optimization times were plotted together for Scenarios 2 and 3 in Figure 6-33. These plots illustrate that peaks and troughs in monitoring time and analysis time correlate closely with those in optimization time. It can be observed that a small increase in optimization time causes a large increase in monitoring time and a smaller increase in analysis time.

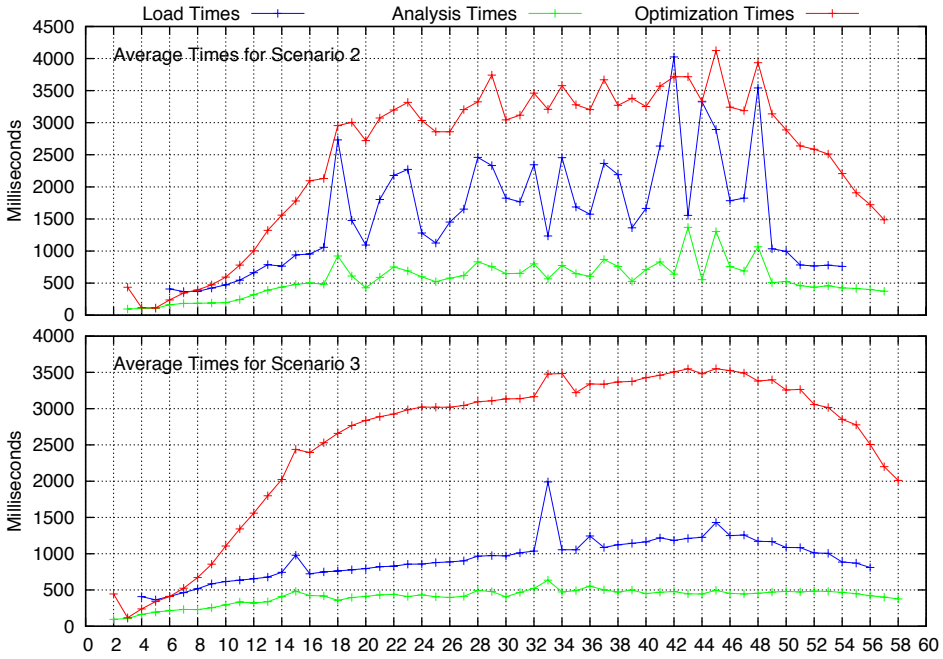


Figure 6-33: Average Times for Experiment 3 Runs

The effect is easier to observe on the plot for Scenario 3. The four peaks noticed in monitoring and analysis times in Scenario 3 at minutes 15, 33, 36, and 45 correspond to small peaks in an otherwise smooth optimization time curve. In the peak at 33 minutes, an increase of 0.3 seconds in optimization time causes an increase of 2 seconds in monitoring time, doubling average monitoring time for that minute. The impact on analysis times is less severe; the same peak in optimization time causes an increase of 0.1 seconds in analysis time.

The reason for this effect is that monitoring, analysis, and optimization planning and execution operations all read and write to the Knowledge Base. As described in Section 4.4.3, access to the Knowledge Base over the Knowledge Bus is synchronized so when optimization planning and execution operations, which use the Knowledge Base heavily, often execute fully as they do in Scenario 2, they tend to lock access to

the Knowledge Base. This causes monitoring and analysis operations to wait, thus increasing monitoring and analysis times.

CPU and memory usage was also analysed statistically, the results of that analysis appear in Section N.3.5; see Table N-47 and Table N-48 for Scenario 2, Table N-52 and Table N-53 for Scenario 3. The CPU and memory usage results are plotted for both Experiment 2 and Experiment 3 in Figure 6-34 and Figure 6-35 respectively. The corresponding plots for Experiment 2 are in Figure 6-21 and Figure 6-22. Aesop did not use system resources excessively. Average CPU usage remained at or below 20% and memory usage remained below 6GB on the Home Gateway running Aesop.

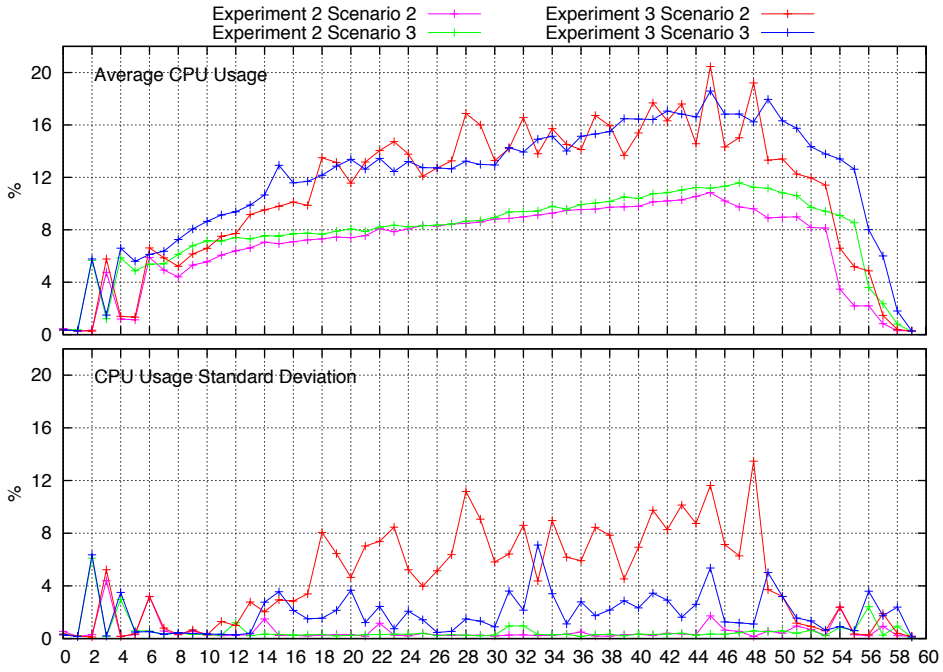


Figure 6-34: CPU Usage Average and Standard Deviation

The average CPU usage plots for both scenarios in Experiment 3 shown in Figure 6-34 follow a similar curve, with the plot for Scenario 2 again showing more fluctuations than for Scenario 3 in the minute 18 to 50 window. After some initial spikes when Aesop is started, CPU usage increases from 8% to 17% over the course of experimental runs, and is approximately 5% higher for Experiment 3 scenarios than for Experiment 2, due to optimization runs executing. Standard deviation in CPU usage is much higher in both scenarios on Experiment 3 than in Experiment 2. In Scenario 2, it fluctuates between 4% and 12% whereas in Scenario 3 it fluctuates between 1% and 7%. The peaks and troughs in the average and standard deviation of CPU usage plots show a high degree of correlation, and close correlation with peaks

and troughs in standard deviation of optimization time observed in Figure 6-30, when the CPU-intensive full optimization planning and execution operations run.

Memory usage, shown in Figure 6-35, increases from 3GB at the beginning of experimental runs to over 5GB at the point at which all sessions are running. From that point it increases very slowly towards 6GB at the end of runs. As explained in Section 6.4.4.2, the memory limit on the Java virtual machine was set at 6GB; as memory usage approaches that value garbage collection operates more aggressively in the JVM [Oracle, 2012] causing memory usage to flatten out. This effect can be observed by comparing the memory usage of experiments 2 and 3. At minute 12, the difference in memory usage is over 2GB but by minute 56, the difference is about 1GB because garbage collection is less aggressive in Experiment 2 runs.

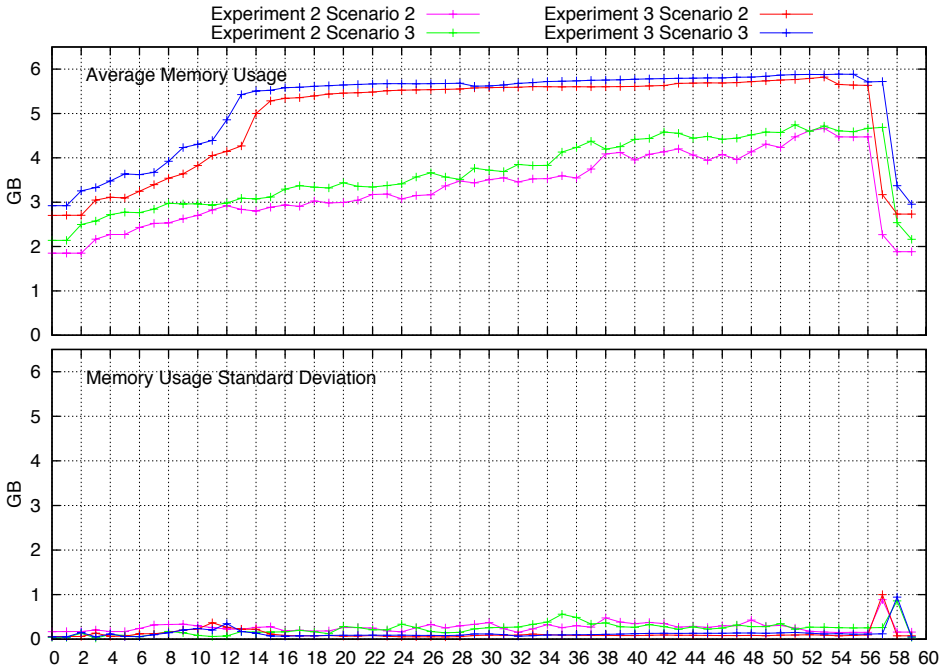


Figure 6-35: Memory Usage Average and Standard Deviation

To summarise, the observed performance for monitoring, analysis and optimization planning and execution autonomic functions in Aesop was fast enough to be considered near real time, within the allocated time window of 10 seconds each. The results indicate that increases in the time taken by optimization planning and execution operations causes corresponding increases in the time taken by monitoring and analysis operations. CPU and memory usage was well behaved.

6.5.5 Summary

Experiment 3 had the purpose of evaluating Aesop as a full autonomic system with all four autonomic functions running. The experiment was conducted to assess the

extent to which semantic monitoring, analysis, and optimization planning and execution could be undertaken in near real time.

The experiment was executed using identical experimental scenarios and setup to Experiment 2, the only difference being that Aesop's optimization planning and execution operations were active in experimental runs for Experiment 3. This allowed the effectiveness of optimization to be determined by comparing the results of Experiment 3 with those of Experiment 2. The consistency, accuracy, and performance of Aesop's optimization planning and execution operations were also measured and analysed.

The results provided evidence that strongly supported the experimental hypothesis. The evidence shows that Aesop effectively, consistently, and accurately prioritized the most important service sessions in a home area network at the expense of less important service sessions. The experiment also provided evidence that Aesop performed monitoring, analysis, and optimization planning and execution inside 10-second windows, showing that terminal report measurements were considered in optimization calculations less than 30 seconds after being received.

The ***effectiveness*** of Aesop in optimizing end user sessions was assessed by comparing the compliance of non-throttled sessions in Experiment 3 with the compliance of the equivalent sessions in Experiment 2 using the calculation in Listing 6-1. The results indicate that Aesop improved the compliance of high priority sessions, particularly when the sessions demonstrated low compliance levels when not optimized. Aesop improved the compliance of Scenario 2 sessions by 136% when all sessions were considered and by 176% when only sessions with low compliance levels prior to optimization were considered. An improvement in compliance of Scenario 3 sessions by 13% was observed when all sessions were considered, and by 60% when only sessions with low compliance levels prior to optimization were considered. The main reason for the difference in effectiveness between Scenario 2 and Scenario 3 was that in Scenario 2, where the issue of degrading compliance was local to the home area network, Aesop resolved the issue. In Scenario 3 where the issue was external to the home area network, Aesop reduced the impact of that issue on the highest priority sessions.

The ***consistency*** of Aesop in optimizing sessions across experimental runs was analysed by checking if Aesop optimized sessions of each run in the same way and by

measuring the time range for each throttling action across runs. In all runs, Aesop displayed a high level of consistency, applying and releasing the appropriate throttling action for optimization in the correct order and within a time spread of less than 3.5 minutes for each action. Aesop handled the time variance inherent in the knowledge for the sessions being considered for optimization and the number of variables taken into consideration in the optimization algorithm well.

As in Experiment 2, the ***accuracy*** of Aesop was assessed by checking how well it deduced the priority of each session, read information into snapshots in the Knowledge Base, analysed those snapshots to determine compliance, and optimized service delivery based on the information in snapshots. The results show that, as in Experiment 2, Aesop deduced the session priority correctly for all sessions, read 97% of all received terminal reports, and lifted the information on all terminal reports into the Knowledge Base correctly. Aesop calculated compliance on 99.3% of snapshots and calculated compliance correctly in all cases. Aesop applied and released throttling with 100% accuracy, always applying throttling on the lowest priority unthrottled session and releasing throttling on the highest priority throttled session.

The ***performance*** results for Aesop from Experiment 3 indicated that optimization planning and execution operations performed very well. The average optimization times plateaued at a level of approximately 3.5 seconds, well within the 10-second time allocation for optimization in the experimental hypothesis. The spread on optimization times demonstrated a standard deviation below 0.5 seconds for most of the duration of experimental runs, exhibiting occasional peaks to 1 or 2 seconds, particularly on Scenario 2 runs where optimization planning and execution operations run past pre-checks to completion more often than in Scenario 3 runs.

Average monitoring times for Experiment 3 remain below 2 seconds with a standard deviation below 2 seconds for most minutes of both runs, exhibiting occasional peaks of up to 4 seconds for average monitoring time and up 5.2 seconds for monitoring time standard deviation. These values are under the 10-second time budgets allocated for report monitoring. Average and standard deviation on monitoring time is heavily influenced by the spread in optimization time, when optimization planning and execution operations are using the Knowledge Base intensively, monitoring operations are blocked and monitoring times increase. Scenario 2 monitoring times exhibited large fluctuations in monitoring times in

comparison with Scenario 3 monitoring times because optimization planning and execution operations are more intense in Scenario 2.

The average analysis times for Experiment 3 are at levels below 0.7 seconds with a standard deviation of below 1 second for most minutes of runs. Average and standard deviation in analysis time show occasional peaks of up to 1.4 seconds and 3.7 seconds respectively in Scenario 2 runs. These values are well within the 10-second time budget allocated to analysis operations. Average analysis times are also affected by Knowledge Base blocking caused by intense optimization planning and execution operations, but not to the same extent as monitoring times.

Average CPU usage rose to approximately 17% during Experiment 3 runs with peaks to 20%, up from levels of 12% for Experiment 2 runs. This increase is attributable to optimization planning and execution operations. The spread in CPU usage was larger in Experiment 3 runs, exhibiting peaks of 12% in Scenario 2 runs and 7% in Scenario 3 runs. These peaks again correlated closely with peaks in the spread of optimization times, indicating that intensive optimization planning and execution operations were indeed running at these times. Average memory usage plateaued at levels near 6GB for both scenarios in Experiment 3, exhibiting very low levels of standard deviation. The memory limit of the Java virtual machine running Aesop was set at 6GB.

6.6 Experience with Semantic Technologies Used

In accordance with the proposed approach, Aesop was developed and implemented using readily available semantic frameworks, libraries, and toolsets. In this section, experience using these technologies is briefly discussed.

6.6.1 Protégé

The Protégé ontology editor is a convenient tool for designing ontologies, even if the user interface is a little crude. One particularly useful feature was the ability to use the Pellet Reasoner at design time to validate the EUSAO ontology during implementation and to validate the global static individual data that was input using Protégé. However, the user interface of Protégé is very cumbersome when inputting large amounts of individual data. The approach taken for specification of some individual knowledge was to specify the individuals directly in RDF/XML. That specification was then loaded into Protégé for validation with the Pellet reasoner.

6.6.2 Jena

The Jena model API was used extensively in the prototype implementation of Aesop and Jena's ARQ SPARQL processor was used to execute queries. The model and querying APIs were intuitive and well documented. The model API was rich enough to access and store knowledge in models at various levels of detail.

The performance of storage and access operations to and from Jena unreasoned models (which was of the order of small numbers of milliseconds) was not the main bottleneck in the system, and was adequate for the prototype implementation.

Jena exhibited some instability, throwing `ArrayIndexOutOfBounds` exceptions during semantic monitoring operations when knowledge was being written into models, as documented in Sections 6.4.4.1 and 6.5.4.3. While the number of operations with exceptions was small in comparison with the overall number of monitoring operations, such a bug would require a solution in a deployed system.

6.6.3 The Pellet Reasoner

The Pellet reasoner was used as the reasoner in the prototype implementation of Aesop. The reasoner was very easy to deploy and use because it is integrated into both Protégé and Jena.

As expected arising out of the literature review (see Section 2.3.3), the performance of semantic operations on models being reasoned over by the Pellet reasoner were slower than those executed towards unreasoned models, of the order of hundreds of milliseconds. In an informal test where the same query was executed towards a reasoned model using the Jena model API and using a SPARQL query, the average and standard deviation on the time taken was roughly equivalent in both methods. The performance of the reasoner was adequate for the prototype implementation because the Aesop design approach ensured the size of reasoned models was small and because reasoning load was reduced due to the decision not to load the metadata for referenced models at run time, as explained in Section 4.4.2.

Most of the time budget for a semantic operation is attributable to operations towards reasoned models. An examination of the semantic algorithm for optimization planning and execution shown in Figure 4-6 shows that it executes up to twelve queries on reasoned models. If each query takes 250ms, three seconds of each operation is spent on reasoning, which is 75% of the average optimization time, as shown in Figure 6-30. Therefore, tuning of the Pellet reasoner or selection of a more

suitable reasoner such as that discussed by Tai et al. (2011a) to improve query time would significantly improve the performance of the Aesop prototype.

6.6.4 SPARQL

All queries in the prototype implementation of Aesop were implemented in SPARQL. The language was found to be flexible, expressive and powerful. Rather complex queries were relatively easy to specify in the language.

One feature that was missing from the language implementation is the ability to express parameters to queries. In the queries used for the semantic analysis algorithm, Aesop must pass the URIs of *SSM_Session*, *SSM_ServiceExpectation*, and *SSM_ServiceExperienceSnapshot* individuals into service-specific SPARQL queries. This was achieved in the prototype implementation by replacing placeholder tags such as the *<---VAR_SESSION--->* tag in Listing C-4 of Appendix C with the appropriate URI in the text of the query using a Java string substitution prior to sending the query to the Jena SPARQL processor.

The performance of queries was dependent on whether they were run on unreasoned or reasoned models, as explained in Sections 6.6.2 and 6.6.3.

6.6.5 SWRL Rules

The implementation of rules in the Aesop prototype was rather straightforward because the Pellet reasoner provides automatic support for SWRL rules. Any SWRL rules in a reasoned model are fired once a pertinent individual is loaded. This approach for rule implementation is certainly convenient and useful.

In Knowledge Monitoring, SWRL rules were used to infer knowledge on loading. The two rules used, shown in Listing C-1 and Listing C-2 of Appendix C, did not cause a measurable increase in processing time when introduced.

6.7 Conclusions from Experimentation

Analysis of the experimental results provides evidence to draw the following conclusions on the use of semantic techniques to autonomically optimize end user service delivery in the Home Area networking domain.

1. The effectiveness, consistency, accuracy, and performance evaluation property results show that Aesop, a semantic end user service management system, manages end user services very well in the Home Area networking domain in near real time.

2. When the root cause of degradation in compliance is an issue internal to the HAN, the strategy of optimization by restricting the bandwidth available to low priority services worked very well. When the root cause was outside the management domain in the access network, this strategy was less effective. In cases where the root cause is outside the domain, an effective approach would be to implement co-operation or at least alerting between Aesop and an Aesop-like system managing end-user service quality in the access network.
3. The optimization planning and execution operations use the Knowledge Base extensively, and tend to block the Knowledge Base. Care must be taken in designing optimization queries to be as efficient as possible to avoid locking the Knowledge Base for significant periods. Alternatively, as optimization planning and execution operations read from the Knowledge Base extensively but write little information, adding support to the Knowledge Base to allow non-blocking access for read-only operations is likely to significantly alleviate this issue.
4. Throttling settings on Social Network Web Browsing sessions in Scenario 2 and Video on Demand priority 5 sessions in Scenario 3 were not sufficiently high to significantly reduce bandwidth consumption of these sessions. Tuning of session throttling must be executed carefully if it is required to still allow a restricted level of service on the throttled sessions. If it is allowable to block throttled sessions completely, then tuning of throttling is much more straightforward.
5. The prototype Aesop system used Jena [Apache, 2013c] to hold semantic models and to run queries, and the Pellet reasoner [Clark & Parsia, 2013] to reason over models. Jena and the Pellet Reasoner are general-purpose semantic software components. Because the End User service management domain as specified in the EUSAO ontology is quite well structured, using more specialized modelling and reasoning components and tuning those components correctly is likely to yield an improvement in performance results.

Experimentation examined the extent to which Aesop could be applied in the Home Area Networking domain. Chapter 7 explores the applicability of Aesop to the Mobile Broadband Access domain.

Chapter 7
Mobile Broadband Case Study

7.1 Introduction

The motivation for the case study was to explore the applicability of the Aesop approach beyond the Home Area Network domain, in order to understand the potential strengths and weaknesses of the approach when applied in other domains. Thus the goal of this particular case study is to explore the applicability of the Aesop approach in the domain of Mobile Broadband Access networks, and to what extent autonomic optimization of end user service delivery can be achieved in near real time in that domain.

The Mobile Broadband Access Network domain was chosen for this study because network operators have identified optimization as one of their top requirements for increasing operational efficiency [NGMN, 2010]. As discussed in Section A.5, delivery of end user services that meet the expectations of end users is more challenging in mobile broadband access networks than in fixed networks [OFCOM, 2011]. Oyman & Singh (2012) pointed out that research on QoE-aware adaptation at radio and network level in mobile networks is required for end user service session optimization. In addition, the author of this thesis is very familiar with management of this networking domain, having worked with management systems for Mobile Broadband access since the inception of UMTS (Universal Mobile Telecommunications System) [3GPP, 2013a] in the late 1990s. The assessment of the application of the Aesop approach to this domain is realistic based on the author's experience in the domain.

Section 7.2 provides background on end user service delivery in Mobile Broadband Access Networks. Section 7.3 specifies a scenario outlining how mobile network operators might implement end user service optimization in their networks. In Section 7.4, the suggested placement of Aesop components in the 3GPP Mobile Network architecture is described. Section 7.5 assesses the applicability of the Aesop

architecture and structure to optimization of end user service delivery in Mobile Broadband Access Networks. Section 7.6 summarises the case study.

7.2 Background

Consider an end user service session on a given UE (User Equipment) running in the context of a cell in a Mobile Broadband Access Network. The context of that service session has three important aspects. Each cell has its own unique *radio context*: its physical topology, the built environment in the cell, the radio technology being used, the type of cell in question, and the current meteorological conditions. The *UE context* includes such factors as the distance of each UE from the base station and the type of devices in question. The *active service context* is the number and type of end user service sessions running in the cell at a given instant.

Figure 7-1 shows the base station sites of a mobile operator in Athlone, Ireland. Comreg (The Commission for Communications Regulation) publish base station data [Comreg, 2013], which has been mapped onto Google maps [Ireland Offline, 2013]. The figure illustrates the importance of context to end user service delivery in mobile networks. User A is experiencing excellent service quality on the LTE network in the town centre. However, user B's service experience is poor because, even though she is in an area of excellent coverage, she is visiting the museum in the dungeon of the 13th century castle in the town. User C is experiencing poor service quality because he is far from the nearest base station. User D is experiencing excellent service experience because, even though she is on a boat, she is very close to a base station.

A cell in a Mobile Broadband Access network may be considered to be a discrete *Cell Wireless Network* in its own right. The Cell Wireless Network is delivering a set of end user service sessions with its own unique individual context. The entire Mobile Broadband Access network is considered as an aggregated network, made up of Cell Wireless Networks. End user service sessions run in a Cell Wireless Network, with the Mobile Broadband Access network controlling details of what sessions run in which Cell Wireless Network, transfer of sessions between Cell Wireless Networks, and other functions such as addressing and routing.

Optimizing end user service delivery in a Mobile Broadband Access network is much more challenging than merely monitoring and analysing end user service delivery [Sterle et al., 2011]. With appropriate sample design, a network wide statistical approach can be applied to monitoring and analysis, which will give a good

picture of how well services are being delivered in the network as a whole [Soldani, 2006]. Indeed, with appropriate sampling, an accurate estimation of quality variations for parameters such as regions, service types, and user categories can be made [Groenendijk et al., 2011]. When using approaches that estimate end user service experience to optimise end user service sessions against service expectations [Latre et al., 2009][Sterle et al., 2011], it is difficult to take the particular network context in which each of those service sessions are being delivered into account. In order to correctly optimize end user service delivery in a Mobile Broadband Access network, experience aware optimization must be carried out at radio and network level [Oyman & Singh, 2012].



Figure 7-1: Base Stations and Possible Cells in Athlone, Ireland

In order to optimize the compliance with expectations of the set of end user service sessions running in a Cell Wireless Network, the context in which those services run must be modified. The UE context such as the position of a UE cannot be controlled by a management system. Although the radio context of a cell may be optimized to improve the bandwidth available through actions like changing antenna tilt, such actions are in the domain of radio planning. Therefore, modifying the active service context is the most practical approach for optimizing end user services in near real time.

In summary, in order to optimize service delivery in a Mobile Broadband Access network in near real time, the approach proposed is to optimize the service sessions in each Cell Wireless network individually.

7.3 Scenario

Red Comma is a network operator that provides mobile broadband service in a European country on their UMTS (Universal Mobile Telecommunications System) [3GPP, 2013a] and LTE (Long Term Evolution) [3GPP, 2012h] mobile networks. The volume of data being carried on Red Comma's Mobile Broadband Network has been increasing rapidly, leading to congestion. Red Comma runs regular radio network optimization and network rollout projects, but traffic analysis has revealed that a certain degree of network congestion is likely to be an ongoing problem in certain cells at peak times.

Red Comma is deploying the Aesop approach for optimization of end user service delivery. Customers can subscribe to four levels of Mobile Broadband service: Platinum, Gold, Silver, or the default level of Bronze. Red Comma sets the service expectations and throttling levels to be applied on the Voice over IP, Web Browsing, and Video services for customers at Platinum, Gold, Silver, and Bronze level, and installs software on the smart phones of Platinum, Gold, and Silver customers. That software sends terminal reports with Quality of Service and Quality of Experience metrics on active service sessions to an Aesop system every minute.

Aesop optimizes the set of end user service sessions in each cell individually. It uses network monitoring to determine the sessions that are running in each cell and the terminal report monitoring to determine the compliance with expectations of the sessions of Platinum, Gold, and Silver customers. When congestion occurs, Aesop progressively throttles the sessions of lower priority users, thus increasing the compliance of high priority sessions with expectations.

7.4 Suggested Placement of Aesop in the 3GPP Network Architecture

The diagram in Figure 7-2 shows the suggested placement of Aesop in the UMTS [3GPP, 2013a] and LTE [3GPP, 2012h] Mobile Broadband Access network architecture as specified by the 3GPP [3GPP, 2012a]. In both UMTS and LTE networks, the traffic of end user sessions (red/blue lines in Figure 7-2) is carried

wirelessly between User Equipment and base stations, traversing the Core Network to the Internet via a Gateway node. Allocation of radio resources for UE connections carrying end user sessions (red/black lines in Figure 7-2) are controlled by a SGSN (Serving GPRS Support Node) in UMTS and by a MME (Mobility Management Entity) in LTE [3GPP, 2012a]. The PCRF (Policy Control and Charging Rule Function) [3GPP, 2012d] makes policy decisions on the Quality of Service class and bit rates of IP flows between User Equipment and the Gateway node using rules supplied by applications [Balbas et al., 2009].

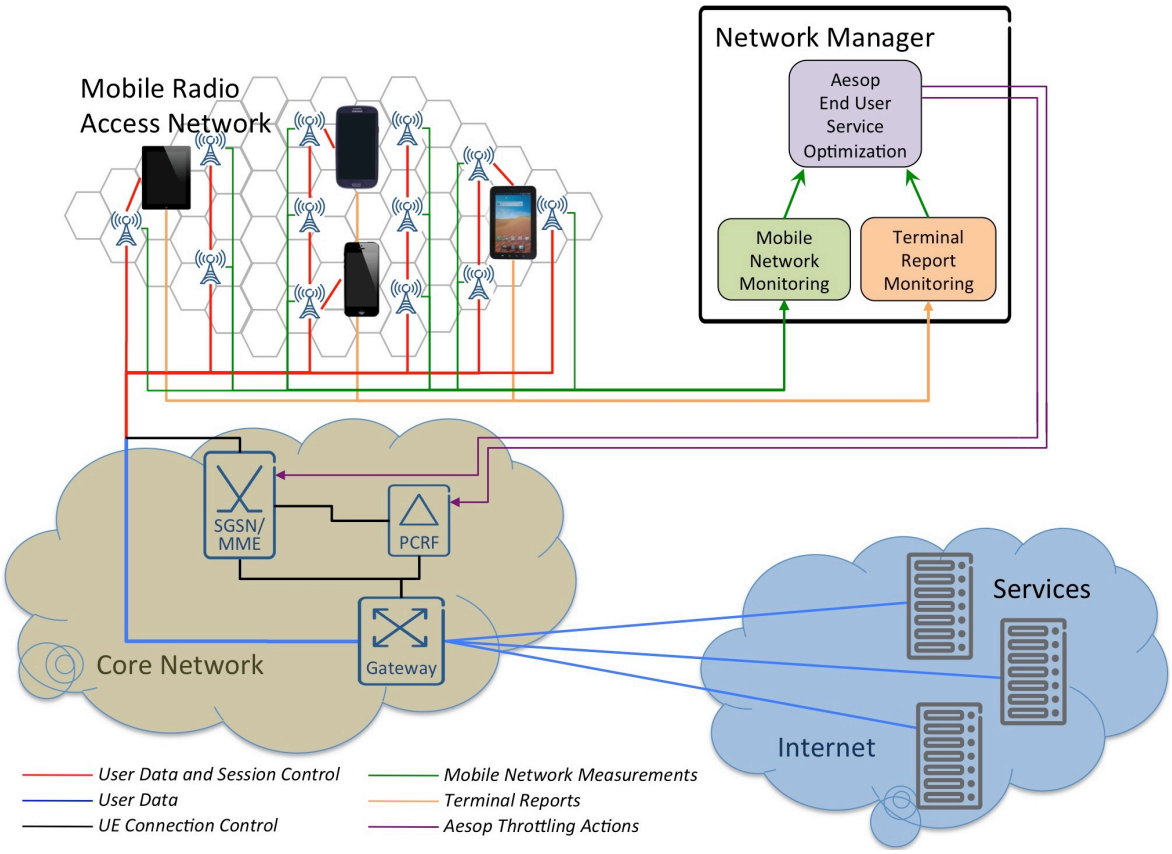


Figure 7-2: Aesop End User Service Delivery Optimization for Mobile Broadband

The Aesop deployment would be placed in the Network Manager of the mobile network. Aesop would monitor end user service experience and context using the Terminal Report Monitoring function of the network manager. That function receives measurements for end user sessions running on terminals as terminal reports (orange lines in Figure 7-2) from software running in the terminals using mechanisms such as those described in 2.3.1.1. Additional contextual information such as details of active end user sessions for which terminal reporting is not active (Bronze end user sessions) and cell measurements would be received from the Mobile Network Monitoring function, which receives measurements from the mobile

network (green lines in Figure 7-2) using mechanisms such as 3GPP performance reporting [3GPP, 2012e] or events from base stations [Achuthan & O'Meara, 2013].

Aesop would use semantic techniques to autonomically analyse and plan optimizations for end user sessions in each cell. Optimizations would be executed by throttling end user sessions. Throttling requests would be sent to the SGSN or MME, which would modify the resources allocated to those sessions in the Mobile Radio Access Network. Throttling requests would also be sent to the PCRF function [3GPP, 2012d], which would modify the Quality of Service class and bit rates of end user sessions.

The Aesop End User Service Optimization deployment in Figure 7-2 would run a number of Aesop instances in order to scale to manage a full Mobile Broadband Access network. Each Aesop instance would manage a number of Cell Wireless Networks.

The placement of Aesop in the Mobile Broadband Access Network architecture as shown in Figure 7-2 complies very well with the cross layer adaptation architecture for QoE optimization suggested by Oyman & Singh (2012). Aesop would use the QoE monitoring function of their architecture to monitor end user service experience and would use their Radio adaptation and Network adaptation engines to execute optimization actions.

7.5 Assessment of Applicability of Aesop Architecture and Structure

A single Aesop instance would be deployed to manage a number of Cell Wireless Networks. The architecture and structure of Aesop as shown in Figure 4-1 would be unchanged. The implementation would be extended as shown in Figure 7-3. The Knowledge Base would manage the Snapshot Knowledge for each Cell Wireless Network separately. Aesop would run separate analysis and optimization planning and execution operations for each Cell Wireless Network. In addition, semantic mappings, queries, and rules would be adapted to monitor, analyse, plan, and execute end user services in the Mobile Broadband Access domain. Four minutes would be allocated for execution of the autonomic optimization loop: 1 minute each for monitoring, analysis, planning and execution functions. Optimization of the autonomic loop in four minutes can be considered as being near real time [NGMN, 2007b] execution.

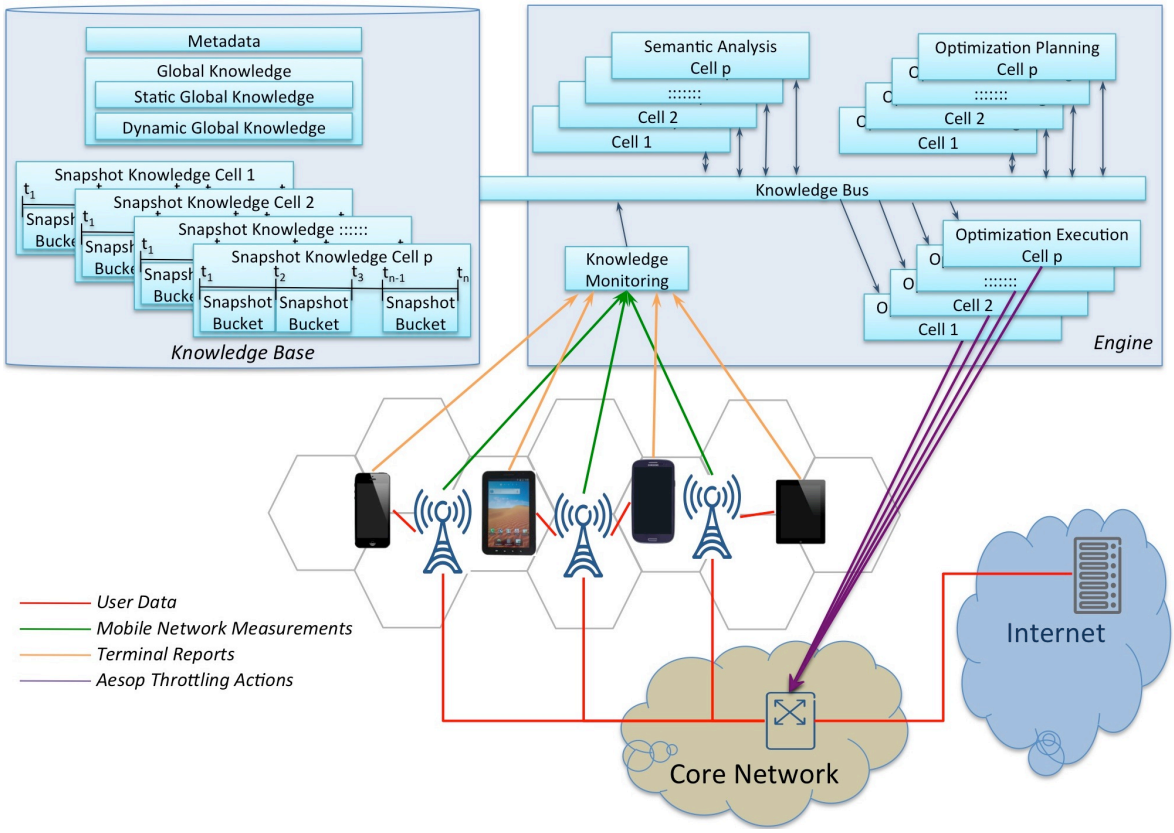


Figure 7-3: An Aesop Instance Managing Multiple Cell Wireless Networks

7.5.1 Applying the EUSAO Ontology and the Knowledge Base

The EUSAO ontology described in Chapter 3 holds the end user service management knowledge required to optimize the delivery of end user services. The concepts that would be used are as in the Home Area Networking domain. The EUSAO ontology would use the SID [TMF, 2011] concepts of *NetworkComposite* and *SubNetwork* shown in Figure 7-4 to model a Mobile Radio Access network and the Cell Wireless Networks it contains. The Mobile Radio Access network would be a *NetworkComposite*, and each Cell Wireless network would be a *SubNetwork*, as shown in Figure 3-2 in Chapter 3. Connectivity services offered by a mobile operator would use the Mobile Radio Access network modelled as a *NetworkComposite*. Each terminal in the network would use a Cell Wireless Network modelled as a *SubNetwork*. In addition, the Dynamic Global Knowledge concepts of *SSM_Terminal*, *SSM_Connection*, *SSM_Session*, and *SSM_User* and the Snapshot Knowledge concepts of *SSM_ConnectionExperience Snapshot*, *SSM_Location Snapshot*, and *SSM_Service ExperienceSnapshot* would be specialized to hold object and data properties specific to the mobile broadband access networking domain. For example, an *SSM_Terminal* specialization would include a data property to hold the International Mobile Station

Equipment Identity (IMEI), a unique identity for each terminal type in a mobile broadband access network.

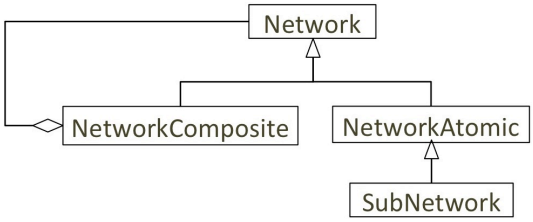


Figure 7-4: TM Forum SID *NetworkComposite* and *SubNetwork* Concepts in UML

The metadata and global knowledge would be held in common models for all Cell Wireless Networks being managed by an Aesop instance. Static Global knowledge such as service expectations would be identical across Cell Wireless Networks. Dynamic Global Knowledge such as sessions and locations of terminals would also be common, because session handoff can occur from one Cell Wireless Network to another. The Aesop instance would hold the snapshot bucket models for each Cell Wireless Network in separate lists.

7.5.2 Applying Semantic Knowledge Monitoring

The Aesop Knowledge Monitoring component can lift information in any textual format such as XML documents or Comma Separated Variable files using transformations, as explained in Section 4.4.4.1. Mappings would need to be designed and implemented which lift information from the textual format produced by Mobile Network Monitoring and Terminal Report Monitoring functions of the Network manager into Dynamic Global Knowledge and Snapshot Knowledge individuals. Specifically, individuals of the *SSM_Terminal*, *SSM_Connection*, *SSM_Session*, and *SSM_User* concepts would be lifted into Dynamic Global Knowledge. *SSM_Connection ExperienceSnapshot*, *SSM_LocationSnapshot*, and *SSM_ServiceExperienceSnapshot* would be lifted into Snapshot Knowledge. One useful source of network context would be the XML files produced by the Bulk Configuration IRP [3GPP, 2011a], which could be lifted into the knowledge base using the mechanism described in Section 4.4.4.1 by semantically tagging the XML schema for the IRP export function with SAWSDL tags and writing XSL transformations to lift the tagged information.

A new SWRL rule would infer which Cell Wireless Network each Snapshot individual belongs using the *ssm_usingSubNetwork* relation of the terminal running an end user service session. The snapshot would then be placed in the snapshot bucket model on the list for that Cell Wireless Network.

7.5.3 Applying Semantic Analysis

The implementation of the Aesop engine would be extended to run multiple Semantic Analysis operations periodically, one *SemanticAnalysis* instance (see Figure 4-7) for each Cell Wireless Network, as shown in Figure 7-3. Each Cell Wireless Network Semantic Analysis operation would operate on the Metadata, the Global Knowledge, and the particular list of Snapshot Bucket models for the Cell Wireless Network in question.

A Semantic Analysis operation of Aesop runs SPARQL queries to analyse the compliance of end user services with expectations, as explained in Section 4.3.4.2. Those queries would be updated to calculate the compliance of each end user service session in a Cell Wireless network from the Quality of Experience and Quality of Service metrics in the Service Experience Snapshot of that session during that snapshot period.

7.5.4 Applying Semantic Optimization Planning and Execution

Aesop executes Semantic Optimization Planning and Execution functions as a single operation. The implementation of the Aesop engine would be extended to run multiple Semantic Optimization planning and execution operations periodically, as with Semantic Analysis. One *OptimizationPlanning* instance (see Figure 4-7) would run for each Cell Wireless Network, as shown in Figure 7-3.

A Semantic Optimization planning and execution operation of Aesop runs SPARQL queries to plan optimizations and uses scripts to execute throttling actions that have been selected, as explained in Section 4.3.4.3. The SPARQL queries used to plan optimization throttling actions are network agnostic and are therefore unchanged. The execution scripts, which apply throttling actions selected by optimization with the throttling levels specified by the network operator, would be modified to send throttling actions to Core Network nodes.

A network operator would specify throttling of user data in the Gateway node and/or throttling of the radio resources allocated to a UE in the Cell Wireless network. Execution scripts that throttle user data would send rules to the PCRF function that modifies the QoS class and bit rates on the IP 5-tuple of the end user session in question. Execution scripts that throttle UE connections in the Mobile Radio Access Network would send commands to the SGSN or MME instructing those

nodes to restrict the radio resources allocated to the UE connection that is carrying the end user session to be throttled.

7.5.5 Deploying Aesop in a Full Network

In order to optimize end user service delivery in a complete Mobile Radio Access network, it must be possible to deploy multiple instances of the Aesop approach. Figure 7-5 shows a number of Aesop instances, where each instance would be managing end user service delivery optimization in regions of a Mobile Radio Access network. In such a deployment, more than one Aesop instance may run on a single machine, and Aesop instances may run on separate machines providing those machines are connected using high speed links with low latency, for example 10 Gbit/s optical connections.

The Knowledge Base would be distributed across Aesop instances. Metadata and global knowledge would be shared across those instances. Due to the Aesop architecture, snapshot knowledge is local to each Cell Wireless Network and would not need to be distributed.

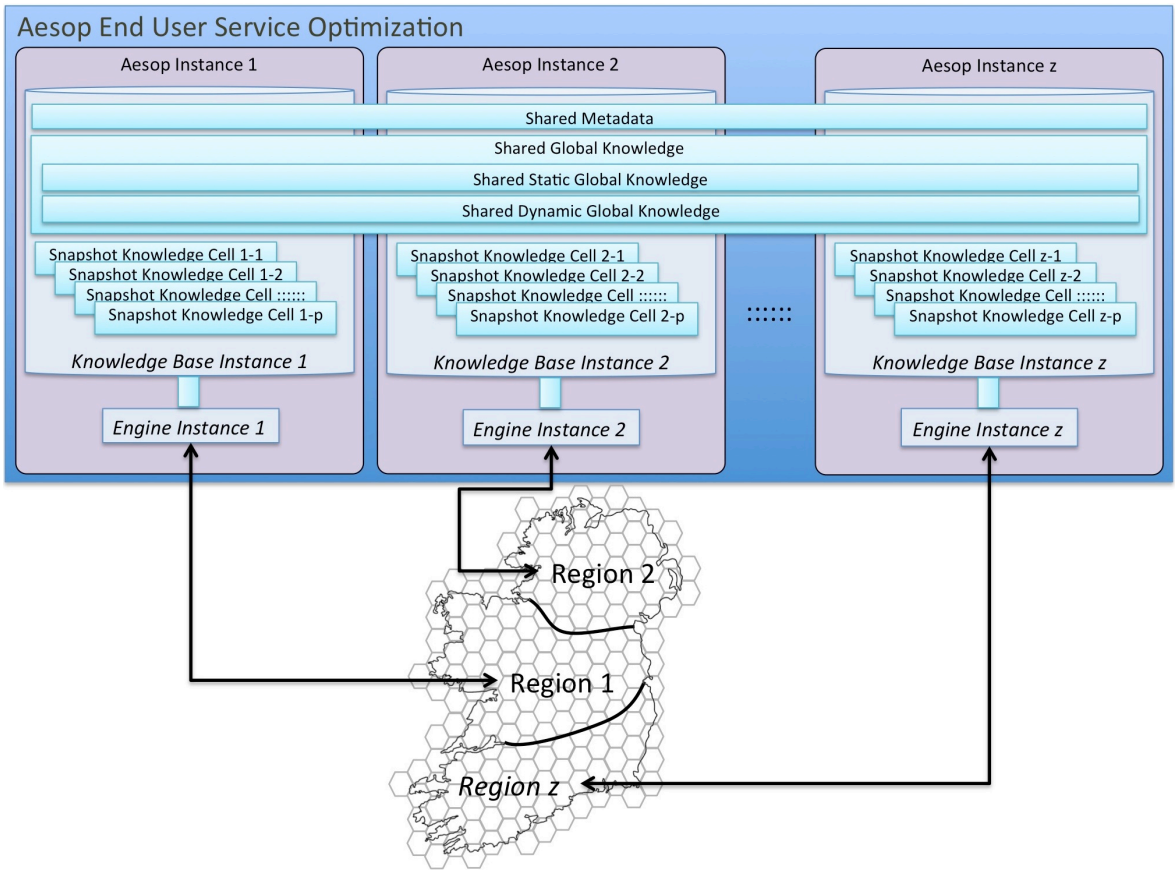


Figure 7-5: Aesop Instance Managing Regions of a Mobile Broadband Access Network

Distributing Metadata and Static Global knowledge would be straightforward. Metadata does not change, and Static Global Knowledge is changed using configuration operations that would trigger a push to all Aesop instances. Dynamic Global Knowledge would be distributed to the Dynamic Global Knowledge model in each Aesop instance using approaches such as a distributed hash table [Coulouris et al., 2012] implementation like Hazelcast [Hazelcast, 2012].

7.5.6 Optimizing End User Service Delivery in a Full Network

This section suggests a deployment of Aesop for a full Mobile Broadband Access Network and estimates the performance of Aesop in such a network.

There are 13,600 base stations in Vodafone's UK network [Vodafone, 2013]. If one assumes an average of three cells per base station [Laiho et al., 2006], there are approximately 40,800 cells in that network. A mobile operator may choose to optimize end user service delivery only in those cells where an issue with service delivery is known to exist. Alternatively, an operator may choose to optimize end user service delivery in all cells in the network.

Table 7-1 outlines how Aesop might be dimensioned and deployed to optimize end user services in a network with in the order of 50,000 cells. Although a Mobile Broadband network can support up to 200 active users [3GPP, 2009a] for bandwidth allocation, an average of 50 active end user service sessions per cell is a realistic estimation [Laiho et al., 2006] of the session load per cell.

Table 7-1: Aesop Deployment in a Mobile Broadband Access Network with 50,000 Cells

Servers	Aesop Instances Per Server	Cells Per Aesop Instance	Average Sessions per Cell	Total Cells	Total Sessions
10	25	200	50	50,000	2.5M

In the author's experience, a system such as Aesop would be expected to run on no more than 10 servers, ideally configured as blades in a rack. In order to manage 50,000 cells, each server must handle 5,000 cells, with optimization of each Cell Wireless Network being assigned to an Aesop instance running on a server.

In Experiment 3, executed on a desktop computer where semantic operations executed every 10 seconds, the maximum CPU usage observed was 25% and the maximum memory usage observed was 6GB, see Section 6.5.4.4. In the Mobile Radio Access Network deployment, semantic operations would be set to execute every minute, thus reducing the CPU load required for those operations. That factor,

combined with engineering improvements to implement the concurrent access and performance improvements suggested in Section 6.6, is likely to make running 25 Aesop instances on a multi core server with at least 150 GB of memory a realistic deployment.

The dimensioning specified in Table 7-1, specifies that each Aesop instance should handle 200 Cell Wireless Networks. Such a load would generate a maximum of 10,000 terminal reports per minute if every user were a Platinum, Gold, or Silver user. If one assumes that 50% of users are default (Bronze) users, then the reporting load on Aesop would be 5,000 terminal reports per minute or 84 terminal reports per second. Times for monitoring, analysis, and optimization planning and execution operations must be below 1 minute for Aesop to execute its autonomic loop in near real time.

In Experiment 3, where four active sessions were reporting every 10 seconds, the best average monitoring times observed were approximately 1 second, see Section 6.5.4.4. The results from Experiment 1 examined how monitoring times increased with increasing terminal reporting loads in a simulated environment. In that experiment, as shown in Figure 6-13, a small increase of 30 milliseconds was observed in the processing time for monitoring as reporting loads increase towards 100 terminal reports per second. This evidence suggests that average monitoring times for reporting loads of 84 terminal reports per second should be of the order of seconds, fast enough to be considered near real time.

Service experience snapshots from four active sessions were analysed every 10 seconds in Experiment 3, see Section 6.5.4.4. At session start when a single session is running and generating 6 service experience snapshots per minute, analysis times of 0.2 seconds were observed, as shown in Figure 6-32. When all four sessions are running and generating 24 service experience snapshots per minute, analysis times of 0.4 seconds were observed, as shown in in Figure 6-32. Albeit from very limited evidence, extrapolating from those points suggests that near real time analysis times of the order of a minute might be expected for a service experience snapshot load of 5,000 snapshots per minute.

The optimization planning and execution algorithm operates on snapshot buckets rather than on snapshots so the number of instances in each snapshot bucket should not influence optimization times significantly. In experimentation, as is noted in Section 6.5.4.4, the best average optimization times observed for a single list of

snapshot buckets was approximately 3 seconds. A naïve estimation based on this evidence would suggest that running optimization planning and execution operations for 200 Cell Wireless Networks would take 600 seconds, or 10 minutes, too slow to be considered near real time. However, executing optimization operations in parallel taken together with implementation of the engineering improvements to improve concurrent access and performance suggested in Section 6.6 are likely to substantially improve optimization times.

In summary, the performance estimations in this section show that the deployment of Aesop suggested in Table 7-1 is likely to be realizable, given some engineering improvements. Conducting further experimentation on an improved Aesop implementation will yield more accurate evidence on the performance of Aesop in such a deployment.

7.5.7 Synopsis of Enhancements and Extensions to Aesop

There would be no fundamental changes to the Aesop approach or its design to support the new domain. The structure and core concepts of the EUSAO ontology and the design of the Aesop knowledge base, knowledge bus, and engine would not change.

The following extensions to the Aesop implementation would be required to support end user service delivery in sub networks. These implementation extensions would be applicable to any other domain that uses sub networks:

1. The methods of the Knowledge Bus interface would be enhanced to support lists of snapshot buckets for multiple Cell Wireless Networks.
2. The Aesop engine implementation would be enhanced to run Semantic Analysis and semantic optimization planning and execution operations for multiple Cell Wireless Networks.

The following metadata, mapping, query, and rule extensions would be required to support the Mobile Broadband Access Networking domain:

1. The EUSAO ontology would be specialized. A Mobile Broadband Access Network would be modelled as a *NetworkComposite*, concept and each Cell Wireless network would be modelled as a *SubNetwork*. Some Dynamic Global and Snapshot concepts would be specialized to hold object and data properties that represent knowledge in the mobile broadband access domain.

2. A SWRL rule would be developed to determine the Cell Wireless Network for each service experience snapshot.
3. Mappings would be developed to lift end user service experience and context information and store it as knowledge in the Knowledge Base.
4. For Semantic Analysis, new SPARQL queries would be developed to deduce the compliance of service experience snapshots with the expectations specified by the mobile network operator.
5. Scripts would be developed to send throttling actions to the PCRF node and the SGSN or MME nodes.

Implementation of the enhancements and extensions above to the Aesop prototype implementation are likely to require 3-4 months of implementation effort.

In addition, the following engineering improvements to the Knowledge Base of the Aesop prototype are required to allow it to manage the scale of a typical Mobile Broadband access network.

1. The engineering improvements suggested in Section 6.6; a better scheme for concurrent access and a reasoner more adapted to the domain of end user service optimization.
2. Support for distribution in the Knowledge Base, to allow access to models from multiple instances of Aesop.

Implementation of these improvements is likely to require 4-6 months of effort.

7.6 Summary

This case study had the purpose of evaluating to what extent the Aesop approach could be applied to a different domain to achieve real time optimization of end user service delivery. The domain chosen was the Mobile Broadband Access Network domain. The case study was conducted to provide evidence that Objectives 2, 3, and 4 could be reached in that networking domain. The analysis arising from the case study indicates that the Aesop approach is applicable in a second networking domain, providing a measure of confidence in the potential applicability of the Aesop approach in arbitrary network types.

This case study demonstrated that the Aesop approach was well suited to application in the Mobile Broadband Access Network domain. The architecture, Knowledge Base structure and Aesop components were directly applicable in this

networking domain, and would be applicable to end user service delivery optimization in any access network delivering end user services. Enhancements to Aesop for management of Cell Wireless Networks to allow nesting of networks could be applied to other networking domains such as DSL access networks, where each Home Area Network could be considered to be a *SubNetwork*. The extensions for scalability would allow Aesop to scale to optimize end user services in very large networks.

In order to apply Aesop in an arbitrary networking domain, that networking domain must support a mechanism for retrieving end user service experience and context. In addition, it must have a mechanism that allows end user sessions to be throttled. Mappings, rules, and queries must be developed for monitoring and analysis operations in each networking domain. Scripts must be developed for each networking domain that invoke the throttling mechanism of that domain to apply optimizations specified by Aesop.

This case study suggested a centralized placement of Aesop. In the 3GPP Network Architecture all end user service experience and throttling information is collected centrally, and all control operations are co-ordinated centrally. In networks where a more distributed approach to management is taken, such as the use of ACS (Auto Configuration Servers) in management of fixed broadband access networks [Broadband Forum, 2010a], a distributed hierarchical deployment of Aesop would be appropriate. Aesop instances in Home Area networks cooperating with Aesop instances in ACS nodes and in a central management system would optimize end user server delivery for the network. Such a deployment, which would require some further extensions to the implementation of Aesop, is not examined further in this thesis and is left for future work.

Chapter 8
Conclusions and Future Work

8.1 Overview

This chapter brings this thesis to a close. In Section 8.2 the degree to which the research question has been answered and the research objectives have been achieved is discussed. The contributions of this research are presented in Section 8.3. Section 8.4 outlines future work, which would build upon the research described in this thesis. In Section 8.5, the thesis is concluded with some final remarks.

8.2 Research Question and Objectives

In Section 1.2, the research question that this thesis addresses was posed:

"To what extent can a semantic model of end user service expectations, experience, and context be used to autonomically optimize end user service session delivery in near real time?"

This research proposes the Aesop approach to address the research question. The EUSAO ontology is designed as "*a semantic model of end user service expectations, experience, and context"*. The Aesop engine is designed so it can "*be used to autonomically optimize end user service session delivery"*. The structure of the EUSAO ontology and the design of the Aesop Knowledge Base and Knowledge Bus enables the semantic-based Aesop Engine to execute *"in near real time"*. The *"To what extent"* facet of the research question, the degree to which the Aesop approach answers the research question, is discussed in the following paragraphs.

The Aesop approach is designed so that it can be applied to manage any end user service; the *extent* to which the approach can be applied to an *arbitrary end user service* is now discussed. In Experiment 3, described in Section 6.5, the approach was evaluated to establish the extent to which sessions of three end user service types could be optimized in a Home Area Network test bed. The approach demonstrated accuracy and consistency in this experiment, calculating compliance correctly and selecting the correct session for throttling actions in experimental runs. The

effectiveness of the approach was high in the experimental scenario where the cause of the issue degrading compliance was local to the home area network, more than doubling the compliance of high priority sessions with expectations. In the experimental scenario where the issue was in a network external to the home area network the approach was less effective, reducing the impact of that issue on the highest priority sessions by 13%. In such cases, as well as optimizing end user service delivery, it would be appropriate for Aesop to delegate optimization, or at least report the issue, to the management entity of that external network.

The Aesop approach assumes that quantitative expectation values for service types can be specified. Such specifications exist and are standardized for most established services (see Section 2.3.1.2.1), but may not exist for some services such as telemetry. The approach also assumes that service experience can be collected or estimated for all high priority[2] end user service sessions of all service types. Obtaining (see Section 2.3.1.1) or estimating (see Section 2.3.1.2) service experience, while possible, is certainly not trivial. The use of customized service delivery clients, as in SECCO, or standardized protocols for service session delivery that report end user service experience are two approaches that can be applied to obtain end user service experience. However, applying either of these approaches to all end user service types may be difficult, especially in the case of proprietary end user services such as Netflix [Netflix, 2013] or Skype [Skype, 2013].

The Aesop approach is also designed so that it can be applied to manage end user services in any networking domain; the *extent* to which it can be applied to an *arbitrary networking domain* is now discussed. In experimentation, optimization of end user service delivery was demonstrated in a Home Area Network test bed and a case study described its application in Mobile Broadband Access networks. The Aesop approach works well in Mobile Broadband access networks or Home Area Networks because it can be applied to manage optimization of end user service delivery in any network where end user service experience and context can be monitored and where end user service sessions can be throttled to use less network resources.

However, if the approach were to be applied for optimization of end user service delivery in a network to which service delivery clients are not directly connected

[2] Sessions with no priority level defined can be assigned a default "basic" priority level and service experience or context need not be collected from them, the approach taken for the "Bronze" service level in the Mobile Broadband case study in Chapter 7.

such as a DSL access network, collection of service experience and context on sessions as well as application of throttling on sessions must be delegated to the sub-networks connected to that network; most likely HANs in the case of a DSL access network. The approach may be very difficult to apply to self-forming networks such as MANETs (Mobile Ad-hoc NETwork). Reporting of service experience and context for each end user service session must be enabled in order to apply the Aesop approach, a factor that must be considered in networks such as Mobile Broadband Access networks where bandwidth is constrained.

The Aesop approach is designed to autonomically optimize end user service delivery in near real time; the *extent* to which it can *optimize service delivery in near real time* is now discussed. In Experiment 3 described in Section 6.5, the performance of a prototype implementation of the approach was evaluated in a real Home Area Network test bed. In both experimental scenarios, the results demonstrated that the average time taken from a measurement being monitored to an optimization being applied based on that observation was less than 40 seconds, well within the *"small number of minutes"* definition of near real time [NGMN, 2007b]. In Experiment 3, average monitoring times, analysis times, and optimization times of the order of 2 seconds, 1 second, and 4 seconds respectively were observed, with average CPU usage at about 20% when four simultaneous service sessions were running. This indicates that spare capacity is available in the prototype implementation of Aesop.

Spare capacity could be used to increase the reporting frequency from terminals and to increase the frequency with which analysis and optimization planning and execution operations to below 10 seconds. This could reduce the execution time of the autonomic loop below 40 seconds. However, in the author's opinion, a reporting frequency of shorter than 10 seconds is unlikely to give a more accurate picture of service experience in the service delivery terminal and may reduce the smoothing effect that averaging over 10 seconds has on momentary spikes in measurements. Indeed, it could be argued that the frequency of execution of the autonomic loop should be a function of the expectations of the user. In certain cases a user may tolerate a reduction in end user service experience of a session for 1-2 minutes if this matches their expectation of the network [Schatz et al., 2011]; in which case an autonomic loop execution time of longer than 40 seconds would be appropriate.

An alternative use of capacity available in an Aesop instance is to increase the number of sessions that are handled. The estimation in Section 7.5.6 of the Mobile

Broadband Access case study suggests that 5,000 simultaneous sessions could be handled if reporting frequency was reduced to 1 report per minute from terminals. However, the engineering improvements suggested in Section 6.7 would be required to achieve such a session load.

Four research objectives were derived in order to support answering the research question above. The following paragraphs details the degree to which each research objective was achieved in this research.

Objective 1: *Identify approaches that are used to manage end user services and technologies that are applicable to autonomic semantic-based optimization of end user service session delivery.*

In order to address this objective, a background study of end user services was conducted to establish the context of the work. Related work was surveyed from four aspects pertinent to this research work: end user service delivery, modelling work related to end user service delivery, semantic techniques applicable to end user service delivery, and semantic approaches applied to autonomic network management.

The background study of end user services in Section 2.2 established that optimization of end user services, the subject of this research, has been raised as a requirement by the community through the TM Forum and the NGMN. It also found that the three most common end user service types are web browsing, video on demand, and file transfer and that DSL access networks and mobile broadband networks are the most common types of access networks to which end user terminals are connected. The study also established that end users on both those access types regularly encounter issues that degrade end user service experience. This study guided the choice of networking domains for experimentation and the case study, as well as the specification of scenarios and input parameters for experimentation.

A state of art survey on end user service delivery in Section 2.3.1 was conducted to *identify approaches that are used to manage end user services*. The survey found that monitoring of telecommunication service delivery context is widely deployed in public networks, and that terminal reporting is increasingly being used to determine end user service experience. Deployed systems and research approaches estimate end user service experience from telecommunication service delivery context but do not analyse end user service experience to determine its compliance with

expectations. The survey found that optimization of end user service delivery is an immature field in practice and in research. As well as influencing the choice of terminal reporting for monitoring end user service experience in experimentation, this survey highlighted the need for investigation of a general approach for optimization of end user service delivery, the subject of this research.

In Section 2.3.2, models, vocabularies, and ontologies, the results of a survey of *applicable to autonomic semantic-based optimization of end user service session delivery* is presented. The survey found that there are many models for telecommunication networks, services and service expectations. Although modelling service context semantically is seen as a promising approach, few applicable semantic models are available. Mapping techniques can be used to translate models in formats such as UM into ontologies. Common vocabularies and ontologies can be applied to represent concepts such as location, time and weather. This survey established the need to design an ontology for the end user service management domain, and identified models, vocabularies, and ontologies to be referenced from that ontology.

Section 2.3.3 describes a survey conducted to *identify technologies that are applicable to autonomic semantic-based optimization of end user service session delivery*. The survey found that semantic reasoning, querying, and rules can be applied. It found that using small models, choosing an appropriate reasoner, and conditioning rules can alleviate performance issues with such techniques. Approaches for handling knowledge with a temporal dimension that could be extended in this research were identified. The findings of this survey influenced the design of the EUSAO ontology and the Aesop knowledge base structure that partitions temporal knowledge to improve the performance of semantic operations.

In Section 2.3.4, the survey of related work on the use of semantics in autonomic network management found that, although some promising work has been undertaken, reference systems and deployments have been very limited. The findings of this survey established that no directly applicable semantically enabled autonomic framework was available that could be extended and applied in this work.

Objective 2: *Establish the EUSAO ontology, a semantic model of end user service expectations, experience, and context.*

The EUSAO ontology described in Chapter 3 addresses Objective 2. Achievement of Objective 2 was confirmed by the use of the EUSAO ontology in the Aesop approach

and in the prototype implementation of Aesop. Aesop uses the EUSAO ontology to structure its knowledge base to partition temporal knowledge. The evaluation of Aesop demonstrated optimization of end user service delivery in near real time, which was achieved using semantic algorithms operating on subsets of knowledge, partitioned using the EUSAO ontology. Achievement of Objective 2 was further confirmed in the Mobile Broadband case study, which demonstrated that the EUSAO ontology could be specialized to model end user service delivery in that domain.

The concepts in the EUSAO ontology related to service expectations, service experience and communication-related service delivery context are well developed because that was the focus of the design work and implementation on EUSAO. The other contextual concept modelled in the ontology is location; its specification is somewhat less well developed because, although individuals were populated during Knowledge Monitoring, they were not referenced in analysis or optimization planning and execution operations. There is much scope for enhancement of the ontology; of the five dimensions of service context identified by Toutain et al. (2011); User, Network, and Device context is partially addressed by the ontology, Social and Physical context is not addressed at all. Such enhancements would allow Semantic Analysis queries to be constructed that consider a fuller service delivery context than mere service experience and telecommunication network context when calculating compliance levels with expectations.

Objective 3: *Design a framework that uses semantic lifting, querying, rules, and reasoning upon the EUSAO ontology, to produce an analysis of the service experience and context reported by service users in near real time.*

The Aesop approach, specifically the *Knowledge Monitoring* and *Semantic Analysis* components of Aesop described in Chapter 4, addresses this objective.

To establish how well objective 3 has been reached, the performance of Aesop's monitoring mechanism was evaluated in a partial implementation of the Aesop prototype in a simulated environment in Experiment 1 as described in Section 6.3. The results of that experiment established that semantic lifting of end user service experience was demonstrable in near real time, showing average monitoring times of 0.5 seconds per terminal report. These results gave an initial positive indication of the applicability of the Aesop approach, leading to a decision to continue the research.

In Experiment 2, described in Section 6.4, evidence was provided that Objective 2 had been achieved in the evaluation of a prototype implementation of the Aesop approach on a Home Area Network test bed with four simultaneous end user service sessions running. The results of this experiment established that Aesop accurately analysed the compliance of sessions with their expectations in near real time in that domain. It demonstrated average monitoring times of 1.3 seconds and average analysis times of 0.5 seconds with average CPU usage below 12%. Knowledge Monitoring executed in a full implementation of the Aesop prototype that was also running Semantic Analysis in Experiment 2, causing observed monitoring times to be longer than those in Experiment 1.

When the Semantic Analysis operation in the Aesop Engine analyses how well the service experience of an end user session matches the expectations of a user, it returns the compliance level as a simple percentage. This approach works very well as input to Optimization Planning and Execution operations, because it provides a single service-agnostic measurement of compliance. However, if Aesop were to be deployed in a real network, it is likely that a much more nuanced analysis result containing at least the input parameters and justification of the compliance level calculation would be expected of the system.

The design of the Aesop Engine relies on the Knowledge Bus to provide safe simultaneous access to the knowledge base. While the Knowledge Bus is designed to have multiple implementations, an implementation that uses a blocking strategy on operations will not scale to large networks. The type of simultaneous safe access to the knowledge base required to scale the Aesop engine to handle large numbers of simultaneous sessions requires a sophisticated Knowledge Bus implementation. The development of such a knowledge bus implementation is not trivial, particularly as it is not at all clear that currently available semantic components such as Jena and the Pellet Reasoner could be adapted to work with such an implementation.

Objective 4: *Develop semantic mechanisms using the EUSAO ontology that can be used to optimize service delivery in near real time by re-allocating network resources to high priority services when the available network resources are insufficient to optimally deliver all running service sessions.*

The full Aesop approach described in Chapter 4, addresses this objective. Assessment of how well objective 4 was reached was examined in Experiment 3

described in Section 6.5. The extent to which the Aesop approach optimized service delivery in near real time was considered earlier in this section in the discussion of how well the full research question was achieved.

The optimization algorithm employed by Aesop used session throttling, and was effective in optimizing end user service sessions. However, there are alternative approaches to throttling that could be employed to optimize end user service sessions. An optimization algorithm could co-operate with a central entity to optimize end user service delivery if such an entity exists [Sterle et al., 2011]. Some service types have in-built mechanisms for managing bandwidth usage [3GPP, 2012c] that optimization algorithms can use [Latre et al., 2009]. In another approach, an optimization algorithm could co-operate with service delivery clients to re-schedule non-interactive sessions such as file transfers to times of low usage or could ask service delivery clients to switch to service consumption modes that consume less bandwidth. Design changes to the Aesop Optimization Planning and Execution operation are likely to be limited to a new `OptimizationExecutionImpl` (See Figure 4-2) the case of such algorithms being used. In an Aesop instance handling a large number of end user service sessions one could envisage a more complex optimization algorithm that executed simultaneous optimization actions on multiple sessions. Such an algorithm would require some minor design changes on the `Optimization Planning` component in Aesop to allow a list of end user service sessions to be passed between the operations of the algorithm shown in Figure 4-6.

8.3 Contributions

The major contribution of this research is **the Aesop approach** and the minor contribution is **the SECCO framework**, as discussed in Section 1.4. The patents that have been provisionally filed and the papers that have been published as a result of this research are also listed in that section.

The design, implementation, and evaluation of **the Aesop approach** has demonstrated autonomic optimization of end user service delivery in near real time and provided results on the effectiveness, consistency and accuracy of the approach in experimentation. The contribution provides *researchers* in the end user service management domain with a reference optimization system for end user service delivery, and with results that can be used to benchmark other approaches. This contribution has the potential to provide a context in which to consider the full set of

end user services running in a network domain to researchers considering network-centric or service specific approaches.

The contribution made by Aesop provides *practitioners* with an approach that can be used to develop commercial systems that autonomically manage end user service delivery in a way that matches expectations expressed by customers, network operators, and service providers. When used with mechanisms in networks, the general applicability of the approach to arbitrary service types and network domains and the flexible manner in which the approach supports customization holds the potential for the approach to see wide use in practice.

Another contribution made by the Aesop approach is that this research has addressed performance concerns described in Section 2.3.3, which have inhibited the widespread adoption of semantic techniques in systems where performance is a critical requirement. This work has shown that semantic techniques can be efficient in a system designed and implemented appropriately. The design of Aesop uses semantics to model the domain, to enable efficient access to knowledge, and to enable Model Driven [Stahl et al., 2006] domain changes implemented using Metadata. Independent autonomic operations around the semantic knowledge base realise the autonomic MAPE loop. The full working prototype developed during this research demonstrated that this design is implementable, and the evaluation of that prototype implementation provides detailed results on the performance of the system. Thus, this research provides a reference design and implementation of a semantic-based autonomic management system that has the potential to be applied to address other problems in other management domains. In addition, the evaluation results provide measurements that can be used as a benchmark for assessing approaches that apply semantics in other work. Experiences in using semantic technologies were documented in Section 6.6 and recommendations to further improve the performance of applications using those technologies were suggested in Section 6.7.

The SECCO framework provides researchers in the end user service management domain with a single framework that can be used in experimentation to specify, launch, control and report on end user service sessions. The framework allows experimental runs to be executed in a uniform, controlled, and repeatable manner. Similarly, SECCO provides practitioners with a framework that can be used to run end user service sessions being used to test and verify components being developed for end user service management in a consistent and controlled manner.

8.4 Future Work

The Aesop approach as currently designed considers service experience and telecommunication service delivery context when assessing the compliance of end user service sessions with expectations. As discussed in the assessment of Objectives 2 and 3 in Section 8.2, there is much scope for considering a much wider set of knowledge when considering how an end user service session is complying with expectations in a certain service delivery context. Future work to consider such improvements to Semantic Analysis is likely to require much careful study of the concepts and relationships that would represent this enhanced service delivery context, and how compliance could be determined from such knowledge. Once the concepts, relationships, and manner of determining compliance is understood, the design changes in the EUSAO ontology and the Semantic Analysis algorithm are likely to be relatively minor. The assessment of Objective 4 in Section 8.2 discussed more sophisticated optimization algorithms that consider simultaneous optimization actions on multiple service sessions. Once the enhanced algorithm is understood, the design changes are also likely to be relatively minor.

The Aesop approach, although designed as a standalone system, does not preclude a distributed deployment or co-operation with other autonomic systems. Future work examining a distributed hierarchical deployment of the Aesop implementation would be interesting for networks that use distributed management. Remote Aesop instances would co-operate with Aesop instances in sub-networks and in a central management system to optimize end user server delivery for the network. In such a deployment, an Aesop implementation could co-operate with service-specific distributed autonomic optimization systems such as that proposed by Latre et al. (2009). Such a deployment would also address the finding from Experiment 3 discussed in Section 6.7 that optimization actions on end user service sessions are more effective when executed in the segment of the network where the impairment occurs rather in the segment of the network where the service is being consumed.

In the prototype implementation of Aesop, the mechanism in the Knowledge Bus for controlling simultaneous access to the Knowledge base is implemented with Java synchronization, which queues the calls to the Knowledge Base. The semantic-based algorithm for Optimization Planning and Execution uses the Knowledge Base extensively and tends to block it, as explained in Section 6.6.3. Future work to examine approaches for implementing the Knowledge Bus that allows simultaneous

operations to run towards the Knowledge Base is merited. One possible approach is to use techniques borrowed from relational databases that address the ACID properties of transactions [Coronel et al., 2009]. A Knowledge Bus implementation that permits simultaneous access to the knowledge base is likely to improve the performance of the prototype implementation of Aesop significantly.

The prototype Aesop system uses the Pellet reasoner to reason over models. As is explained in Section 6.6.3, reasoning consumes a large portion of the time taken to execute a run of a semantic-based algorithm. Research into approaches for improving the performance of the reasoner such as tuning the Pellet reasoner, reducing the amount of knowledge required as input to the reasoner [Tai et al., 2011a], or identifying a more suitable reasoner for the domain is likely to significantly reduce the amount of time required to execute a run of a semantic-based algorithm.

8.5 Final Remarks

Only recently has it has been recognized that, rather than optimizing individual services or networks, it is the set of end user services in a networking domain that should be optimized. This recognition happily coincides with the emergence of techniques in networks that allow individual end user service sessions to be monitored, sufficient capacity in management systems to handle and process the considerable volumes of management data generated by such techniques, and mechanisms that allow individual service session flows to be controlled. It is, therefore, timely to carry out research on end user service delivery optimization, a field that has been somewhat neglected to date. The research work described in this thesis addresses this precise research area and describes the Aesop approach, which can be applied to optimize end user service delivery for any service in any network that supports end user service session monitoring and control. The use of semantic techniques in this work was key to achieving the general applicability of the approach, and may inspire others to look at how the powerful modelling capabilities inherent in such techniques might be applied to solve problems in their research area. This work may provide inspiration for and be of some aid to others in the field of end user service management, so that approaches are researched and deployed that ensure the best possible service experience is always delivered to every user on every service they use.

References

[3GPP, 2009a] 3GPP, "Requirements for Evolved UTRA (E-UTRA) and Evolved UTRAN (E-UTRAN)", *3GPP*, no. 3GPP TS 25.913, Dec 2009

[3GPP, 2009b] 3GPP, "Telecommunication Management; Automatic Neighbour Relation (ANR) Management; Concepts and Requirements", *3GPP*, no. 3GPP TS 32.511, Dec 2009

[3GPP, 2009c] 3GPP, "Telecommunication Management; Configuration Management (CM); Generic Network Resources Integration Reference Point (IRP): Network Resource Model (NRM)", *3GPP*, no. 3GPP TS 32.622, Dec 2009

[3GPP, 2010b] 3GPP, "Transparent end-to-end Packet-switched Streaming Service (PSS); Protocols and codecs", *3GPP*, no. 3GPP TS 26.234, Dec 2010

[3GPP, 2011a] 3GPP, "Telecommunication Management; Configuration Management (CM); Bulk CM Integration Reference Point (IRP): Information Service (IS)", *3GPP*, no. 3GPP TS 32.612, Jan 2011

[3GPP, 2012a] 3GPP, "Network Architecture", *3GPP*, no. 3GPP TS 23.002, Dec 2012

[3GPP, 2012c] 3GPP, "Transparent end-to-end Packet-switched Streaming Service (PSS); Progressive Download and Dynamic Adaptive Streaming over HTTP (3GP-DASH)", *3GPP*, no. 3GPP TS 26.247, Jun 2012

[3GPP, 2012d] 3GPP, "Policy and Charging Control (PCC); Reference points", *3GPP*, no. 3GPP TS 29.212, Dec 2012

[3GPP, 2012e] 3GPP, "Telecommunication Management; Performance Management (PM); Concepts and Requirements", *3GPP*, no. 3GPP TS 32.401, Sep 2012

[3GPP, 2012h] 3GPP, "LTE: The Mobile Broadband Standard", *http://www.3gpp.org/LTE*, Sep 2012, [Accessed: 2012-09-05]

[3GPP, 2013a] 3GPP, "UMTS: Universal Mobile Telecommunications System", *http://www.3gpp.org/Technologies/Keywords-Acronyms/article/umts*, Jan 2013, [Accessed: 2013-01-29]

[Achuthan & O'Meara, 2013] Achuthan, S. & O'Meara, J., "A System for Monitoring Mobile Networks using Performance Management Events", *Integrated Network Management, 2013. IM '13. IFIP/IEEE International Symposium on, to Appear*, May 2013

[Aduna, 2013] Aduna, "User Guide for Sesame 2.7", *http://www.openrdf.org/doc/sesame2/2.7.0-beta1/users/*, 2013, [Accessed: 2013-02-23]

[Agoulmine et al., 2006] Agoulmine, N., Balasubramaniam, S., Botvich, D., Strassner, J., Lehtihet, E. & Donnelly, W., "Challenges for Autonomic Network Management", *Proceedings of 1st IEEE International Workshop on Modelling Autonomic Communications Environments (MACE)*, IEEE, 2006

[Akamai, 2012] Akamai, "The State of the Internet: 3rd Quarter, 2012 Report", *Akamai*, Oct 2012

[Aliu et al., 2013] Aliu, O.G., Imran, A., Imran, M.A. & Evans, B., "A Survey of Self Organisation in Future Cellular Networks", *Communications Surveys Tutorials, IEEE*, vol. 15, no. 1, pp. 336-361, Jan 2013

[Allemang & Hendler, 2008] Allemang, D. & Hendler, J., "Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL", Morgan Kaufmann Publishers Inc., ISBN 0123735564, 9780123735560, 2008

[Apache, 2013a] Apache, "Apache Commons Net Library", *Apache Software Foundation*, 2013

[Apache, 2013b] Apache, "The Apache HTTP Server Project", *Apache Software Foundation*, 2013

[Apache, 2013c] Apache, "Apache Jena Java Framework for building Semantic Web Applications", *Apache Software Foundation*, 2013

[Apache, 2013d] Apache, "The Apache Tomcat Java Servlet Implementation", *Apache Software Foundation*, 2013

[Asghar et al., 2009] Asghar, J., Le Faucheur, F. & Hood, I., "Preserving Video Quality in IPTV Networks", *Broadcasting, IEEE Transactions on*, vol. 55, no. 2, pp. 386-395, Jun 2009

[BBC, 2013] BBC, "The BBC Programmes Ontology", *http://www.bbc.co.uk/ontologies/programmes/2009-09-07.shtml*, 2013, [Accessed: 2013-03-04]

[Broadband Forum, 2010a] Broadband Forum, "CPE WAN Management Protocol", *Broadband Forum*, BBF, no. TR-069, Nov 2010

[Broadband Forum, 2010b] Broadband Forum, "Data Model Template for TR-069-Enabled Devices", *Broadband Forum*, BBF, no. TR-106, Nov 2010

[Bae et al., 2009] Bae, J., Park, H.S., Chin, Y.M. & Park, H.C., "An Identification of Correlation between Network QoS and Video Quality Index for IPTV Service", *Advanced Communication Technology, 2009. ICACT 2009. 11th International Conference on*, vol. 02, pp. 1399-1402, Feb 2009

[Balbas et al., 2009] Balbas, J.J.P., Rommer, S. & Stenfelt, J., "Policy and charging control in the evolved packet system", *Communications Magazine, IEEE*, vol. 47, no. 2, pp. 68-74, Feb 2009

[Batteram et al., 2010] Batteram, H., Damm, G., Mukhopadhyay, A., Philippart, L., Odysseos, R. & Urrutia-Valdés, C., "Delivering Quality of Experience in Multimedia Networks", *Bell Labs Technical Journal*, vol. 15, no. 1, pp. 175-193, 2010

[Bedini et al., 2011] Bedini, I., Matheus, C., Patel-Schneider, P.F., Boran, A. & Nguyen, B., "Transforming XML Schema to OWL Using Patterns", *Semantic Computing (ICSC), 2011 Fifth IEEE International Conference on*, pp. 102-109, Mar 2011

[Bhushan et al., 2004] Bhushan, B., Hall, J., Kurtansky, P. & Stiller, B., "Operations Support System for End-to-End QoS Reporting and SLA Violation Monitoring in Mobile Services Environment", *Quality of Service in the Emerging Networking Panorama*, Springer, pp. 378-387, 2004

[Blajic et al., 2007] Blajic, T., Nogulic, D. & Druzijanic, M., "Latency Improvements in 3G Long Term Evolution", *Proceedings of MIPRO'07, Opatija, Croatia*, May 2007

[Bojars et al., 2008] Bojars, U., Breslin, J.G., Peristeras, V., Tummarello, G. & Decker, S., "Interlinking the Social Web with Semantics", *Intelligent Systems, IEEE*, vol. 23, no. 3, pp. 29-40, May 2008

[Bouch et al., 2000] Bouch, A., Kuchinsky, A. & Bhatti, N., "Quality is in the Eye of the Beholder: Meeting Users' Requirements for Internet Quality of Service", *Proceedings of the SIGCHI conference on Human factors in computing systems*, ser. CHI '00, ACM, ISBN 1-58113-216-6, pp. 297-304, 2000

[Bouten et al., 2012b] Bouten, N., Famaey, J., Latre, S., Huysegems, R., Vleeschauwer, B.D., Leekwijck, W.V. & Turck, F.D., "QoE Optimization Through In-Network Quality Adaptation for HTTP Adaptive Streaming", *Network and Service Management (CNSM), 2012 8th International Conference on*, pp. 336-342, Oct 2012

[Buzan & Buzan, 2006] Buzan, T. & Buzan, B., "The Mind Map Book", New American Library, ISBN 978-0452273221, 2006

[C4DM, 2011] C4DM, "The Centre for Digital Music Event Ontology", *http://motools.sourceforge.net/event/event.html*, 2011, [Accessed: 2011-12-22]

[Ce Junior et al., 2012] Cé Júnior, J., Prudencio, A.C., Willrich, R. & da Silva, M.P., "A Semantic Approach for QoS Specification of Communication Services using QoE Parameters", *Journal of the Brazilian Computer Society*, Springer-Verlag, pp. 1-15, Nov 2012

[Chen et al., 2011] Chen, N., Jiang, X., Wang, C. & Su, J., "Study on Relationship between Network Video Packet Loss and Video Quality", *Image and Signal Processing (CISP), 2011 4th International Congress on*, vol. 1, pp. 282-286, Oct 2011

[Cheng et al., 2008] Cheng, Y., Leon-Garcia, A. & Foster, I., "Toward an Autonomic Service Management Framework: A Holistic Vision of SOA, AON, and Autonomic Computing", *IEEE Communications Magazine*, vol. 46, no. 5, pp. 138-146, May 2008

[Choi et al., 2010] Choi, J., Kwak, S.H., Lim, M.J., Chae, T., Shim, B.K. & Yoo, J.H., "Service Traffic Management System for Multiservice IP Networks: Lessons Learned and Applications", *IEEE Communications Magazine*, vol. 48, no. 4, pp. 58-65, Apr 2010

[Clark & Parsia, 2013] Clark & Parsia, "The Pellet Reasoner", *http://clarkparsia.com/pellet/features*, 2013, [Accessed: 2013-01-05]

[Comreg, 2013] Comreg, "Siteviewer Service: Mobile Sites; Base Stations and Masts", *http://www.askcomreg.ie/mobile/siteviewer.273.LE.asp*, 2013, [Accessed: 2013-01-31]

[Coronel et al., 2009] Coronel, C., Morris, S. & Rob, P., "Database Systems: Design, Implementation, and Management", Cengage Learning, ISBN 978-0538469685, Nov 2009

[Coulouris et al., 2012] Coulouris, G., Dollimore, J., Kindberg, T. & Blair, G., "Distributed Systems: Concepts and Design", Addison-Wesley, ISBN 978-0-13-214301-1, 2012

[DMTF, 2012] DMTF, "CIM Schema: Version 2.33.0", *DMTF*, rev. Version 2.33.0, Jun 2012

[Davy et al., 2007] Davy, S., Barrett, K., Serrano, M., Strassner, J., Jennings, B. & van der Meer, S., "Policy Interactions and Management of Traffic Engineering Services Based on Ontologies", *Network Operations and Management Symposium, 2007. LANOMS 2007. Latin American*, pp. 95-105, Sep 2007

[De Simone et al., 2009] De Simone, F., Naccari, M., Tagliasacchi, M., Dufaux, F., Tubaro, S. & Ebrahimi, T., "Subjective Assessment of H.264/AVC Video Sequences Transmitted over a Noisy Channel", *Quality of Multimedia Experience, 2009. QoMEx 2009. International Workshop on*, pp. 204-209, Jul 2009

[Dischinger et al., 2007] Dischinger, M., Haeberlen, A., Gummadi, K.P. & Saroiu, S., "Characterizing Residential Broadband Networks", *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, ser. IMC '07, ACM, ISBN 978-1-59593-908-1, pp. 43-56, 2007

[Dobrian et al., 2011] Dobrian, F., Sekar, V., Awan, A., Stoica, I., Joseph, D., Ganjam, A., Zhan, J. & Zhang, H., "Understanding the Impact of Video Quality on User Engagement", *Proceedings of the ACM SIGCOMM 2011 Conference*, ser. SIGCOMM '11, ACM, ISBN 978-1-4503-0797-0, pp. 362-373, 2011

[Epiphaniou et al., 2010] Epiphaniou, G., Maple, C., Sant, P. & Reeve, M., "Affects of Queuing Mechanisms on RTP Traffic: Comparative Analysis of Jitter, End-to-End Delay and Packet Loss", *Availability, Reliability, and Security, 2010. ARES '10 International Conference on*, pp. 33-40, Feb 2010

[Ericsson, 2012] Ericsson, "Ericsson Network IQ (ENIQ)", *Ericsson*, 2012

[Etzioni et al., 2011] Etzioni, Z., Feeney, K., Keeney, J. & O'Sullivan, D., "Federated Homes: Secure Sharing of Home Services", *Proc. 2011 IEEE Consumer Communications and Networking Conference (CCNC 2011)*, 2011

[FOAF, 2010] FOAF, "FOAF Vocabulary Specification 0.98", *Friend of a Friend (FOAF) Project*, Aug 2010

[Fallon & Huang, 2011] Fallon, L. & Huang, Y., "GSQR: A Generic Service Quality Reporting Protocol for Terminals", *Integrated Network Management (IM), 2011 IFIP/IEEE International Symposium on*, pp. 1122-1128, May 2011

[Famaey et al., 2010] Famaey, J., Latre, S., Strassner, J. & De Turck, F., "A Hierarchical Approach to Autonomic Network Management", *Network Operations and Management Symposium Workshops (NOMS Wksps), 2010 IEEE/IFIP*, pp. 225-232, Apr 2010

[Feigenbaum et al., 2007] Feigenbaum, L., Herman, I., Hongsermeier, T., Neumann, E. & Stephens, S., "The Semantic web in Action", *Scientific American Magazine*, Scientific American, vol. 297, no. 6, pp. 90-97, 2007

[Fensel et al., 2011] Fensel, D., Facca, F.M., Simperl, E. & Toma, I., "Web Service Modeling Ontology", *Semantic Web Services*, Springer, ISBN 978-3-642-19192-3, pp. 107-129, 2011

[Fiedler et al., 2010] Fiedler, M., Hossfeld, T. & Tran-Gia, P., "A Generic Quantitative Relationship between Quality of Experience and Quality of Service", *Network, IEEE*, vol. 24, no. 2, pp. 36-41, Mar 2010

[Gamma et al., 1994] Gamma, E., Helm, R., Johnson, R. & Vlissides, J., "Design patterns: Elements of Reusable Object-Oriented Software", Addison-Wesley, ISBN 978-0201633610, vol. 206, 1994

[Garcia Clemente et al., 2011] García Clemente, F., Alcaraz Calero, J., Bernal Bernabé, J., Marín Pérez, J., Martínez Pérez, G. & Gómez Skarmeta, A., "Semantic Web-Based Management of Routing Configurations", *Journal of Network and Systems Management*, Springer New York, vol. 19, pp. 209-229, 2011

[Garschhammer et al., 2001] Garschhammer, M., Hauck, R., Hegering, H.G., Kempter, B., Radisic, I., Rolle, H., Schmidt, H., Langer, M. & Nerb, M., "Towards Generic Service Management Concepts: A Service Model Based Approach", *Integrated Network Management Proceedings, 2001 IEEE/IFIP International Symposium on*, pp. 719-732, May 2001

[Groenendijk et al., 2011] Groenendijk, J., Huang, Y. & Fallon, L., "Adaptive Terminal Reporting for Scalable Service Quality Monitoring in Large Networks", *Network and Service Management (CNSM), 2011 7th International Conference on*, pp. 1-5, Oct 2011

[HP, 2012] HP, "HP Service Quality Management Platform (SQM)", *Hewlett Packard*, 2012

[Handurukande et al., 2011] Handurukande, S., Fedor, S., Wallin, S. & Zach, M., "Magneto Approach to Quality of Service (QoS) Monitoring", *Integrated Network Management, 2011. IM '11. IFIP/IEEE International Symposium on.*, May 2011

[Hazelcast, 2012] Hazelcast, "Hazelcast Documentation", *Hazelcast*, Aug 2012

[Heitmann et al., 2009] Heitmann, B., Kinsella, S., Hayes, C. & Decker, S., "Implementing Semantic Web Applications: Reference Architecture and Challenges", *SWESE 2009*, pp. 16, Oct 2009

[Hibberd, 2013] Hibberd, M., "Apple Vetting Operators on LTE Network Performance", *http://www.telecoms.com/54319/apple-vetting-operators-on-lte-network-performance/*, 2013, [Accessed: 2013-02-06]

[Holm-Oste & Norling, 2008] Holm-Oste, G. & Norling, M., "Ericsson's User Service Performance Framework", *Ericsson Review*, no. 1, pp. 43-46, 2008

[Hossfeld et al., 2011] Hoßfeld, T., Schatz, R., Zinner, T., Seufert, M. & Tran-Gia, P., "Transport Protocol Influences on YouTube Videostreaming QoE", *Multimedia (ISM), 2011 IEEE International Symposium on*, pp. 494-499, Sep 2011

[IBM, 2012a] IBM, "Rational Unified Process (RUP)", *IBM*, 2012

[IBM, 2012b] IBM, "IBM Tivoli Netcool Service Quality Manager (SQM)", *IBM*, 2012

[IETF, 1985] IETF, "File Transfer Protocol (FTP)", *IETF*, RFC Editor, United States, no. RFC 959, Oct 1985

[IETF, 1991] IETF, "Management Information Base for Network Management of TCP/IP-Based Internets: MIB-II", *IETF*, RFC Editor, United States, no. RFC 1213, Mar 1991

[IETF, 1998a] IETF, "Real Time Streaming Protocol (RTSP)", *IETF*, ser. Request for Comments, IETF, no. RFC 2326, Apr 1998

[IETF, 1999] IETF, "Structure of Management Information version 2 (SMIv2)", *IETF*, RFC Editor, United States, no. RFC 2578, Apr 1999

[IETF, 2002] IETF, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", *IETF*, ser. Request for Comments, IETF, no. RFC 3411, Dec 2002

[IETF, 2003a] IETF, "RTP: A Transport Protocol for Real-Time Applications", *IETF*, IETF, Jul 2003

[IETF, 2006a] IETF, "The Application/JSON Media Type for JavaScript Object Notation (JSON)", *IETF*, ser. Request for Comments, IETF, no. RFC 4627, Jul 2006

[IETF, 2006b] IETF, "NETCONF Configuration Protocol", *IETF*, ser. Request for Comments, IETF, no. RFC 4741, Dec 2006

[IETF, 2009] IETF, "The Syslog Protocol", *IETF*, no. RFC 5424, Mar 2009

[IETF, 2010b] IETF, "Network Time Protocol Version 4: Protocol and Algorithms Specification", *IETF*, RFC Editor, United States, no. RFC 5905, Jun 2010

[IETF, 2010c] IETF, "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", *IETF*, ser. Request for Comments, IETF, no. RFC 6020, Oct 2010

[IETF, 2011] IETF, "Extensible Messaging and Presence Protocol (XMPP): Core", *IETF*, RFC Editor, United States, no. RFC 6120, Mar 2011

[iOpus, 2013] iOpus, "A Firefox Automator", *https://addons.mozilla.org/en-US/firefox/addon/imacros-for-firefox/*, 2013, [Accessed: 2013-02-20]

[ITU-T, 1994] ITU-T, "Information technology - Open Systems Interconnection - Basic Reference Model: The basic model", *ITU-T*, no. X.200, Jul 1994

[ITU-T, 2005a] ITU-T, "Estimating End-to-End Performance in IP Networks for Data Applications", *ITU-T*, ITU, no. G.1030, Nov 2005

[ITU-T, 2008a] ITU-T, "The E-model: a Computational Model for use in Transmission Planning", *ITU-T*, ITU, no. G.107, Aug 2008

[ITU-T, 2008b] ITU-T, "Quality of Experience Requirements for IPTV Services", *ITU-T*, ITU, no. G.1080, Dec 2008

[ITU-T, 2011] ITU-T, "Internet Protocol Data Communication Service -- IP Packet Transfer and Availability Performance Parameters", *ITU-T*, no. Y.1540, Mar 2011

[Ihm & Pai, 2011] Ihm, S. & Pai, V.S., "Towards Understanding Modern Web Traffic", *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, ser. IMC '11, ACM, ISBN 978-1-4503-1013-0, pp. 295-312, 2011

[Ireland Offline, 2013] Ireland Offline, "Ireland Offline Map of Broadband Coverage in Ireland", *http://irelandoffline.org/map/*, 2013, [Accessed: 2013-01-31]

[Jennings et al., 2007] Jennings, B., van der Meer, S., Balasubramaniam, S., Botvich, D., Ó Foghlú, M., Donnelly, W. & Strassner, J., "Towards Autonomic Management of Communications Networks", *IEEE Communications Magazine*, IEEE, vol. 45, no. 10, pp. 112-121, 2007

[Johansson et al., 2009] Johansson, K., Bergman, J., Gerstenberger, D., Blomgren, M. & Wallén, A., "Multi-carrier HSPA evolution", *Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th*, pp. 1-5, Apr 2009

[Keeney et al., 2007] Keeney, J., Lewis, D. & O'Sullivan, D., "Ontological Semantics for Distributing Contextual Knowledge in Highly Distributed Autonomic

Systems", *Journal of Network and Systems Management*, Springer New York, vol. 15, pp. 75-86, 2007

[Keeney et al., 2010] Keeney, J., Stevens, C. & O'Sullivan, D., "Extending a knowledge-based network to support temporal event reasoning", *Network Operations and Management Symposium (NOMS), 2010 IEEE*, pp. 631-638, Apr 2010

[Keeney et al., 2011] Keeney, J., Boran, A., Bedini, I., Matheus, C.J. & Patel-Schneider, P.F., "Approaches to Relating and Integrating Semantic Data from Heterogeneous Sources", *Web Intelligence and Intelligent Agent Technology (WI-IAT), 2011 IEEE/WIC/ACM International Conference on*, vol. 1, pp. 170-177, Aug 2011

[Kephart & Chess, 2003] Kephart, J.O. & Chess, D.M., "The Vision of Autonomic Computing", *IEEE Computer Magazine*, Jan 2003

[Kim et al., 2009] Kim, D.W., Lim, H.M., Yoo, J.H. & Kim, S.H., "Experience in Developing a Prototype for WiBro Service Quality Management System", *Management Enabling the Future Internet for Changing Business and New Computing Services*, Springer, pp. 241-250, 2009

[Kruchten, 2000] Kruchten, P., "The Rational Unified Process (RUP)", Addison-Wesley, ISBN 978-0201707101, 2000

[Laiho et al., 2006] Laiho, J., Wacker, A. & Novosad, T., "Radio Network Planning and Optimisation for UMTS", John Wiley and Sons, ISBN 978-0-470-01575-9, 2006

[Lamanna et al., 2003] Lamanna, D.D., Skene, J. & Emmerich, W., "SLAng: A Language for Defining Service Level Agreements", *Distributed Computing Systems, 2003. FTDCS 2003. Proceedings. The Ninth IEEE Workshop on Future Trends of*, pp. 100-106, May 2003

[Laner et al., 2012] Laner, M., Svoboda, P., Romirer-Maierhofer, P., Nikaein, N., Ricciato, F. & Rupp, M., "A Comparison between One-Way Delays in Operating HSPA and LTE Networks", *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), 2012 10th International Symposium on*, pp. 286-292, May 2012

[Latre et al., 2008] Latre, S., Simoens, P., De Vleeschauwer, B., Van de Meerssche, W., De Turck, F., Dhoedt, B., Demeester, P., Van Den Berghe, S. & de Lumley, E.G., "Design for a Generic Knowledge Base for Autonomic QoE Optimization in Multimedia Access Networks", *Network Operations and Management Symposium Workshops, 2008. NOMS Workshops 2008. IEEE*, pp. 335-342, Apr 2008

[Latre et al., 2009] Latré, S., Simoens, P., de Vleeschauwer, B., de Meerssche, W.V., de Turck, F., Dhoedt, B., Demeester, P., van den Berghe, S. & de Lumley, E.G., "An Autonomic Architecture for Optimizing QoE in Multimedia Access Networks", *Computer Networks*, vol. 53, no. 10, pp. 1587-1602, 2009

[Lehtihet & Agoulmine, 2008] Lehtihet, E. & Agoulmine, N., "Towards Integrating Management Interfaces", *Network Operations and Management Symposium, 2008. NOMS 2008. IEEE*, pp. 807-810, Apr 2008

[Leijon et al., 2008] Leijon, V., Wallin, S. & Ehnmark, J., "SALmon A Service Modeling Language and Monitoring Engine", *Service-Oriented System Engineering, 2008. SOSE '08. IEEE International Symposium on*, pp. 202-207, Dec 2008

[Linux Bridging, 2013] Linux Bridging, "Linux Bridging Support", *http://www.linuxfoundation.org/collaborate/workgroups/networking/bridge*, 2013, [Accessed: 2013-01-03]

[Linux Sysstat, 2013] Linux Sysstat, "Linux System Statistics Utilities", *http://sebastien.godard.pagesperso-orange.fr/index.html*, 2013, [Accessed: 2013-01-04]

[Lopez de Vergara et al., 2009] López de Vergara, J., Guerrero, A., Villagrá, V. & Berrocal, J., "Ontology-Based Network Management: Study Cases and Lessons Learned",

Journal of Network and Systems Management, Springer New York, vol. 17, pp. 234-254, 2009

[Maier et al., 2009] Maier, G., Feldmann, A., Paxson, V. & Allman, M., "On Dominant Characteristics of Residential Broadband Internet Traffic", *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*, ser. IMC '09, ACM, ISBN 978-1-60558-771-4, pp. 90-102, 2009

[Mischke et al., 2003] Mischke, J., Stiller, B. & Series, T., "Peer-to-Peer Overlay Network Management through AGILE", *Integrated network management VIII: managing it all: IFIP/IEEE Eighth International Symposium on Integrated Network Management (IM 2003), March 24-28, 2003, Colorado Springs, USA*, pp. 337, 2003

[Moore et al., 2009] Moore, D.S., McCabe, G. & Craig, B.A., "Introduction to the Practice of Statistics", WH Freeman, ISBN 978-1-4292-1621-0, 2009

[Mortier & Kiciman, 2006] Mortier, R. & Kiciman, E., "Autonomic Network Management: Some Pragmatic Considerations", *INM '06: Proceedings of the 2006 SIGCOMM workshop on Internet network management*, ACM, ISBN 1-59593-570-3, pp. 89-93, 2006

[Mozilla, 2013a] Mozilla, "The Firebug Firefox Development Tools", *http://getfirebug.com/*, 2013, [Accessed: 2013-02-26]

[Mozilla, 2013b] Mozilla, "The Firefox Web Browser", *http://www.mozilla.org/en-US/firefox/new/*, 2013, [Accessed: 2013-02-20]

[Munoz Frutos et al., 2009] Muñoz Frutos, H., Kotsiopoulos, I., Vaquero Gonzalez, L. & Rodero Merino, L., "Enhancing Service Selection by Semantic QoS", *The Semantic Web: Research and Applications*, ser. Lecture Notes in Computer Science, Springer, vol. 5554, pp. 565-577, 2009

[NGMN, 2007a] NGMN, "Use Cases related to Self Organising Network, Overall Description", *NGMN*, May 2007

[NGMN, 2007b] NGMN, "NGMN Informative List of SON Use Cases", *NGMN*, Apr 2007

[NGMN, 2010] NGMN, "NGMN Top OPE Recommendations", *NGMN*, Sep 2010

[NNEW, 2011] NNEW, "NextGen Network Enabled Weather (NNEW) Ontology", *https://wiki.ucar.edu/display/NNEWD/Data+Models+and+Formats*, 2011, [Accessed: 2011-12-22]

[Nardi & Brachman, 2007] Nardi, D. & Brachman, R.J., "The Description Logic Handbook", Cambridge University Press, ISBN 978-0-521-15011-8, pp. 1-40, 2007

[Netfilter Framework, 2012] Netfilter Framework, "The Linux netfilter.org Packet Filtering Framework", *http://www.netfilter.org*, 2012, [Accessed: 2012-08-25]

[Netflix, 2013] Netflix, "Netflix Video on Demand Service", *http://www.netflix.com*, 2013, [Accessed: 2013-03-12]

[Nyberg & Ronnblom, 2008] Nyberg, M. & Rönnblom, N., "Basing the Key Performance Indicators on User Experience Methods rather than Estimations", *Wireless World Research Forum*, no. WG1 Paper, 2008

[O'Connor & Das, 2011] O'Connor, M.J. & Das, A.K., "A Method for Representing and Querying Temporal Information in OWL", *Communications in Computer and Information Science*, pp. 97-110, 2011

[OECD, 2011a] OECD, "Fixed and Wireless Broadband Subscriptions by Technology", *OECD*, Jun 2011

[OECD, 2011b] OECD, "Average Advertised Download Speeds, by Technology", *OECD*, Sep 2011

[OECD, 2011c] OECD, "Average Advertised Upload Speeds, by Technology", *OECD*, Sep 2011

[OFCOM, 2012] OFCOM, "UK Fixed-Line Broadband Performance, Nov 2011", *Office of Communications*, Feb 2012

[OFCOM, 2011] OFCOM, "Measuring Mobile Broadband in the UK: Performance delivered to PCs via Dongles and Datacards", *Office of Communications*, May 2011

[OMG, 2008] OMG, "Semantics Of Business Vocabulary And Rules (SBVR)", *OMG*, Jan 2008

[OMG, 2010] OMG, "OMG Unified Modeling Language (OMG UML), Superstructure", *OMG*, May 2010

[O'Sullivan et al., 2007] O'Sullivan, D., Wade, V. & Lewis, D., "Understanding as We Roam", *IEEE Internet Computing*, IEEE Computer Society, pp. 26-33, 2007

[Oracle, 2012] Oracle, "Java SE 6 HotSpot Virtual Machine Garbage Collection Tuning", *Oracle*, Dec 2012

[Oracle, 2013] Oracle, "Remote Method Invocation Home", *http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136424.html*, 2013, [Accessed: 2013-02-23]

[Oyman & Singh, 2012] Oyman, O. & Singh, S., "Quality of Experience for HTTP Adaptive Streaming Services", *IEEE Communications Magazine*, vol. 50, no. 4, pp. 20-27, Apr 2012

[Papyrus, 2011] Papyrus, "The Papyrus News Ontology", *http://www.ict-papyrus.eu/?page=newsontology*, 2011, [Accessed: 2011-12-22]

[The PostgreSQL Group, 2013] The PostgreSQL Group, "The PostgreSQL Database", *http://www.postgresql.org/*, 2013, [Accessed: 2013-02-24]

[Qiu et al., 2010] Qiu, T., Feng, J., Ge, Z., Wang, J., Xu, J. & Yates, J., "Listen to Me if you Can: Tracking User Experience of Mobile Network on Social Media", *Proceedings of the 10th annual conference on Internet measurement*, ser. IMC '10, ACM, ISBN 978-1-4503-0483-2, pp. 288-293, Nov 2010

[Quinn & Matusikowa, 2010] Quinn, K. & Matusikowa, K., "System and Method for Transforming a UML Model into an OWL Representation", *Ericsson*, no. 2,207,104, Jul 2010

[Rana et al., 2011] Rana, A.I., Jennings, B., Foghlu, M.O. & van der Meer, S., "Autonomic Policy-Based HAN Traffic Classification using Augmented Meta Model for Policy Translation", *Wireless and Optical Communications Networks (WOCN), 2011 Eighth International Conference on*, pp. 1-8, May 2011

[Rana & Jennings, 2012] Rana, A.I. & Jennings, B., "Semantic Uplift of Monitoring Data to Select Policies to Manage Home Area Networks", *Advanced Information Networking and Applications (AINA), 2012 IEEE 26th International Conference on*, pp. 368-375, Mar 2012

[Raymer et al., 2006] Raymer, D., Strassner, J., Lehtihet, E. & van der Meer, S., "End-to-End Model Driven Policy Based Network Management", *Policies for Distributed Systems and Networks, IEEE International Workshop on*, IEEE Computer Society, ISBN 0-7695-2598-9, vol. 0, pp. 67-70, 2006

[Rodosek, 2003] Rodosek, G.D., "A Generic Model for IT Services and Service Management", *Integrated Network Management, 2003. IFIP/IEEE Eighth International Symposium on*, pp. 171-184, Mar 2003

[Rodrigues et al., 2012] Rodrigues, C., Lima, S.R., Álvarez Sabucedo, L.M. & Carvalho, P., "An Ontology for Managing Network Services Quality", *Expert Systems with Applications*, vol. 39, no. 9, pp. 7938-7946, Jul 2012

[Roman et al., 2005] Roman, D., Keller, U., Lausen, H., de Bruijn, J., Lara, R., Stollberg, M., Polleres, A., Feier, C., Bussler, C. & Fensel, D., "Web Service Modeling Ontology", *Appl. Ontol.*, IOS Press, vol. 1, pp. 77-106, Jan 2005

[SNMP4J.org, 2013] SNMP4J.org, "SNMP4J - The Object Oriented SNMP API for Java Managers and Agents", *http://www.snmp4j.org/*, 2013, [Accessed: 2013-02-28]

[Salvador et al., 2010] Salvador, A., De Vergara, J.E.L., Tropea, G., Blefari-Melazzi, N., Ferreiro, Ã. & Katsu, Ã., "A Semantically Distributed Approach to Map IP Traffic Measurements to a Standardized Ontology", *International*

Journal Of Computer Networks and Communications (IJCNC), vol. 2, no. 1, pp. 13-31, Jan 2010

[Sandvine, 2012] Sandvine, "Global Internet Phenomena Report 2H 2012", *Sandvine*, Nov 2012

[Schatz et al., 2011] Schatz, R., Egger, S. & Platzer, A., "Poor, Good Enough or Even Better? Bridging the Gap between Acceptability and QoE of Mobile Broadband Data Services", *Communications (ICC), 2011 IEEE International Conference on*, pp. 1-6, Jun 2011

[Seo et al., 2011] Seo, S.s., Kwon, A., Kang, J.M. & Hong, J.W.K., "OSLAM: Towards Ontology-based SLA Management for IPTV Services", *Proc. 3rd IFIP/IEEE International Workshop on Management of Future Internet (ManFI 2011)*, May 2011

[Serrano et al., 2009] Serrano, M., Strassner, J. & Foghlu, M.O., "A Formal Approach for the Inference Plane Supporting Integrated Management Tasks in the Future Internet", *Integrated Network Management-Workshops, 2009. IM '09. IFIP/IEEE International Symposium on*, pp. 120-127, Jun 2009

[Shannon, 1949] Shannon, C.E., "Communication in the Presence of Noise", *Proceedings of the IRE*, vol. 37, no. 1, pp. 10-21, Jan 1949

[Sheridan-Smith et al., 2006] Sheridan-Smith, N., O'Neill, T., Leaney, J. & Hunter, M., "A Policy-based Service Definition Language for Service Management", *Network Operations and Management Symposium, 2006. NOMS 2006. 10th IEEE/IFIP*, pp. 282-293, Apr 2006

[Sidibe & Mehaoua, 2008] Sidibe, M. & Mehaoua, A., "QoS Monitoring for End-to-End Heterogeneous Networks Configurations Management", *Automation, Quality and Testing, Robotics, 2008. AQTR 2008. IEEE International Conference on*, vol. 1, pp. 364-368, May 2008

[Simoens et al., 2008] Simoens, P., Latre, S., De Vleeschauwer, B., Van De Meerssche, W., De Turck, F., Dhoedt, B., Demeester, P., van den Berghe, S. & Gilon, E., "Design of an Autonomic QoE Reasoner for Improving Access Network Performance", *Autonomic and Autonomous Systems, 2008. ICAS 2008. Fourth International Conference on*, pp. 233-240, Mar 2008

[Singhal, 2012] Singhal, A., "Introducing the Knowledge Graph: things, not strings", *Google*, May 2012

[Skype, 2013] Skype, "The Skype Communcation Service", *http://www.skype.com*, 2013, [Accessed: 2013-03-12]

[Soldani, 2006] Soldani, D., "Means and Methods for Collecting and Analyzing QoE Measurements in Wireless Networks", *World of Wireless, Mobile and Multimedia Networks, 2006. WoWMoM 2006. International Symposium on a*, pp. 5pp.-535, Jul 2006

[Sommerville, 2010] Sommerville, I., "Software Engineering", Pearson, ISBN 978-0137035151, Apr 2010

[Song et al., 2012] Song, Y., Brennan, R., Lewis, D. & Conlan, O., "Towards a Framework to Support Novice Users in Understanding and Monitoring of Home Area Networks", *Pervasive Computing and Communications Workshops, IEEE International Conference on*, IEEE Computer Society, ISBN 978-1-4673-0905-9, vol. 0, pp. 82-87, Mar 2012

[Staehle et al., 2008] Staehle, B., Binzenhoefer, A., Schlosser, D. & Boder, B., "Quantifying the Influence of Network Conditions on the Service Quality Experienced by a Thin Client User", *Measuring, Modelling and Evaluation of Computer and Communication Systems (MMB), 2008 14th GI/ITG Conference* -, pp. 1-15, Apr 2008

[Staelens et al., 2010] Staelens, N., Moens, S., Van den Broeck, W., Marien, I., Vermeulen, B., Lambert, P., Van de Walle, R. & Demeester, P., "Assessing Quality of Experience of IPTV and Video on Demand Services in Real-Life Environments", *Broadcasting, IEEE Transactions on*, IEEE, vol. 56, no. 4, pp. 458-466, Dec 2010

[Stahl et al., 2006] Stahl, T., Voelter, M. & Czarnecki, K., "Model-Driven Software Development: Technology, Engineering, Management", John Wiley Sons, ISBN 0470025700, 2006

[Stanford University, 2012] Stanford University, "The Protégé Ontology Editor and Knowledge Acquisition System", *http://protege.stanford.edu/*, 2012, [Accessed: 2013-02-12]

[Stankiewicz et al., 2011] Stankiewicz, R., Cholda, P. & Jajszczyk, A., "QoX: What is it Really?", *IEEE Communications Magazine*, vol. 49, no. 4, pp. 148-158, Apr 2011

[Sterle et al., 2011] Sterle, J., Volk, M., Sedlar, U., Bester, J. & Kos, A., "Application-Based NGN QoE Controller", *IEEE Communications Magazine*, vol. 49, no. 1, pp. 92-101, Jan 2011

[Stewart et al., 2011] Stewart, L., Hayes, D.A., Armitage, G., Welzl, M. & Petlund, A., "Multimedia-Unfriendly TCP Congestion Control and Home Gateway Queue Management", *Proceedings of the second annual ACM conference on Multimedia systems*, ser. MMSys '11, ACM, ISBN 978-1-4503-0518-1, pp. 35-44, 2011

[Stollberg & Norton, 2007] Stollberg, M. & Norton, B., "A Refined Goal Model for Semantic Web Services", *Proceedings of the 2nd International Conference on Internet and Web Applications and Services (ICIW 2007), Mauritius*, 2007

[Strassner, 2002] Strassner, J., "DEN-ng: Achieving Business-Driven Network Management", *Network Operations and Management Symposium, 2002. NOMS 2002. 2002 IEEE/IFIP*, pp. 753-766, Aug 2002

[Strassner, 2004] Strassner, J.S., "Policy-based Network Management: Solutions for the Next Generation", Morgan Kaufmann, ISBN 978-1558608597, 2004

[Strassner et al., 2007] Strassner, J., O'Sullivan, D. & Lewis, D., "Ontologies in the Engineering of Management and Autonomic Systems: A Reality Check", *Journal of Network and Systems Management*, Springer New York, vol. 15, pp. 5-11, 2007

[Strassner et al., 2009a] Strassner, J., de Souza, J., van der Meer, S., Davy, S., Barrett, K., Raymer, D. & Samudrala, S., "The Design of a New Policy Model to Support Ontology-Driven Reasoning for Autonomic Networking", *Journal of Network and Systems Management*, Springer New York, vol. 17, pp. 5-32, 2009

[Strassner et al., 2009b] Strassner, J., van der Meer, S., O'Sullivan, D. & Dobson, S., "The Use of Context-Aware Policies and Ontologies to Facilitate Business-Aware Network Management", *Journal of Network and Systems Management*, Springer New York, vol. 17, pp. 255-284, 2009

[Sundaresan et al., 2011] Sundaresan, S., de Donato, W., Feamster, N., Teixeira, R., Crawford, S. & Pescapè, A., "Broadband Internet Performance: A View from the Gateway", *SIGCOMM Comput. Commun. Rev.*, ACM, vol. 41, no. 4, pp. 134-145, Aug 2011

[Sventek et al., 2011] Sventek, J., Koliousis, A., Sharma, O., Dulay, N., Pediaditakis, D., Sloman, M., Rodden, T., Lodge, T., Bedwell, B., Glover, K. & Mortier, R., "An Information Plane Architecture Supporting Home Network Management", *Integrated Network Management (IM), 2011 IFIP/IEEE International Symposium on*, IEEE, pp. 1-8, May 2011

[TMF, 2009] TMF, "Technical Report: Managing the Quality of Customer Experience (MCE)", *TM-Forum*, no. TR148, rev. Version 0.9, Nov 2009

[TMF, 2010a] TMF, "SLA Management Handbook, Release 3.0", *TM-Forum*, no. GB917, rev. Version 0.7, May 2010

[TMF, 2011] TMF, "Information Framework (SID), Release 9.7", *TM-Forum*, no. GB922, rev. Release 9.5, Sep 2011

[TRAMMS, 2008] TRAMMS, "TRAMMS IP Traffic Report", *TRAMMS*, no. 3, Jun 2008

[Tai et al., 2009] Tai, W., Brennan, R., Keeney, J. & O'Sullivan, D., "An Automatically Composable OWL Reasoner for Resource Constrained Devices", *Semantic Computing, 2009. ICSC '09. IEEE International Conference on*, pp. 495-502, Sep 2009

[Tai et al., 2011a] Tai, W., Keeney, J. & O'Sullivan, D., "COROR: A COmposable Rule-Entailment Owl Reasoner for Resource-Constrained Devices", *Rule-Based Reasoning, Programming, and Applications*, ser. Lecture Notes in Computer Science, Springer Berlin Heidelberg, ISBN 978-3-642-22545-1, vol. 6826, pp. 212-226, Jul 2011

[Tai et al., 2011b] Tai, W., Keeney, J. & O'Sullivan, D., "RESP: A Computer Aided OWL REasoner Selection Process", *Semantic Computing (ICSC), 2011 Fifth IEEE International Conference on*, pp. 27-34, Sep 2011

[Martin Theimer, 2013] Martin Theimer, "An HTTP analyzer addon for Firefox", *https://addons.mozilla.org/en-us/firefox/addon/httpfox/*, 2013, [Accessed: 2013-02-20]

[Theisz et al., 2011] Theisz, Z., Salamanca, E. & Cleary, D., "Semantic Digital Model Repository for Communications Network Management Models", *Ericsson*, no. WO/2011/072722, Jun 2011

[Toktar et al., 2007] Toktar, E., Pujolle, G., Jamhour, E., Penna, M. & Fonseca, M., "An XML Model for SLA Definition with Key Indicators", *IP Operations and Management*, ser. Lecture Notes in Computer Science, Springer, vol. 4786, pp. 196-199, 2007

[Toutain et al., 2011] Toutain, F., Bouabdallah, A., Zemek, R. & Daloz, C., "Interpersonal Context-Aware Communication Services", *IEEE Communications Magazine*, vol. 49, no. 1, pp. 68-74, Jan 2011

[Verma & Sheth, 2007a] Verma, K. & Sheth, A., "Semantically Annotating a Web Service", *IEEE Internet Computing*, IEEE Computer Society, vol. 11, no. 2, pp. 83-85, Mar 2007

[VideoLAN, 2013] VideoLAN, "VLC Media Player", *The VideoLAN Organization*, 2013

[Vitvar et al., 2007b] Vitvar, T., Mocan, A., Kerrigan, M., Zaremba, M., Zaremba, M., Moran, M., Cimpian, E., Haselwanter, T. & Fensel, D., "Semantically-enabled Service Oriented Architecture: Concepts, Technology and Application", *Service Oriented Computing and Applications*, Springer, vol. 1, no. 2, pp. 129-154, Jun 2007

[Vodafone, 2013] Vodafone, "Our Network", *http://www.vodafone.com/content/index/uk_corporate_responsibility/our_network.html*, 2013, [Accessed: 2013-01-28]

[Vodafone, 2013] Vodafone, "Vodafone Smart Feedback App", *https://play.google.com/store/apps/details?id=ie.vodafone.sf&hl=en*, 2013, [Accessed: 2013-02-06]

[W3C, 1999a] W3C, "XML Path Language (XPath)", *W3C*, 1999

[W3C, 1999b] W3C, "XSL Transformations (XSLT)", *W3C*, 1999

[W3C, 2001] W3C, "Web Services Description Language (WSDL) 1.1", *W3C*, 2001

[W3C, 2004a] W3C, "Resource Description Framework (RDF) Model and Syntax", *W3C*, 2004

[W3C, 2004b] W3C, "OWL Web Ontology Language: Parsing OWL in RDF/XML", *W3C*, vol. W3C Working Group Note, Jan 2004

[W3C, 2004c] W3C, "SWRL: A Semantic Web Rule Language Combining OWL and RuleML", *W3C*, vol. W3C Member Submission, May 2004

[W3C, 2006a] W3C, "Time Ontology in OWL", *W3C*, Sep 2006

[W3C, 2006b] W3C, "WGS84 Geo Positioning: an RDF Vocabulary", *W3C*, Feb 2006

[W3C, 2007] W3C, "Semantic Annotations for WSDL and XML Schema", *W3C*, Aug 2007

[W3C, 2008] W3C, "SPARQL Query Language for RDF", *W3C*, vol. 4, Jan 2008

[W3C, 2009] W3C, "OWL 2 Web Ontology Language Document Overview", *W3C*, Oct 2009

[Yamagishi & Hayashi, 2008] Yamagishi, K. & Hayashi, T., "Parametric Packet-Layer Model for Monitoring Video Quality of IPTV services", *Communications, 2008. ICC '08. IEEE International Conference on*, IEEE, vol. 19, pp. 110-114, May 2008

[Ye et al., 2007] Ye, J., Coyle, L., Dobson, S. & Nixon, P., "Ontology-Based Models in Pervasive Computing Systems", *The Knowledge Engineering Review*, vol. 22, no. 04, pp. 315-347, 2007

[Zrelli et al., 2012] Zrelli, S., Ishida, A., Okabe, N. & Teraoka, F., "ENM: A service oriented architecture for ontology-driven network management in heterogeneous network infrastructures", *Network Operations and Management Symposium (NOMS), 2012 IEEE*, pp. 1096-1103, Apr 2012

Semantic-based Service Analysis and Optimization

A thesis submitted to the

University of Dublin, Trinity College,

for the degree of

Doctor of Philosophy

Liam Fallon

Knowledge and Data Engineering Group,

Department of Computer Science,

Trinity College, University of Dublin,

Ireland

2013

Volume II

Appendices

Appendix A
End User Service Delivery on DSL and Mobile Broadband Access Networks

A.1 Introduction

This appendix examines service delivery of the three most popular end user services, namely video on demand, web browsing, and file transfer, on DSL access and mobile broadband access networks.

A survey of the most heavily used end user services on the Internet appears in Section A.2. Section A.3 summarises findings from literature that examines how changes in network context can affect the experience of users of those services. DSL connected Home Area Networks and Mobile Broadband Connected terminals are two of the most widely used access network domains for end user services [OECD, 2011a]. Findings from literature describing the likely expectations, experience, and network context of end user service delivery in those networking domains is presented in Section A.4 and Section A.5 respectively.

A.2 End User Services on the Internet

Since the middle of the last decade there has been a large shift in the end user services that generate the most Internet traffic. In the mid 2000s peer-to-peer file transfer services generated by far the most traffic [TRAMMS, 2008]. By 2009, HTTP services had overtaken peer-to-peer file transfer services and generated over 50% of Internet traffic [Maier et al., 2009], largely driven by the use of HTTP to carry video services [Ihm & Pai, 2011]. Because of the emergence of streaming services over HTTP, recent reports on traffic usage have categorized HTTP traffic as web (non streaming HTTP) and streaming (Video and Audio traffic over HTTP such as YouTube as well as RTMP, MPEG, and proprietary streaming protocols such as those used by Netflix) [Sandvine, 2012][Akamai, 2012].

In the 2009 study of a European DSL broadband access network by Maier et al. (2009), 13% of traffic was generated by peer-to-peer file transfer services, 22.6% by

web browsing, and 35% by streaming services. The Sandvine (2012) report, which is compiled using traffic statistics submitted by Sandvine's customers, shows the trend towards streaming services continued and in 2012 streaming services dominated traffic usage in North American fixed access networks observed, generating 58.6% of traffic. Web browsing and peer-to-peer services generate 12.7% and 12.0% of traffic respectively. It is interesting to note that Netflix alone accounts for 33% of observed traffic.

The Akamai (2012) State of the Internet report shows the service usage in mobile broadband networks globally for Quarter 3, 2012 as reported by Akamai's partner Ericsson. Although the access technology is different, the ranking of services generating the most traffic is similar. Streaming services generate 29% of traffic; with web browsing and peer-to-peer file transfer services generating 14% and 20% of traffic respectively. The Sandvine report for end user service usage on mobile broadband networks in North America shows a similar ranking; streaming services consume 49.9% of bandwidth, with web browsing services consuming 26.5% of bandwidth. Peer-to-peer file transfer services did not feature in the top five services consuming bandwidth.

A.3 Impact of Network Context on Services

This section looks at how the perceived quality of video on demand, web browsing, and file transfer service sessions are affected by congestion or impairments, such as those introduced by congestion control and queuing mechanisms [Epiphaniou et al., 2010][Stewart et al., 2011], on a broadband access network connection.

A.3.1 Video on Demand

Asghar et al. (2009) state that standard definition and high definition MPEG4 video streams require bandwidths of between 2-4 Mbit/s and 6-12 Mbit/s respectively. They also point out that video streams are sensitive to packet loss, particularly to I-frame loss where loss of a single frame can cause freezes of the order of seconds [Chen et al., 2011]. Bae et al. (2009) show a sharp linear increase in video block distortion and a less sharp increase in blurriness as packet loss and errors increase in video sessions carried over RTP. They show, however, that delay has little effect on video quality because video players employ buffering to overcome delay.

Dobrian et al. (2011) and Hossfeld et al. (2011) have examined how network QoS impacts user perception of quality and these are discussed in the following paragraphs.

Dobrian et al. (2011) studied how video quality affects user engagement by examining the results of 2 million views by 1 million viewers from several content providers in the autumn of 2010. They examined how the time scales of user engagement for short video on demand clips of 2-5 minutes, long video on demand clips of at least 35 minutes, and live content varied as quality metrics varied.

The authors' analysis shows that for long video on demand clips, there is a strong linear negative correlation between user engagement and buffering ratio (the percentage of session time spent buffering) and a strong linear positive correlation between user engagement and rendering quality. The rate of buffering events is also important; user engagement drops from over 40 minutes to around 10 minutes as the rate of buffering events increases from 0 per minute to 0.2 per minute. There was little correlation between user engagement and either join time or bit rate. If join time is long, users simply do not view the video at all.

Hossfeld et al. (2011) assess how transport protocols affect the QoE of YouTube video streams. They point out that the transport protocol is important: packet loss on the connection-oriented TCP protocol results in content being stalled due to buffering as the transport protocol retransmits the lost packets, when the connectionless UDP is impacted by packet loss, picture quality suffers because the protocol makes no effort to recover the lost packets. The authors carried out a user survey of the QoE of YouTube videos (transported over TCP) using a crowd sourcing approach and correlated the survey results with the results of a previous study by De Simone et al. (2009) where video streams were transported over UDP. They showed an exponential relationship between packet loss and QoE. As packet loss ratio increases from 0% to 2% MOS values decrease from 5 to 2, but an increase in packet loss from 2% to 10% causes MOS to decrease from 2 to 1. They also find that video resolution has only a minor influence on Quality of Experience.

In summary, frame freezes and picture distortion caused by packet loss and errors in the underlying network has a large affect on the quality of experience of Video on Demand service sessions, that reduction in video quality has a much less severe affect, and that large delays in the network have little affect as long as video on

demand sessions start a small number of seconds after they are requested. These insights are used in experimentation to frame the video on demand service expectations specified in Section 6.2.2 and the video on demand aspects of the experimental scenarios in Section 6.2.3.

A.3.2 Web Browsing

The Sundaresan et al. (2011) study examined the influence of changes in throughput and latency on the download time of small web pages with an average size of 125KB. As one would expect, there is a relationship between throughput and page load time. At throughputs greater than 10 Mbit/s, the page load time remains constant at about 0.8s. As throughput decreases, page load time increases in a roughly exponential manner, taking about 1s at a throughput of 5Mbit/s. At 2Mbit/s, page load increases to 1.3s, increasing to 1.7s at 1Mbit/s. The results also show a roughly linear relationship between page load time and latency. Page load time is around 0.8s with a latency of 10ms. It increases to about 1.1s as latency increases to 50ms. Minimum download time is more sensitive to latency, increasing by about 50% from 0.45s to 0.75s as latency increases from 10ms to 40ms.

The OFCOM (2012) report shows average web page load times of about 0.7s and peak load times of about 2.2s on connections advertised as being "up to" 20Mbit/s. The measurements were collected using similar methodology to that used in Sundaresan et al. and are consistent with that study's results.

Ihm & Pai (2011) used five years of real data collected from a web proxy system to study web traffic from four countries. They observe a doubling of the average size and average number of objects on web pages between 2006 and 2010. To compensate for this increasing size and complexity, modern web browsers use parallel sessions to optimize page loading. Parallel sessions are used to load elements on a page, with observations from 2010 showing 50% of users holding more than 5 concurrent sessions while browsing the web. In 2010, average page loading latency was 6s, 9s, 10.5s, and 13s for the four studied countries.

Staehle et al. (2008) studied the influence of packet loss and delay on user tasks such as typing, scrolling, and selecting on office applications running in a thin client using Citrix products. The results show that there is a linear relationship between both delay and packet loss and the amount of time it takes to complete a task on a thin client, with delay generally having a bigger influence than packet loss. One

finding was that a delay increase of 200ms could increase the time taken to execute a menu action by 30%. A user study carried out by the authors showed that an increase in time taken to execute tasks correlated with a decrease in user satisfaction "to a satisfying degree".

In summary, there is a linear relationship between web page load times and latency. The relationship on web page load times as throughput decreases is exponential. Modern web browsers use parallel communication sessions to increase the speed of web page loads at the expense of using more bandwidth. Delay is an important factor in carrying out tasks on interactive web pages; a small increase in delay can cause a large increase in the time taken to execute a task. The web browsing service expectations specified in Section 6.2.2 and the web browsing aspects of the experimental scenarios in Section 6.2.3 are specified in the context of these factors.

A.3.3 File Transfer

When compared with the Video on Demand or Web Browsing services, File Transfer over FTP is a simple service where a single metric, file transfer time, can be used to express both the service expectation and service experience.

The parameter that affects file transfer time is the available bandwidth for the service. Network delay and jitter have little impact because for files of any significant size, the transfer time is many orders of magnitude greater than network delay and jitter times. The experienced quality of the file transfer service on is directly correlated to download speed. Download speed is discussed in Section A.4 for DSL connected Home Area networks and in Section A.5 for Mobile Broadband Access networks.

The file transfer service expectations specified in Section 6.2.2 and the file transfer aspects of the experimental scenarios in Section 6.2.3 are specified in bandwidth consumption terms.

A.4 Service Delivery Context in DSL-Connected HANs

As of 2011, DSL was the dominant fixed broadband access technology worldwide with over 50% of subscribers using it [OECD, 2011a]. In the following paragraphs, relevant information relating to the characteristics and performance of such connections are discussed.

A number of reports and studies are available that describe measurements carried out on real DSL networks. The average advertised maximum download and upload speeds for DSL worldwide are 14.8Mbit/s and 1.9Mbit/s respectively [OECD, 2011b][OECD, 2011c]. OFCOM, the UK telecommunication regulator, issues regular reports on fixed residential broadband performance in the UK [OFCOM, 2012] summarising data collected by their partner SamKnows from special devices installed in over 1700 customer homes. The relevant report findings for DSL connections advertised as "up to" 20Mbit/s are shown in Table A-1, which shows that the actual average download and upload speeds vary considerably from advertised maximum speeds.

Table A-1: UK DSL Advertised 20Mbit/s DSL Residential Broadband by ISP, November 2011

Parameter	Lowest ISP	Highest ISP	Comment
Average Download Speed	5.8MBits/s	11Mbit/s	
Average Upload Speed	0.45Mbits/s	1.05Mbits/s	
Average Packet Loss	0.1%	2.2%	1.5% to 2.2% on one network at peak time
Average Latency	19ms	39ms	
Average Jitter	0.5ms	1.5ms	

The report also shows considerable spread in the actual achieved maximum and average download speed for subscribers. Table A-2 shows that 17% of customers have an average download speed 2Mbit/s or less and 40% have an average download speed of 4Mbit/s or less.

Table A-2: UK Ad. 20Mbit/s DSL Residential Broadband Download Spread, November 2011

Measure	Up to 2Mbit/s	> 2Mbit/Sec to 4Mbit/s	>4Mbit/Sec to 10Mbit/s	>10Mbit/s
Maximum Download Speed	14%	23%	37%	26%
Average Download Speed	17%	23%	36%	24%

Sundaresan et al. (2011) analyse results from 2010 and 2011 gathered from homes in the USA by 4,200 SamKnows devices and much more detailed results gathered by 16 BISMark devices deployed in homes in the Atlanta area. Homes connected using DSL and cable connections were monitored. Although parameters are not detailed by access technology, the observed parameters are largely in line with the OFCOM report except that achieved download and upload speeds are

roughly in line with advertised rates, perhaps due to a different regulatory regime in the USA. This study analyses and explains some observed impairments in some detail.

At peak hours, performance deteriorates. Some subscribers on a particular ISP saw a drop of 40% in throughput at peak hours. All subscribers saw increased variability in throughput at peak times, with a standard deviation of up to 0.3 on normalized throughput. Packet loss also increases at peak time, again roughly in line with the OFCOM figures.

Latency increases very significantly when the access link is heavily loaded because packets are queued in buffers. This effect is more pronounced on the uplink than on the downlink because the downlink is usually much faster than the uplink and the home network usually has a much higher capacity than the uplink, causing queues to build up. Downstream latency can increase by a factor of 10 to 50, bringing it into the range of hundreds of milliseconds. Upstream latency can increase by a factor of 60 to 160, bringing it into the range of seconds. In the worst case, an upstream latency of more than 10 seconds was observed.

DSL Latency can vary significantly with line quality and can reach values of 50ms to 60ms on poor quality lines due to the need for more complex error avoidance mechanisms such as interleaving.

In the study by Dischinger et al. (2007) a different methodology was used. In this study published in 2007, 1093 DSL broadband links with advertised speeds below 10Mbit/s were probed with packet trains sent from the network. Even though the study was carried out on connections with slower speeds than those encountered today, the findings are interesting and relevant. In the study, tests were carried out towards last hop routers and to residential hosts behind those last hop routers and the results were compared.

The results show that the broadband access network is indeed the bottleneck in Internet access for residential hosts. The bandwidth available to the Internet at a residential host is about half what is available at the last hop router. Latency at residential host is 10 times higher than at the last hop router. Packet loss rates at residential hosts are 20 times higher than at the last hop router.

The impact of packet buffering in load conditions on latency was also observed in this study. Under load, downstream delays of between 100ms and 400ms and

upstream delays of between 200ms and 6000ms were observed. Packet loss rates were in line with the OFCOM report and with Sundaresan.

Maier et al. (2009) studied the characteristics of residential broadband Internet traffic by carrying out passive observations on more than 20,000 DSL lines of a major European operator in 2008 and 2009. They confirm that the delay introduced by the broadband access network exceeds the delay in the Internet by more than a factor of 2:1. They also observed that the 90th and 99th percentile for latency is 278ms and 1328ms respectively, users will occasionally encounter significant latency on connections.

In summary, the background material examined in this section shows that DSL access networks are the most widespread fixed access network type, making this network access type a reasonable choice for experimentation, as is explained in Section 6.2.1. DSL access network connection characteristics such as download speed, latency and packet loss can vary by location and time of day. Significant numbers of subscribes can experience download speeds of less than 2Mbit/s. DSL access networks contribute the most to overall latency on sessions, and latency can increase to seconds at times of heavy load. The background material in this section shows that continuous and transient network impairment events do occur in DSL access networks, thus validating the congestion and impairment experimental scenarios described in Section 6.2.3. The DSL access network characteristics detailed in this section are used to specify the values for the characteristics of the experimental degraded network connections described in Section 6.2.1.

A.5 Service Delivery Context in Mobile Broadband Access Networks

In 2011 there were 403 million wireless broadband subscriptions globally, 28% more than the total number of fixed subscriptions and over twice the number of DSL subscriptions [OECD, 2011a]. In the following paragraphs, relevant information relating to the characteristics and performance of connections on UMTS mobile broadband networks (Universal Mobile Telecommunications System) [3GPP, 2013a], commonly known as 3G networks. Independent information on the characteristics of LTE (Long Term Evolution) [3GPP, 2012h] mobile networks, commonly known as 4G networks, is not readily available because LTE networks have only been deployed on a large scale since 2012.

The average speed and average highest speed measured by Akamai and Ericsson on mobile broadband connections for 85 mobile networks in 55 countries worldwide is 2.13Mbit/s and 11.9Mbit/s respectively [Akamai, 2012]. OFCOM, the UK telecommunication regulator, carried out a comprehensive study on the characteristics of mobile broadband access connections from September to December 2010 [OFCOM, 2011]. They measured network performance at 97 fixed locations, and carried out case studies as five day drive tests in four regions with different characteristics: dense urban city, urban sprawl, provincial town, and rural and semi-rural counties. As well as presenting the measurement results, the report compares mobile broadband performance with that of fixed access performance, and draws conclusions on the implication of that performance for end user services.

Figure A-1 shows the data from Figure 3.51 of the OFCOM report [OFCOM, 2011]. It shows that mobile broadband connections have poorer characteristics than DSL connections for all measurements except upload speeds. Mobile broadband characteristics deteriorate by between 10% and 20% at peak times, whereas fixed connections show a deterioration of less than 10% at peak times. Latency and web page download times are significantly longer on mobile broadband connections.

Figure 3.51 Comparison of fixed and mobile broadband performance

	Download speed (Mbit/s)		Upload speed (Mbit/s)		Latency (milliseconds)		*Web page download (milliseconds)	
	24-hour	Peak (8-10pm)	24-hour	Peak (8-10pm)	24-hour	Peak (8-10pm)	24-hour	Peak (8-10pm)
Mobile (3G & HSPA connections)	2.1	1.7	0.7	0.6	117	128	2,508	3,196
ADSL 'up to' 8Mbit/s	3.4	3.1	0.5	0.5	47	50	430	473
ADSL 'up to' 20/24Mbit/s	6.2	6.0	0.7	0.7	41	44	227	299
Cable 'up to' 10Mbit/s	9.6	9.2	0.5	0.5	30	37	190	236
Cable 'up to' 20Mbit/s	18.0	17.3	0.9	0.9	32	37	190	213
Cable 'up to' 50Mbit/s	45.8	45.1	2.3	2.6	27	29	164	177
FTTC 'up to' 40Mbit/s	30.4	26.3	7.4	7.5	25	27	139	157

** Results for webpage downloading are not directly comparable as the mobile broadband tests measured only the time to download the html code of the web pages, while the fixed-line broadband tests included downloading a test page containing both html and images. The data may therefore indicate slower performance for fixed-line web page download than would be the case if identical tests to those used for mobile broadband had been deployed.*

Figure A-1: Comparison of Fixed and Mobile Broadband Performance, from [OFCOM, 2011]

The authors of the report point out that slow web page downloads are caused by low bandwidth and by high latency. Latency causes long DNS lookup times, as shown in Table A-3, which impacts the load time of web pages.

Connection failures, shown in Table A-3, are another important metric in mobile broadband networks. Due to the nature of mobile broadband access networks, a user may not be able to establish a broadband connection due to insufficient signal strength or sufficient capacity being unavailable. In contrast, fixed access connections are usually permanently connected to the network. The packet loss and jitter measurements in Table A-3 are comparable with or better than those observed on DSL connections, see Table A-1.

Table A-3: Mobile Broadband Characteristics Measured by OFCOM

Measurement (Averages)	Value
Failed First Time Connection	4.1%
DNS Resolution Times	100ms
Packet Loss on Download	0.6%
Packet Loss on Upload	0.4%
Jitter on Download	0.5ms
Jitter on Upload	1ms

The signal strength of a mobile broadband access connection significantly influences the characteristics of the connection. As shown in Figure A-2, connections with high signal strength enjoy download speeds that over five times those of connections with low signal strength (RSCP, Received Signal Code Power) [Laiho et al., 2006].

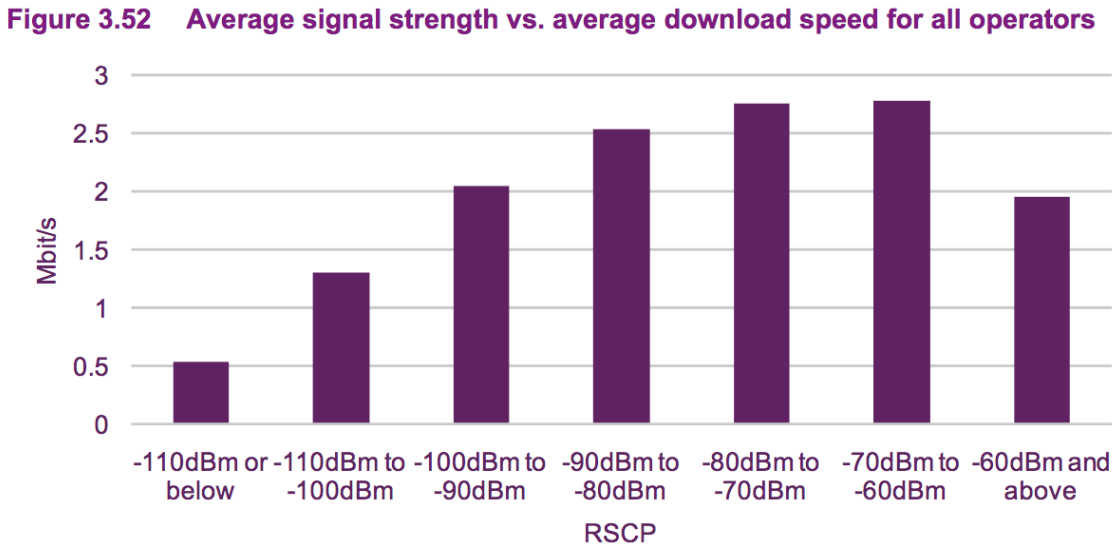


Figure A-2: Comparison of Download Speeds as Signal Strength Varies, from [OFCOM, 2011]

The bar chart in Figure A-3 shows how download speed varies by region. While, as one would expect, the highest proportion of slow connections were observed in rural areas, there are significant numbers of slow connections in built up areas. The authors of the report point out that capacity can be limited in urban areas and that contention may have been a factor causing slower speeds on some connections in urban areas.

The report points out that because the performance of mobile broadband is less than fixed broadband services, applications such as web browsing, file download, voice over IP, video streaming, and on line gaming are unlikely to perform as well on mobile broadband connections as on fixed connections. The report goes on to state that mobile broadband connections and fixed connections serve different needs; mobile broadband may be appropriate for users require connectivity at different locations and for users where fixed connections are not available. A study by Schatz et al. (2011) found that users found service sessions with a lower quality of experience acceptable when they were using mobile broadband connections in the field than in laboratory conditions, indicating that the context in which a user experiences a service can affect their expectations.

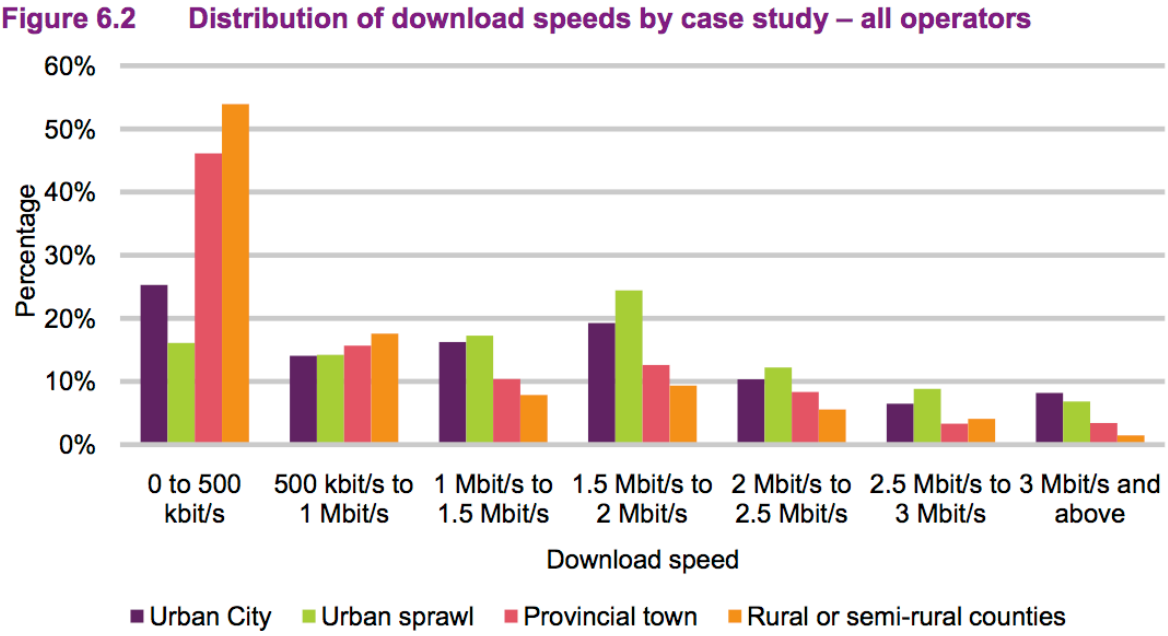


Figure A-3: Comparison of Download Speeds for Case Studies, from [OFCOM, 2011]

Connection failures and low bandwidth occur in mobile broadband access networks for two reasons, distance from a base station and contention for available bandwidth. As Shannon's theorem [Shannon, 1949] states, the transmission rates that are possible decrease as noise increases. Transmission over longer distances

introduces more noise, thus decreasing bandwidth [Laiho et al., 2006] [Aliu et al., 2013]. In mobile broadband networks, because all users in a cell share the available bandwidth, the average user throughput is inversely proportional to the load on a cell [Johansson et al., 2009].

Latency is known to be an issue in UMTS networks [Blajic et al., 2007]. The original release of UMTS showed latency values of approximately 120 milliseconds, which has improved to around 80 milliseconds with HSPA (High Speed Packet Access) enhancements to UMTS networks. Reducing the contribution of the mobile broadband access network to the order of 10 milliseconds is a primary requirement of LTE [3GPP, 2009a]. In LTE, both the radio access network and core network have been improved to reduce latency, as shown in recent comparison of UMTS and LTE networks by Laner et al. (2012). They measured round trip times of between 20 and 40 milliseconds on mobile broadband connections using LTE networks.

In summary, end user service sessions using mobile broadband access connections are more likely to experience degradation in service than sessions using fixed access connections because mobile broadband access networks exhibit lower bandwidth and higher latency than in fixed networks. The location of the terminal and the amount of traffic in the cell are the most important factors that affect the bandwidth that a session experiences. High latency is inherent in mobile broadband networks, which reduces the service experience of conversational services and online gaming. Recent technological improvements have substantially reduced latency values. The need for optimization of end user service session delivery in mobile access networks is even more acute than in DSL access networks, and is the reason why this access type was chosen for examination in the case study in Chapter 7.

Appendix B
The EUSAO Ontology

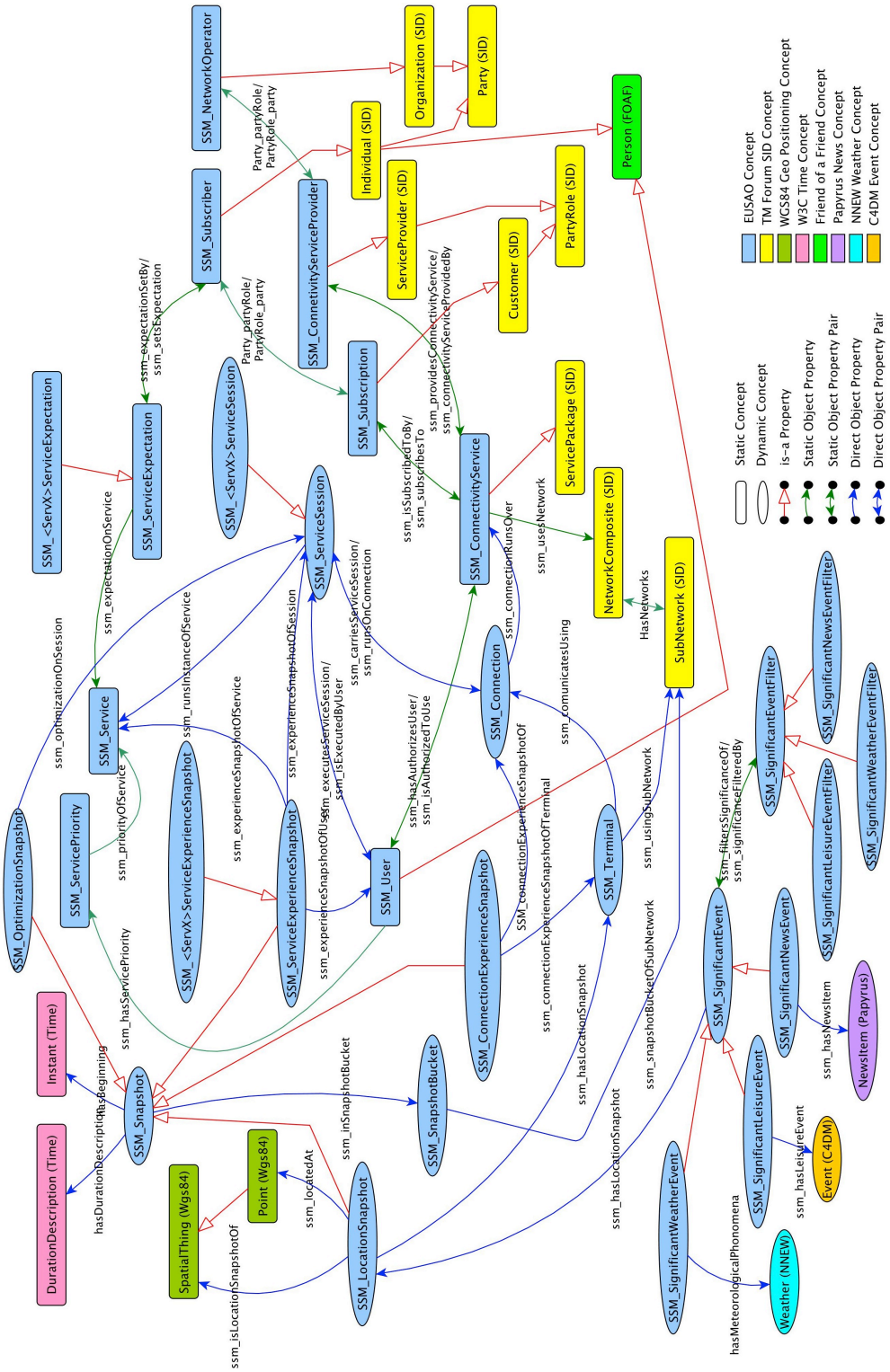


Figure B-1: The EUSAO Ontology

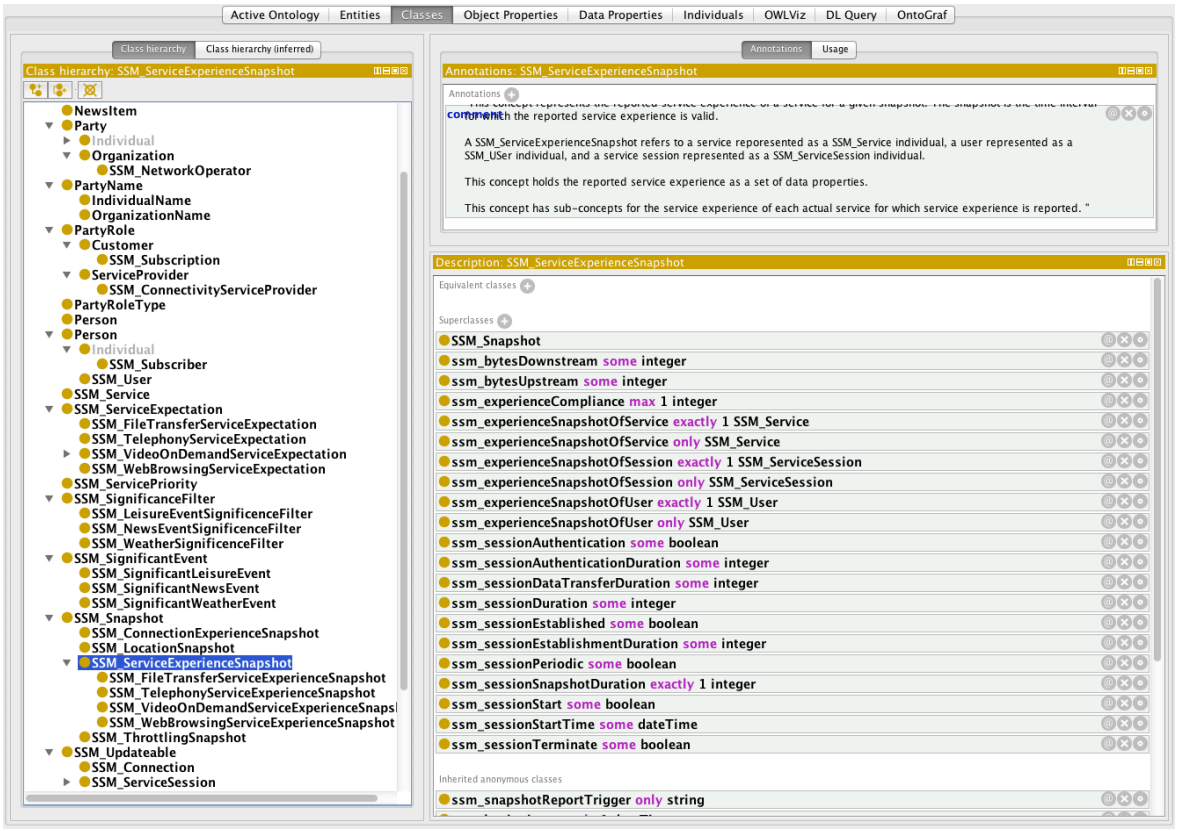


Figure B-2: The EUSAO Ontology in Protégé showing Concepts as Classes

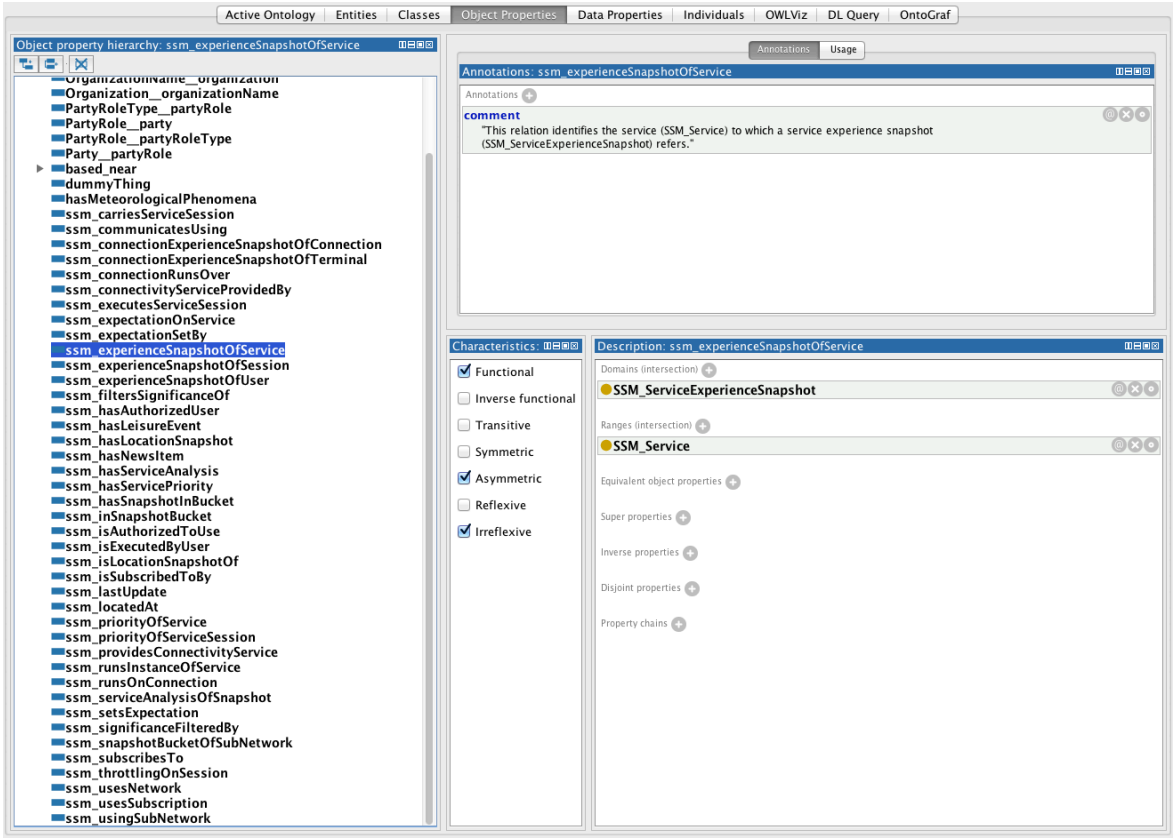


Figure B-3: The EUSAO Ontology in Protégé showing References as Object Properties

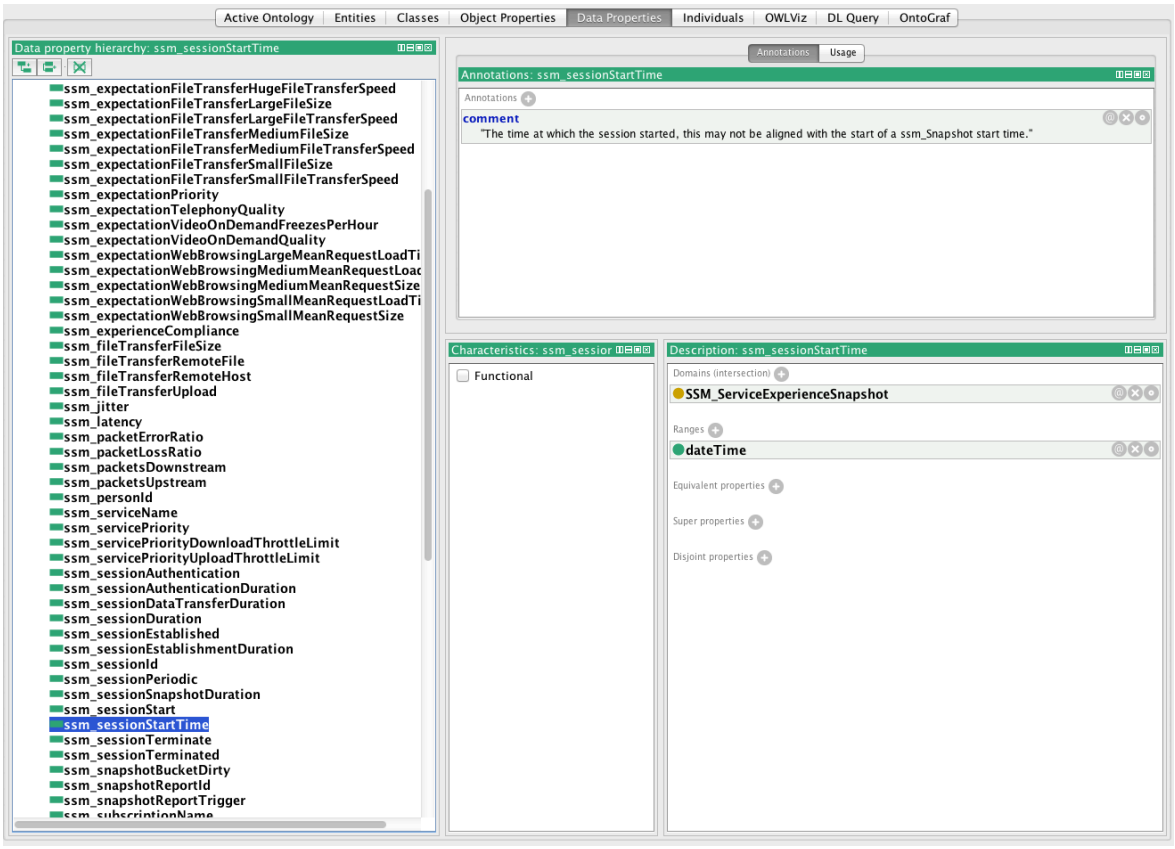


Figure B-4: The EUSAO Ontology in Protégé showing Attributes as Data Properties

Appendix C
SPARQL Queries and SWRL Rules

C.1 SWRL Rules for Knowledge Loading

```
ssm_hasServicePriority(?user, ?prio),
ssm_isExecutedByUser(?sess, ?user),
ssm_priorityOfService(?prio, ?serv),
ssm_runsInstanceOfService(?sess, ?serv) ->
 ssm_priorityOfServiceSession(?sess, ?prio)
```

Listing C-1: SWRL Rule for Deducing Session Priority

```
ssm_experienceSnapshotOfSession(?snapshot, ?session),
ssm_sessionTerminate(?snapshot, true) ->
 ssm_sessionTerminated(?session, true)
```

Listing C-2: SWRL Rule for Deducing Session Termination

C.2 SPARQL Queries for Semantic Analysis

```
PREFIX rdfs:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX saTBox: <http://aesop/ServiceAnalysisTBox.owl#>
PREFIX xsd:    <http://www.w3.org/2001/XMLSchema#>
PREFIX sid:    <http://example.org/SID_Phase_VII_Models#>

SELECT ?service ?expectation ?session ?snapshot WHERE
{
?snapshot rdfs:type saTBox:SSM_ServiceExperienceSnapshot .
?snapshot     saTBox:ssm_experienceSnapshotOfSession ?session .
?snapshot     saTBox:ssm_experienceSnapshotOfService ?service .
?session      saTBox:ssm_runsOnConnection            ?connection .
?connection   saTBox:ssm_usesSubscription            ?subscription .
?subscription sid:PartyRole__party                   ?subscriber .
?expectation  saTBox:ssm_expectationSetBy            ?subscriber .
?expectation  saTBox:ssm_expectationOnService        ?service .
}
```

Listing C-3: SPARQL Query for Matching Service Experience Snapshots with Expectations

```
PREFIX saTBox: <http://aesop/ServiceAnalysisTBox.owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX sid: <http://example.org/SID_Phase_VII_Models#>

SELECT ?compliance ?size ?upload ?bytes ?expectedSpeed ?actualSpeed ?duration
     WHERE
{
 #
 # Get the values we're interested in from the various instances
 #
 <--VAR_SESSION--> saTBox:ssm_fileTransferUpload ?upload .
 <--VAR_SESSION--> saTBox:ssm_fileTTranFileSize ?size .
 <--VAR_SNAPSHOT--> saTBox:ssm_bytesUpstream ?bytesUp .
 <--VAR_SNAPSHOT--> saTBox:ssm_bytesDownstream ?bytesDown .
 <--VAR_SNAPSHOT--> saTBox:ssm_sessionSnapshotDuration ?duration .
 <--VAR_EXPECTATION--> saTBox:ssm_expFileTransmallFileSize ?smallSize .
 <--VAR_EXPECTATION--> saTBox:ssm_expFileTransmallFileTranspeed ?smallSpeed .
 <--VAR_EXPECTATION--> saTBox:ssm_expFileTTranMediumFileSize ?medSize .
 <--VAR_EXPECTATION--> saTBox:ssm_expFileTTranMediumFileTranspeed ?medSpeed .
 <--VAR_EXPECTATION--> saTBox:ssm_expFileTTranLargeFileSize ?largeSize .
 <--VAR_EXPECTATION--> saTBox:ssm_expFileTTranLargeFileTranspeed ?largeSpeed .
 <--VAR_EXPECTATION--> saTBox:ssm_expFileTTranHugeFileSize ?hugeSize .
 <--VAR_EXPECTATION--> saTBox:ssm_expFileTTranHugeFileTranspeed ?hugeSpeed .
 <--VAR_EXPECTATION--> saTBox:ssm_expFileTTranEnormousFileTranspeed ?enorSpeed .

 #
 # Find the number of bytes tTranred depending on tTran direction
 #
 BIND ( if ( ?upload = true, ?bytesUp, ?bytesDown ) AS ?bytes ).

 #
 # Calculate the expectation on file transfer speed based on the file size
 #
 BIND (
  if (?size <= ?smallSize, ?smallSpeed,
   if (?size <= ?medSize, ?medSpeed,
    if (?size <= ?largeSize, ?largeSpeed,
     if (?size <= ?hugeSize, ?hugeSpeed, ?enorSpeed
     )
    )
   )
  )
 AS ?expectedSpeed ).

 #
 # Calculate the actual file tTran speed convert bytes to bits, then multiply by
     1000, duration is in milliseconds
 #
 BIND ( ( (?bytes * 8000) / ?duration ) AS ?actualSpeed ).

 #
 # Calculate percentage compliance with expectations, if met, it's 100%
 #
 BIND ( if (?actualSpeed >= ?expectedSpeed, 100, ( ?actualSpeed * 100 /
     ?expectedSpeed ) ) AS ?compliance ).
}
```

Listing C-4: SPARQL Query for Calculating File Transfer Session Compliance

```
PREFIX saTBox: <http://aesop/ServiceAnalysisTBox.owl#>
PREFIX rdfs:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX xsd:  <http://www.w3.org/2001/XMLSchema#>
PREFIX sid:  <http://example.org/SID_Phase_VII_Models#>

SELECT ?compliance ?high_quality ?quality ?expected_high_quality
     ?expected_standard_quality WHERE
{
 #
 # Get the values we're interested in from the various instances
 #
 <--VAR_SESSION-->  saTBox:ssm_videoOnDemandHighQuality   ?high_quality.
 <--VAR_SNAPSHOT--> saTBox:ssm_videoOnDemandQuality  ?quality.
 OPTIONAL {
   <--VAR_EXPECT--> saTBox:ssm_expectVODQuality ?expected_high_quality.
   <--VAR_EXPECT--> rdfs:type     saTBox:SSM_VODServiceExpectationHQ.
 }
 OPTIONAL {
   <--VAR_EXPECT--> saTBox:ssm_expectVODQuality ?expected_standard_quality.
   <--VAR_EXPECT--> rdfs:type     saTBox:SSM_VODServiceExpectationSQ.
 }

 #
 # Find the compliance level of high quality sessions with expectations
 #
 BIND (if (?expected_high_quality,
  if (?high_quality = true,
   if (?quality >= ?expected_high_quality, 100,
       ?quality * 100.0 / ?expected_high_quality
      ),
   -1),
  -1
  )
  AS ?compliance).

 #
 # Find the compliance level of standard quality sessions with expectations
 #
 BIND (if (?expected_standard_quality,
  if (?high_quality = false,
   if (?quality >= ?expected_standard_quality, 100,
       ?quality * 100.0 / ?expected_standard_quality
      ),
   -1),
  -1
 )
 AS ?compliance).
}
```

Listing C-5: SPARQL Query for Calculating Video on Demand Session Compliance

```
PREFIX saTBox: <http://aesop/ServiceAnalysisTBox.owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX sid: <http://example.org/SID_Phase_VII_Models#>

SELECT ?compliance ?load_time_mean ?load_size_mean ?exp_size_small
     ?exp_size_medium ?exp_time_small ?exp_time_medium ?exp_time_large ?exp_time
     WHERE
{
 #
 # Get the values we're interested in from the various instances
 #
 <--VAR_SNAPSHOT--> saTBox:ssm_webRequestLoadTimeMean ?load_time_mean;
                    saTBox:ssm_webBrowsingRequestSizeMean ?load_size_mean.
 <--VAR_EXPECT-->   saTBox:ssm_expectWEBSmallMeanRequestSize ?exp_size_small;
 saTBox:ssm_expectWEBMediumMeanRequestSize ?exp_size_medium;
 saTBox:ssm_expectWEBSmallMeanRequestLoadTime ?exp_time_small;
 saTBox:ssm_expectWEBMediumMeanRequestLoadTime ?exp_time_medium;
 saTBox:ssm_expectWEBLargeMeanRequestLoadTime ?exp_time_large.

 #
 # Calculate the expectation on load time based on request size
 #
 BIND (
   if (?load_size_mean <= ?exp_size_small, ?exp_time_small,
   if (?load_size_mean <= ?exp_size_medium, ?exp_time_medium, ?exp_time_large
 )
 )
 AS ?exp_time ).

 #
 # Now calculate the percentage compliance with expectations, if expectations are
     met, it's 100%
 #
 BIND ( if (?load_time_mean <= ?exp_time, 100, ( ?exp_time * 100 /
     ?load_time_mean ) ) AS ?compliance ).
}
```

Listing C-6: SPARQL Query for Calculating Web Browsing Session Compliance

C.3 SPARQL Queries for Optimization Planning and Execution

```
PREFIX rdfs:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX saTBox: <http://aesop/ServiceAnalysisTBox.owl#>
PREFIX xsd:    <http://www.w3.org/2001/XMLSchema#>

SELECT ?new_session_count ?terminated_session_count WHERE
{
 {
  SELECT (COUNT(?session) AS ?new_session_count) WHERE
  {
   ?snapshot_bucket rdfs:type        saTBox:SSM_SnapshotBucket;
        saTBox:ssm_snapshotBucketDirty   ?dirty.
   ?se_snapshot  rdfs:type        saTBox:SSM_ServiceExperienceSnapshot;
        saTBox:ssm_inSnapshotBucket    ?snapshot_bucket;
        saTBox:ssm_sessionStartTime    ?session_start.
   ?session   rdfs:type        saTBox:SSM_ServiceSession.
   FILTER (?dirty = false)
   FILTER (?session_start >= "--VAR_WINDOW_BEGINNING--"^^xsd:dateTime)
  }
 }
 {
  SELECT (COUNT(?session) AS ?terminated_session_count) WHERE
  {
   ?snapshot_bucket rdfs:type        saTBox:SSM_SnapshotBucket;
        saTBox:ssm_snapshotBucketDirty   ?dirty.
   ?se_snapshot  rdfs:type        saTBox:SSM_ServiceExperienceSnapshot;
        saTBox:ssm_inSnapshotBucket    ?snapshot_bucket;
        saTBox:ssm_sessionTerminate    ?session_terminate.
   ?session   rdfs:type        saTBox:SSM_ServiceSession.
   FILTER (?dirty = false)
   FILTER (?session_terminate = true)
  }
 }
 FILTER (?new_session_count > 0 || ?terminated_session_count > 0)
}
```

Listing C-7: SPARQL Query for New or Deleted Session Pre-Check

```
PREFIX rdfs:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX saTBox: <http://aesop/ServiceAnalysisTBox.owl#>
PREFIX xsd:    <http://www.w3.org/2001/XMLSchema#>

SELECT ?session ?session_id ?session_priority ?bucket_beginning ?compliance
     ?serv_name ?hostaddr ?userid ?bytes_down ?bytes_up ?thrdl ?thrul WHERE
{
 {
  SELECT DISTINCT ?session ?session_id ?session_priority ?bucket_beginning
     ?compliance ?serv_name ?hostaddr ?userid ?bytes_down ?bytes_up ?thrdl ?thrul
     WHERE
  {
   ?snapshot_bucket rdfs:type saTBox:SSM_SnapshotBucket;
        saTBox:ssm_beginning ?bucket_beginning;
        saTBox:ssm_snapshotBucketDirty ?dirty.
   ?snapshot saTBox:ssm_inSnapshotBucket ?snapshot_bucket;
        saTBox:ssm_experienceCompliance ?compliance;
        saTBox:ssm_bytesDownstream ?bytes_down;
        saTBox:ssm_bytesUpstream  ?bytes_up;
          saTBox:ssm_experienceSnapshotOfSession   ?session.
   ?session rdfs:type saTBox:SSM_ServiceSession;
        saTBox:ssm_runsInstanceOfService ?serv;
        saTBox:ssm_isExecutedByUser ?user;
        saTBox:ssm_runsOnConnection ?conn;
        saTBox:ssm_sessionId ?session_id;
        saTBox:ssm_priorityOfServiceSession    ?priority.
   ?priority   saTBox:ssm_servicePriority             ?session_priority;
          saTBox:ssm_servicePriorityDownloadThrottleLimit ?thrdl;
          saTBox:ssm_servicePriorityUploadThrottleLimit ?thrul.
   ?serv saTBox:ssm_serviceName ?serv_name.
   ?user saTBox:ssm_personId?userid.
   ?conn saTBox:ssm_connectionAddress ?hostaddr.
   OPTIONAL {
    ?session saTBox:ssm_sessionTerminated ?term.
   }
   OPTIONAL {
    ?session saTBox:ssm_throttlingActive ?throt.
   }
   OPTIONAL {
    ?session saTBox:ssm_throttlingActive ?throt_active.
   }
   BIND (if (NOT EXISTS {?session saTBox:ssm_throttlingActive ?thr}, false,
     ?throt_active) AS ?throttled).
   BIND (if (NOT EXISTS {?session saTBox:ssm_sessionTerminated ?trm},  false,
     ?sess_term) AS ?terminated).
   FILTER (?throttled = false)
   FILTER (?terminated = false)
   FILTER (?dirty = false)
  }
 }
 {
  SELECT (MIN(?session_priority) as ?min_prio)
  {
   ?session saTBox:ssm_priorityOfServiceSession ?priority.
   ?priority saTBox:ssm_servicePriority ?session_priority.
   OPTIONAL {
    ?session saTBox:ssm_sessionTerminated ?term.
   }
   BIND (if (NOT EXISTS {?session saTBox:ssm_sessionTerminated ?trm}, false,
     ?term) AS ?terminated).
   OPTIONAL {
    ?session saTBox:ssm_throttlingActive  ?throt.
   }
   BIND (if (NOT EXISTS {?session saTBox:ssm_throttlingActive ?thr}, false,
     ?throt) AS ?throttled).
   FILTER (?throttled = false)
   FILTER (?terminated = false)
  }
 }
 {
  SELECT (MAX(?session_priority) as ?max_prio)
  {
   ?session   saTBox:ssm_priorityOfServiceSession ?priority.
```

```
   ?priority   saTBox:ssm_servicePriority ?session_priority.
   OPTIONAL {
    ?session saTBox:ssm_sessionTerminated ?term.
   }
   BIND (if (NOT EXISTS {?session saTBox:ssm_sessionTerminated ?trm}, false,
     ?term) AS ?terminated).
   OPTIONAL {
    ?session saTBox:ssm_throttlingActive  ?throt.
   }
   BIND (if (NOT EXISTS {?session saTBox:ssm_throttlingActive ?thr}, false,
     ?throt) AS ?throttled).
   FILTER (?throttled = false)
   FILTER (?terminated = false)
  }
 }
 FILTER (?session_priority > ?min_prio)
 FILTER (?session_priority = ?max_prio)
}
ORDER BY ?id
```

Listing C-8: SPARQL Query to Select an End-User Session for Throttling Application

Appendix D
Aesop Implementation Details

D.1 The Knowledge Bus Interface

Table D-1: Methods on the Knowledge Bus Interface

Method	Parameters	Description
getStaticReasonedModel	*None*	Get the metadata and static knowledge as a reasoned model.
getEmptyReasonedModel	*None*	Get the metadata as a reasoned model.
getBaseUnreasonedModel	*None*	Get the metadata and all global knowledge as a plain model.
getBaseReasonedModel	*None*	Get the metadata and all global knowledge as a reasoned model.
getFullReasonedModel	*None*	Get all knowledge in the Knowledge Base as a reasoned model
getRangedReasonedModel	*fromDate* *toDate*	Get all global knowledge and snapshot knowledge between the given dates as a reasoned model
getSinceReasonedModel	*sinceDate*	Get all global knowledge and snapshot knowledge since the given date as a reasoned model
updateDynamicModel	*instances* *URI*	Update the dynamic global individual identified by the URI with the knowledge instances provided.
getDynamicIndiviudal	*URI*	Get the dynamic global individual identified by the URI.
getSnapshotBuckets	*None*	Get all snapshot buckets.
getSnapshotBucket	*date*	Get the snapshot bucket for the given date.
removeSnapshotBucket	*date*	Remove the snapshot bucket for the given date.
SnapshotBucket	*model* *date*	Create a snapshot bucket for the given date that references the given global dynamic model
updateSnapshot	*URI* *proprty* *value*	Update the given property with the given value on the snapshot with the given URI in a snapshot bucket.

D.2 Static Global Knowledge Configuration in Protégé

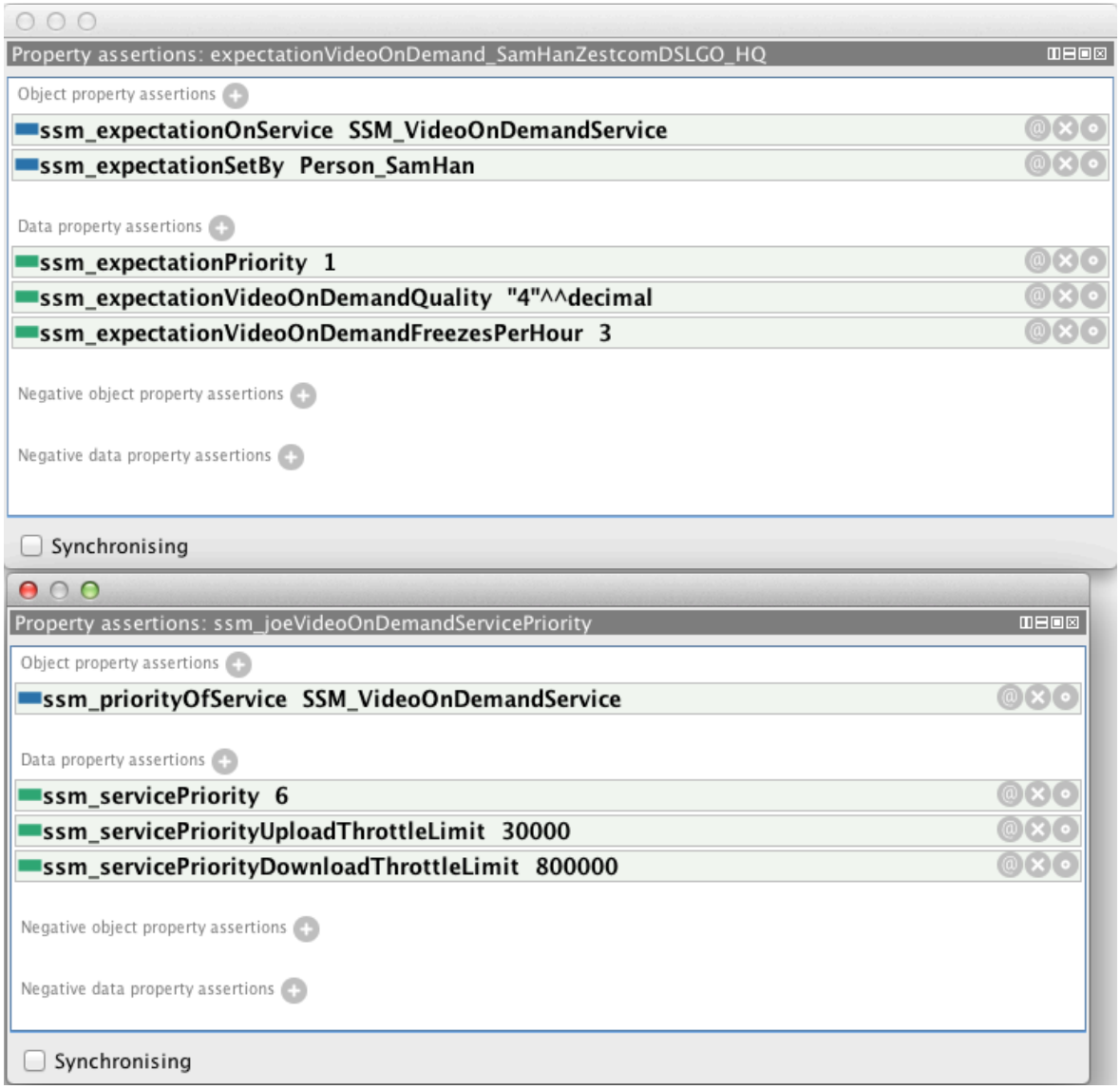


Figure D-1: Service Expectation and Service Priority Configuration

D.3 SAWSDL Annotations

```
<xs:complexType name="location"
 sawsdl:modelReference="http://aesop/ServiceAnalysisABoxDynamic.owl#"
 sawsdl:liftingSchemaMapping="http://aesop/TerminalReports/Location.xslt">
 <xs:sequence>
  <xs:element name="Latitude" type="xs:float" maxOccurs="1"
   minOccurs="1"></xs:element>
  <xs:element name="Longitude" type="xs:float" maxOccurs="1"
   minOccurs="1"></xs:element>
  <xs:element name="Altitude" type="xs:float" maxOccurs="1"
   minOccurs="1"></xs:element>
 </xs:sequence>
</xs:complexType>

<xs:complexType name="terminal"
 sawsdl:modelReference="http://aesop/ServiceAnalysisABoxDynamic.owl#"
 sawsdl:liftingSchemaMapping="http://aesop/TerminalReports/Terminal.xslt">
 <xs:sequence>
  <xs:element name="ID" type="xs:string" maxOccurs="1"
   minOccurs="1">
  </xs:element>
  <xs:element name="Name" type="xs:string" maxOccurs="1"
   minOccurs="0">
  </xs:element>
  <xs:element name="Type" type="xs:string" maxOccurs="1"
   minOccurs="0">
  </xs:element>
  <xs:element name="Manufacturer" type="xs:string"
   maxOccurs="1" minOccurs="0">
  </xs:element>
  <xs:element name="Model" type="xs:string" maxOccurs="1"
   minOccurs="0">
  </xs:element>
  <xs:element name="SerialNumber" type="xs:string"
   maxOccurs="1" minOccurs="0">
  </xs:element>
  <xs:element name="Location" type="location" maxOccurs="1"
   minOccurs="1">
  </xs:element>
  <xs:element name="Connection" type="connection"
   maxOccurs="unbounded" minOccurs="1">
  </xs:element>
  <xs:element name="Session" type="session"
   maxOccurs="unbounded" minOccurs="0">
  </xs:element>
 </xs:sequence>
</xs:complexType>

<xs:complexType name="user"
 sawsdl:modelReference="http://aesop/ServiceAnalysisABoxDynamic.owl#"
 sawsdl:liftingSchemaMapping="http://aesop/TerminalReports/User.xslt">
 <xs:sequence>
  <xs:element name="ID" type="xs:string" maxOccurs="1"
   minOccurs="1" />
  <xs:element name="UserID" type="xs:string" maxOccurs="1"
     minOccurs="0"></xs:element>
  <xs:element name="Surname" type="xs:string" maxOccurs="1"
   minOccurs="0" />
  <xs:element name="FirstName" type="xs:string" maxOccurs="1"
   minOccurs="0">
  </xs:element>
 </xs:sequence>
</xs:complexType>
```

Listing D-1: SAWSDL Annotations on a Terminal Report XML Schema

D.4 XSL Transformations for Lifting Terminal Report Elements

Table D-2: Transformations Used to Lift Terminal Report Elements

XML Element	XSL Transformation File
TerminalReport	TerminalReportElement.xslt
Terminal	TerminalElement.xslt
Location	LocationElement.xslt
User	UserElement.xslt
Session	SessionElement.xslt
Metrics	MetricsElementTerminal.xslt MetricsElementSession.xslt

```
<?xml version="1.0" encoding="UTF-8"?>

<xsl:transform version="2.0"
 xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
 xmlns:str="http://aesop/TerminalReport">

 <xsl:strip-space elements="*" />
 <xsl:output method="text" version="1.0" encoding="iso-8859-1" indent="yes" />

 <xsl:template match="/">
 <xsl:if test="str:Terminal/str:Name">
@prefix :       &lt;http://aesop/ServiceAnalysisABoxDynamic.owl#&gt; .
@prefix saTBox: &lt;http://aesop/ServiceAnalysisTBox.owl#&gt; .
@prefix owl:    &lt;http://www.w3.org/2002/07/owl#&gt; .

:SSM_Terminal_<xsl:value-of select="str:Terminal/str:ID"/> a owl:NamedIndividual,
     saTBox:SSM_Terminal ;
 saTBox:ssm_terminalId   "<xsl:value-of select="str:Terminal/str:ID"/>" ;
 saTBox:ssm_terminalName         "<xsl:value-of select="str:Terminal/str:Name"/>"
     ;
 saTBox:ssm_terminalType         "<xsl:value-of select="str:Terminal/str:Type"/>"
     ;
 saTBox:ssm_terminalManufacturer "<xsl:value-of
     select="str:Terminal/str:Manufacturer"/>" ;
 saTBox:ssm_terminalModel  "<xsl:value-of select="str:Terminal/str:Model"/>" ;
 saTBox:ssm_terminalSerialNumber "<xsl:value-of
     select="str:Terminal/str:SerialNumber"/>" .
 </xsl:if>
 </xsl:template>
</xsl:transform>
```

Listing D-2: XSL Transformation for Lifting a *Terminal* Element in a Terminal Report

```
<?xml version="1.0" encoding="UTF-8"?>

<xsl:transform version="2.0"
 xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
 xmlns:str="http://aesop/TerminalReport">

 <xsl:strip-space elements="*" />
 <xsl:output method="text" version="1.0" encoding="iso-8859-1" indent="yes" />

 <xsl:template match="/">
@prefix :        &lt;http://aesop/ServiceAnalysisABoxDynamic.owl#&gt; .
@prefix saTBox:  &lt;http://aesop/ServiceAnalysisTBox.owl#&gt; .
@prefix saABoxS: &lt;http://aesop/ServiceAnalysisABoxStatic.owl#&gt; .
@prefix foaf:    &lt;http://xmlns.com/foaf/0.1#&gt; .
@prefix owl:     &lt;http://www.w3.org/2002/07/owl#&gt; .

    <xsl:for-each select="str:User">
  <xsl:if test="str:UserID">
saABoxS:SSM_User_<xsl:value-of select="str:ID"/>
 foaf:geekCode      "<xsl:value-of select="str:UserID"/>" ;
 foaf:familyName      "<xsl:value-of select="str:Surname"/>" ;
 foaf:firstNameName     "<xsl:value-of select="str:FirstName"/>" .
  </xsl:if>
    </xsl:for-each>
 </xsl:template>
</xsl:transform>
```

Listing D-3: XSL Transformation for Lifting a *User* Element in a Terminal Report

Appendix E
Experimental Equipment

Table E-1: Experimental Equipment

Name	Machine Model	Host Type	Processor	Mem	Disk Space	Interface Actual (Capability)
Laptop 1	HP Compaq nw8240 Laptop	Client	Intel Pentium M @ 2.0GHz, 32-bit 1 core	1GB	60GB	HAN: 100 (1000) Mbit/s Ethernet
Laptop 2	HP Compaq tc4200 Laptop	Client	Intel Pentium M @ 1.86GHz, 32-bit 1 core	1.5GB	60GB	HAN: 100 (1000) Mbit/s Ethernet
Laptop 3	HP Compaq tc4200 Laptop	Client	Intel Pentium M @ 1.86GHz, 32-bit 1 core	1.5GB	60GB	HAN: 100 (1000) Mbit/s Ethernet
Laptop 4	HP Compaq tc4200 Laptop	Client	Intel Pentium M @ 1.86GHz, 32-bit 1 core	1.5GB	60GB	HAN: 100 (1000) Mbit/s Ethernet
HAN Switch	3Com 3CFSU08 Switch 8	HAN Ethernet Switch	N/A	N/A	N/A	HAN: 8x 100 (100) Mbit/s Ethernet
saor	HP Z600 Workstation	Home Gateway	Intel Xeon E5620 @ 2.4GHz, 64-bit 4 cores	24GB	250GB	Outbound: 100 (1000) Mbit/s Ethernet HAN: 100 (100) Mbit/s Ethernet
scagaire	HP xw6000 Workstation	Traffic Shaper	Intel Xeon @ 2.8GHz, 32-bit, 1 core	1GB	36GB	WAN: 1000 (1000) Mbit/s Ethernet HGW side: 100 (100) Mbit/s Ethernet

Name	Machine Model	Host Type	Processor	Mem	Disk Space	Interface Actual (Capability)
WAN Switch	D-Link DGS-1248T	WAN Ethernet Switch	N/A	N/A	N/A	WAN: 48x 1000 (1000) Mbit/s Ethernet
cara	HP xw6200 Workstation	Server for Video on Demand, FTP, Web	2 x Intel Xeon @ 2.8GHz, 32-bit, 1 core	2GB	250GB	WAN: 1000 (1000) Mbit/s Ethernet

All hosts except Ethernet switches ran Fedora 17 Linux.

Appendix F
Test Bed Photographs

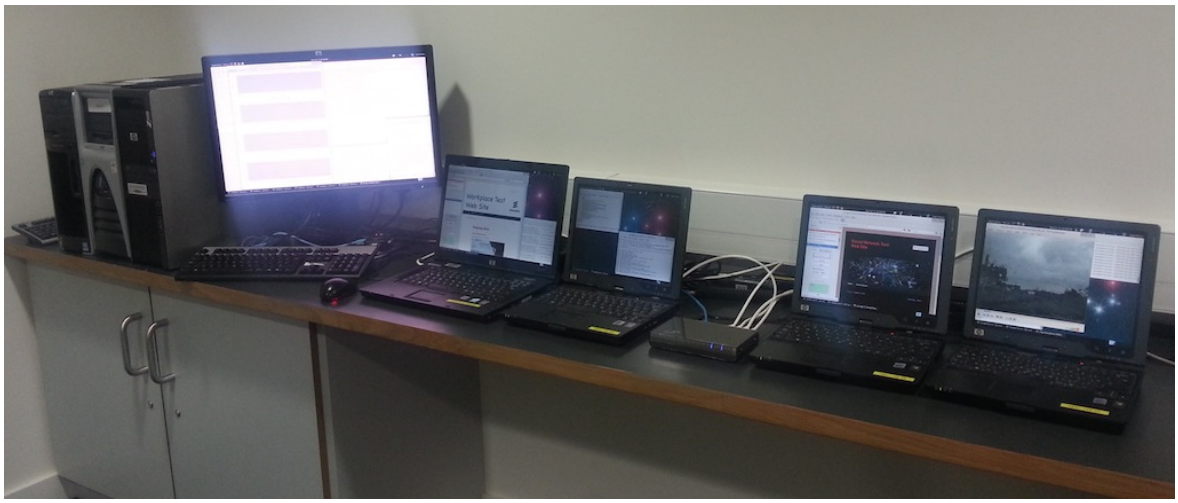


Figure F-1: The Experimental Test Bed



Figure F-2: The Server, Traffic Shaper, and Home Gateway

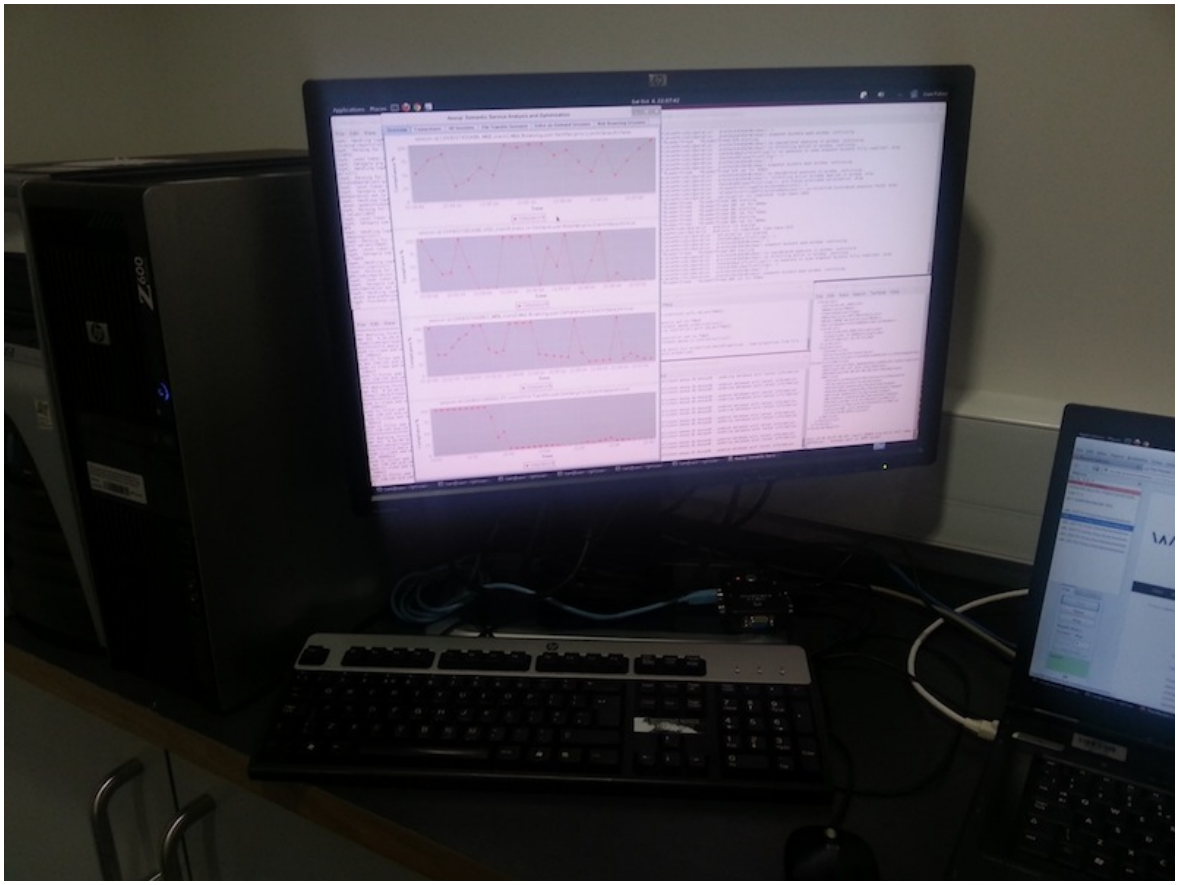


Figure F-3: Home Gateway Monitor

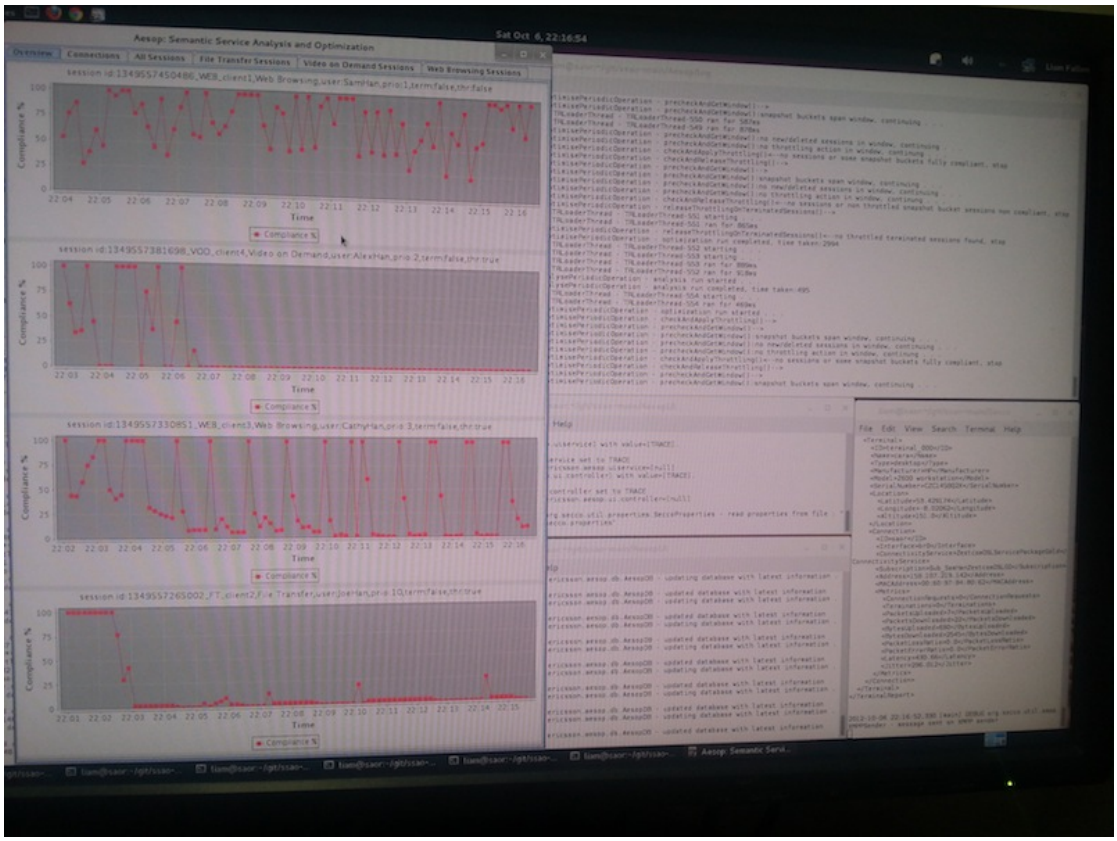


Figure F-4: Close up of Home Gateway Monitor

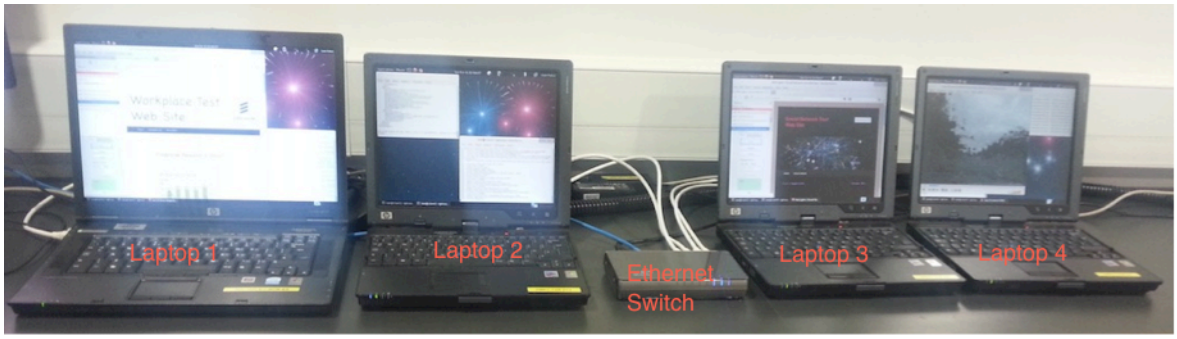


Figure F-5: Laptops and HAN Ethernet Switch

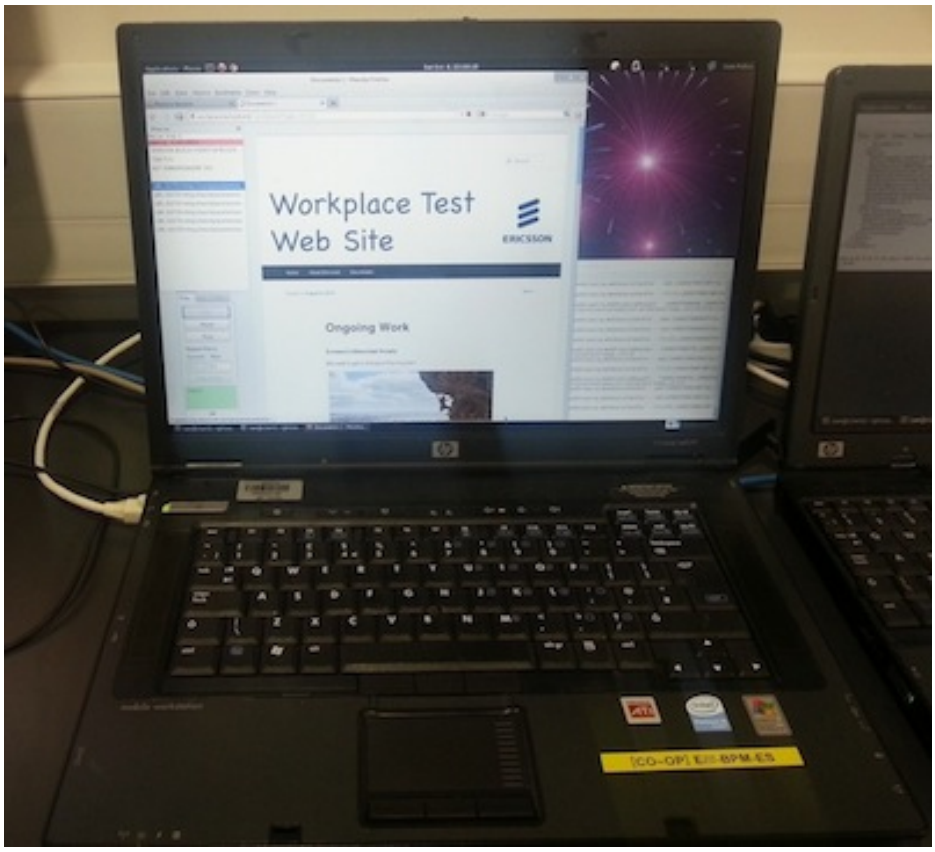


Figure F-6: Laptop 1 running a Workplace Web Session

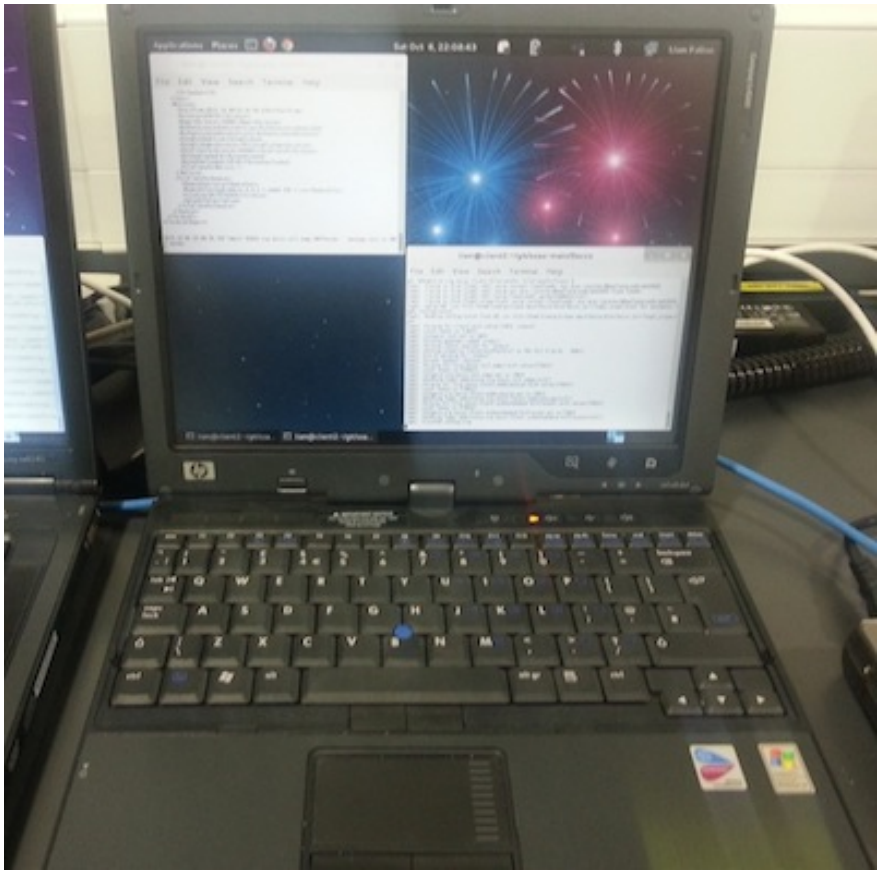


Figure F-7: Laptop 2 running a File Transfer Session

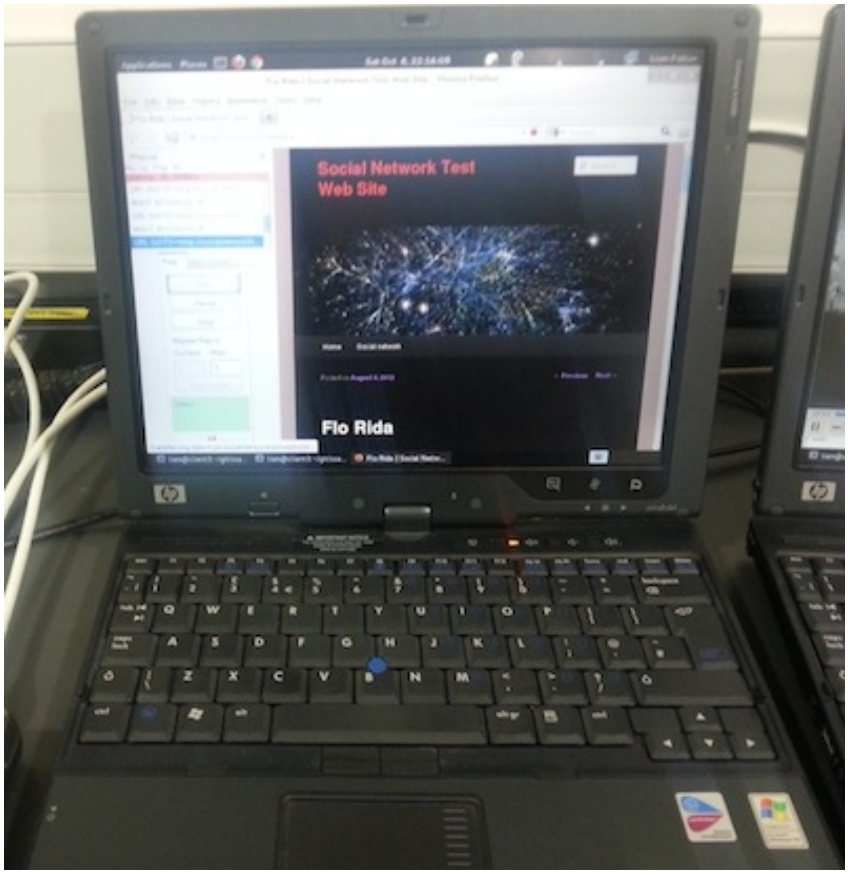


Figure F-8: Laptop 3 running a Social Networking Web Session



Figure F-9: Laptop 4 running a Video on Demand Session

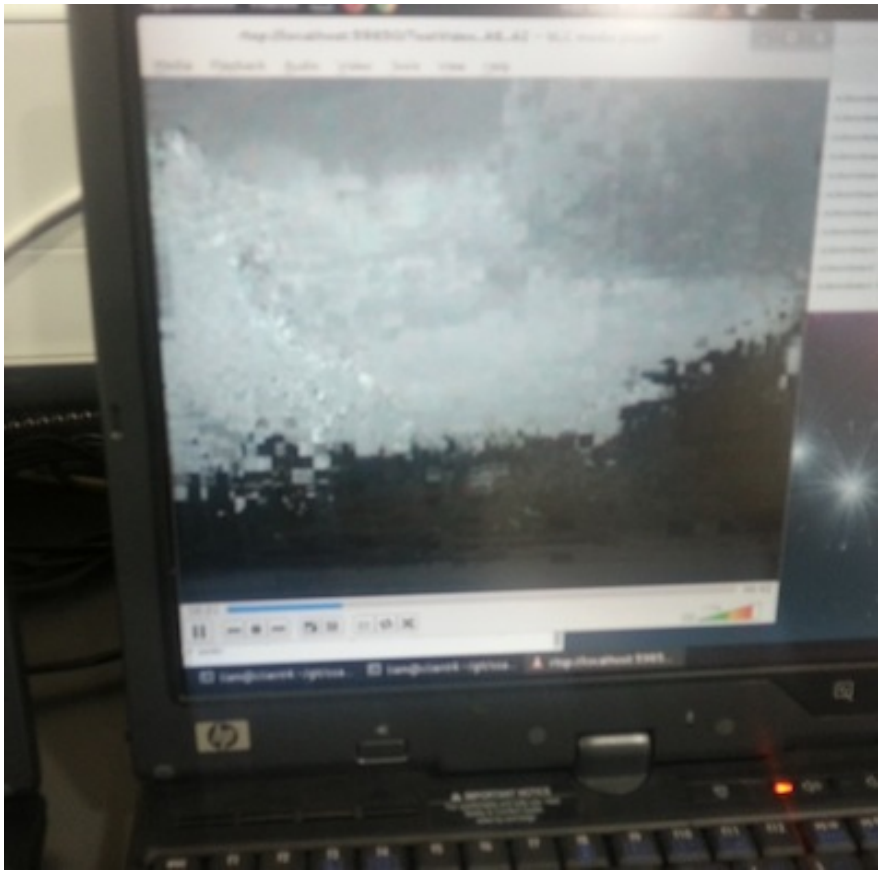


Figure F-10: Close Up of Degraded Video Session running on Laptop 4

Appendix G
Third Party Software

Table G-1: Third Party Software Used

Name	Version	Used	Details
ant	1.8.2	Development and Experiments	A software build and execution tool mainly for Java applications. Used to build and execute Aesop and SECCO, and to run simulation experiments. http://ant.apache.org/
Apache Commons Math	3.0	SECCO	A Java maths library used in SECCO to calculate mean and standard deviation metrics on end user sessions. http://commons.apache.org/math
Apache Commons Net	3.0.1	SECCO	A communication library for Java. SECCO used FTP support in this library to implement the File Transfer client. http://commons.apache.org/net
Apache Jena	2.7.0 Incubating	Aesop	A Java framework for building semantic applications. Used to hold semantic models in OWL 2 and to manipulate individuals in models. The Jena ARQ query engine was used for SPARQL queries. http://jena.apache.org
Apache Tomcat	6.0.33	Simulation	A Java servlet execution environment. Apache Tomcat was used to run the Sesame servlet in the simulation environment. http://tomcat.apache.org
Firefox	15.0.1	SECCO	A web browser. Used under the control of SECCO to run end user web browsing sessions. http://www.mozilla.org/
git	1.7.12.3	Development and Experiments	A software repository system for storing and versioning software and other information. Used during implementation for software and during experimentation to hold configuration information, scripts, raw results, analysed results, and plots. http://git-scm.com

Name	Version	Used	Details
HttpFox	0.8.11	SECCO	A plugin for Firefox that records metrics on web requests. It was modified in SECCO to write metrics to a log file https://addons.mozilla.org/en-us/firefox/addon/httpfox/
iMacros	7.6.0.2	SECCO	A plugin for Firefox that allows a web browsing session to be scripted. It was used in SECCO to automate end user web sessions. https://addons.mozilla.org/en-US/firefox/addon/imacros-for-firefox
Java	1.6 Java 6	Aesop and SECCO	A programming language. Used for all software development. java.com
Jaxen	1.2	Aesop	A Java XPath library used in Aesop for semantic lifting. http://jaxen.codehaus.org
JDOM	1.1.1	Aesop and SECCO	A Java library for XML technologies. JDOM was used to validate, read, build, and manipulate XML documents. JDOM support for XML schemas, XPath, and XSL transformations was used. http://www.jdom.org
JFreeChart	1.0.14	Aesop Client	A Java plotting library used in the Aesop client to plot compliance. http://www.jfree.org/jfreechart
Openfire	3.7.1	Experiments	An XMPP server. Openfire ran on the Home Gateway machine. SECCO sent terminal reports to it and Aesop received terminal reports from it. http://www.igniterealtime.org/
Pellet	2.3.0	WAN Ethernet Switch	A Java OWL 2 reasoner library for Java. Aesop uses Pellet for reasoning and for execution of SWRL rules. http://clarkparsia.com/pellet
PostgreSQL	9.1	Aesop Client and Experiments	A RDBMS used by the Aesop client and during experimentation to log results. SQL queries towards PostgreSQL were used to analyse results. http://www.postgresql.org
Sesame	2.6.2	Aesop	A Java semantic store implementation. Aesop used sesame to store semantic knowledge persistently. http://www.openrdf.org

Name	Version	Used	Details
Smack	3.2.2	Aesop and SECCO	An XMPP library for Java. Aesop and SECCO used the Smack API to send and receive terminal reports over XMPP. http://www.igniterealtime.org/
SNMP4J	2.0.3	SECCO	A Java API for running SNMP requests and receiving SNMP traps. Used in SECCO to collect interface metrics. http://www.snmp4j.org
VLC	2.0.3	SECCO and Experiments	A video player and streaming server. Used in SECCO to run end user video on demand sessions and as a video on demand server in experiments. http://www.videolan.org/vlc/index.html
Xerces	2.6.0	Aesop and SECCO	A Java library for parsing XML documents. Used in Aesop and SECCO for parsing terminal reports. http://xerces.apache.org
XSOM	2011-01-01 snapshot	Aesop	A Java library for handling XML schemas. Aesop uses XSOM to parse XML schema documents for SAWSDL annotations http://xsom.java.net

Appendix H
Normal Quantile Plots for Experimental Metrics

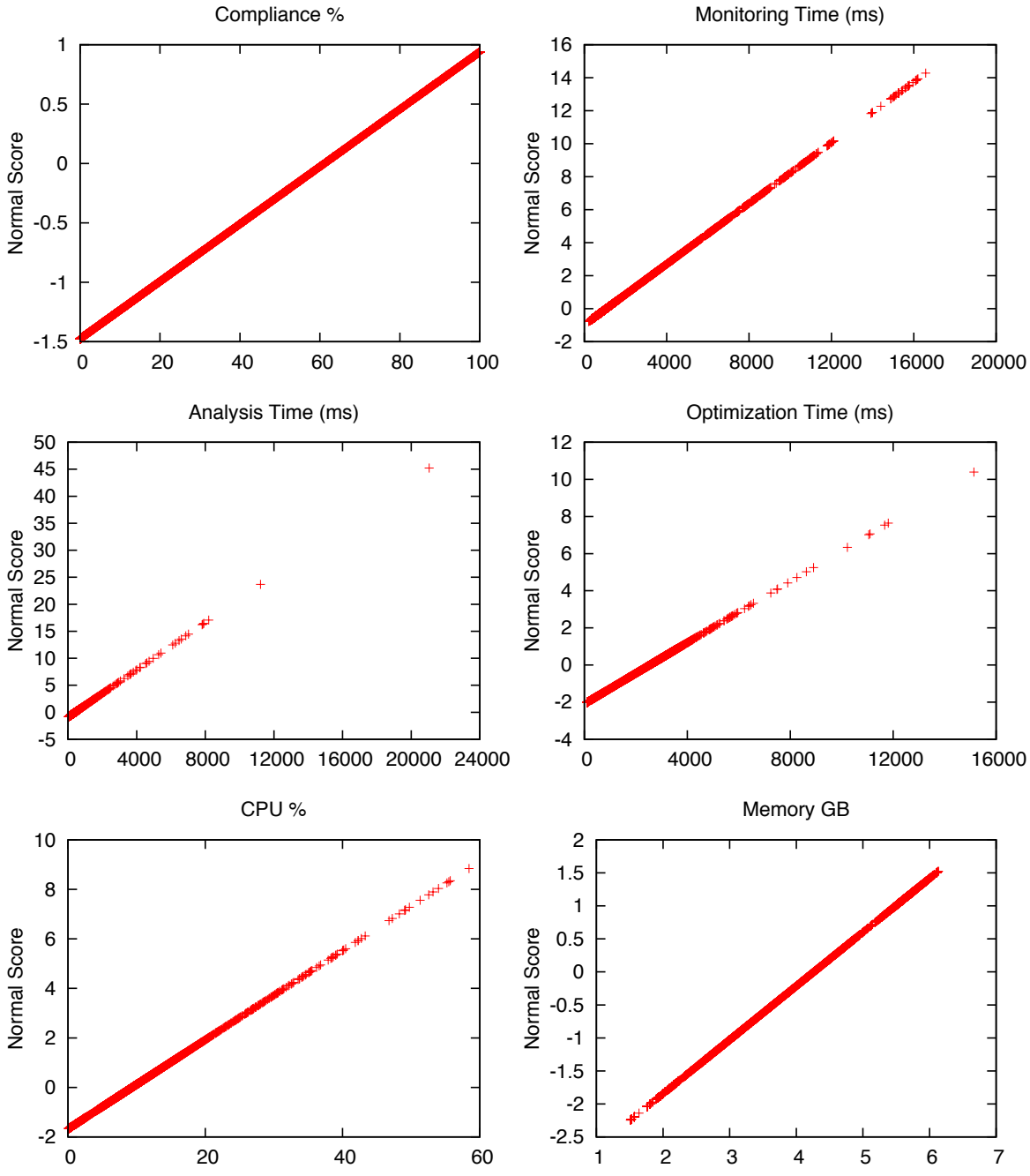


Figure H-1: Normal Quantile Plots for Experimental Metrics

Appendix I
Experimental Scenario Specifications

I.1 Scenario 1: Degradation in Video on Demand Session

This scenario specification implements the scenario described in Section 6.2.3.1 and was run in the simulated environment described in Section 6.2.4. A number of simultaneous Video on Demand sessions are running, and the quality of each session deteriorates as the sessions progress until each session becomes unwatchable.

For each run, the simulator ran through a series of tests, where the number of simultaneous Video on Demand sessions was varied. The simulator was seeded as laid out in Table I-1 and generated terminal reports for the required terminal report load to disk. Aesop read those terminal reports from disk and processed them.

Table I-1: Scenario 1 Experimental Specification

Test	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Terminals	6	10	20	30	40	50	60	100	200	300	150	175	125	80	275	250	225

I.2 Scenario 2: Congestion on Broadband Access Network Connection

This scenario specification implements the scenario described informally in Section 6.2.3.2 and was run on the experimental test bed described in Section 6.2.5. Four service sessions are running simultaneously in a HAN, causing congestion on the broadband connection. Aesop monitors the services and shows the degree of compliance of each service with its expectations over the period of experimental runs.

All experimental runs begin with observation of HGW CPU and Memory usage. The Aesop server is started three minutes after CPU/Memory usage observation starts. The first client session starts a further three minutes into experimental runs, to allow transient effects due to its start to settle down.

The client sessions shown in Table I-2 are started at 1-minute intervals in order to provide time for any effect a starting client session has on client sessions that are already running to be observed. Client sessions are terminated at 2-minute intervals,

allowing time for the effect of a client session termination on running sessions to be observed. The Aesop server is stopped three minutes after the last client session terminates. HGW CPU and memory usage is monitored for a further 3 minutes. The total run time is 60 minutes for each experimental run.

Table I-2: Scenario 2 Experimental Specification

Time	Client	User	Type	Session Parameters
t	HGW		CPU/Memory Monitor Start	Start monitoring CPU/Memory usage
t+3m	HGW		Aesop Start	Start Aesop server, UI and DB Clients
t+6m	HAN		SECCO Start	Start SECCO server on all HAN hosts
t+9m	client2	Joe	File Transfer Start	Debian 6.0.2 DVD, 4.7GB
t+10m	client3	Cathy	Social Network Web Start	Sequence of web sites repeated: http://socialnetworktestwebsite/socialnetwork/?page_id=2 http://socialnetworktestwebsite/socialnetwork/?p=34 http://socialnetworktestwebsite/socialnetwork/?p=54 http://socialnetworktestwebsite/socialnetwork/?p=57 http://socialnetworktestwebsite/socialnetwork/?p=61 http://socialnetworktestwebsite/socialnetwork/?p=67 http://socialnetworktestwebsite/socialnetwork/?p=71 http://socialnetworktestwebsite/socialnetwork/?p=74 http://socialnetworktestwebsite/socialnetwork/?p=92 http://socialnetworktestwebsite/socialnetwork/?p=108 http://socialnetworktestwebsite/socialnetwork/?p=116 http://socialnetworktestwebsite/socialnetwork/?p=124
t+11m	client4	Alex	VoD, Standard Quality Start	Video streamed from VOD server: rtsp://cara/TestVideo_46_42
t+12m	client1	Sam	Workplace Web Start	Sequence of web sites revisited http://workplacetestwebsite/workplace/?page_id=110 http://workplacetestwebsite/workplace/?p=151 http://workplacetestwebsite/workplace/?p=143 http://workplacetestwebsite/workplace/?p=130 http://workplacetestwebsite/workplace/?p=139 http://workplacetestwebsite/workplace/?p=9
t+45m	client2	Joe	File Transfer Stop	Stop File Transfer session
t+47m	client3	Cathy	Social Network Web Stop	Stop Social Network Web Browsing Session
t+49m	client4	Alex	VoD, Standard Quality Stop	Stop Video on Demand Session
t+51m	client1	Sam	Workplace Web Stop	Stop Web Browsing Session
t+54m	HAN		SECCO Stop	Stop SECCO server on all HAN hosts
t+57m	HGW		Aesop Stop	Stop Aesop server, UI and DB Clients
t+60m	HGW		CPU/Memory Monitor Stop	Stop monitoring CPU/Memory usage

I.3 Scenario 3: Transient Impairment on Broadband Access Network Connection

The scenario for the experiment is described informally in Section 6.2.3.3 and was run on the experimental test bed described in Section 6.2.5. Three service sessions are running simultaneously in a HAN. Initially, all services run within expectations. Over the period of the experiment network delay fluctuates, causing service degradation and improvement. The Aesop system monitors the services and shows the degree of compliance of each service with its expectations over the period of experimental runs as network delay fluctuates.

A HAN is connected to the broadband access network as shown in Figure 6-7. The broadband access connection characteristics are set for "Normal Operation" as described in Table 6-2 using traffic controls on the Traffic Shaper node.

Four service sessions are set up in the sequence and with the service parameters are shown in Table I-3, starting at time t. An interval of 1 minute is allocated between activities in order to give Aesop a chance to react to and calculate the service delivery parameters resulting from each experimental stimulus.

The progressive delay impairment described in Table 6-3 is introduced using traffic controls 12 minutes into each run.

Table I-3: Scenario 3 Experimental Specification

Time	Client	User	Type	Session Parameters
t	HGW		CPU/Memory Monitor Start	Start monitoring CPU/Memory usage
t+2m	HGW		Aesop Start	Start Aesop server
t+4m	HAN		SECCO Start	Start SECCO server on all HAN hosts
t+6m	client2	Sam	VoD Children, High Quality Start	Children's Movie streamed from VOD server: rtsp://cara:5554/BigBuckBunny480px7
t+7m	client3	Cathy	VoD Movie, High Quality Start	Movie streamed from VOD server: rtsp://cara:5554/Insecurity

Time	Client	User	Type	Session Parameters
t+8m	client4	Joe	Social Network Web Start	Sequence of web sites repeated: http://socialnetworktestwebsite/socialnetwork/?page_id=2 http://socialnetworktestwebsite/socialnetwork/?p=34 http://socialnetworktestwebsite/socialnetwork/?p=54 http://socialnetworktestwebsite/socialnetwork/?p=57 http://socialnetworktestwebsite/socialnetwork/?p=61 http://socialnetworktestwebsite/socialnetwork/?p=67 http://socialnetworktestwebsite/socialnetwork/?p=71 http://socialnetworktestwebsite/socialnetwork/?p=74 http://socialnetworktestwebsite/socialnetwork/?p=92 http://socialnetworktestwebsite/socialnetwork/?p=108 http://socialnetworktestwebsite/socialnetwork/?p=116 http://socialnetworktestwebsite/socialnetwork/?p=124
t+9m	client1	Alex	Workplace Web Start	Sequence of web sites revisited http://workplacetestwebsite/workplace/?page_id=110 http://workplacetestwebsite/workplace/?p=151 http://workplacetestwebsite/workplace/?p=143 http://workplacetestwebsite/workplace/?p=130 http://workplacetestwebsite/workplace/?p=139 http://workplacetestwebsite/workplace/?p=9
t+12m	Shaper		Delay Impairment Start	A progressive delay impairment of 30m duration starts, see Table 6-3.
t+42m	Shaper		Delay Impairment Over	
t+48m	client2	Joe	VoD Children, High Quality Stop	Stop File Transfer session
t+50m	client3	Cathy	VoD Movie, High Quality Stop	Stop Social Network Web Browsing Session
t+52m	client4	Alex	Social Network Web Stop	Stop Video on Demand Session
t+54m	client1	Sam	Workplace Web Stop	Stop Web Browsing Session
t+56m	HAN		SECCO Stop	Stop SECCO server on all HAN hosts
t+58m	HGW		Aesop Stop	Stop Aesop server
t+60m	HGW		CPU/Memory Monitor Stop	Stop monitoring CPU/Memory usage

Appendix J
Experimental Run Logs

J.1 Experiment 1

Table J-1: Experiment 1 Scenario 1 Run Log

Session	Runs in January 2012									
	4th	4th to 5th	5th	5th	5th	5th to 6th	6th	6th	6th	6th to 7th
	1	2	3	4	5	6	7	8	9	10
1	15:04	20:41	02:04	10:11	15:40	21:08	02:39	08:09	13:38	19:08
2	15:17	20:54	02:17	10:24	15:53	21:21	02:52	08:22	13:51	19:20
3	15:30	21:08	02:30	10:37	16:07	21:35	03:05	08:35	14:05	19:34
4	15:44	21:22	02:44	10:51	16:20	21:48	03:19	08:49	14:18	19:47
5	15:58	21:36	02:58	11:05	16:34	22:02	03:33	09:03	14:32	20:01
6	16:12	21:50	03:13	11:19	16:48	22:16	03:47	09:17	14:46	20:15
7	16:27	22:05	03:27	11:33	17:02	22:31	04:01	09:31	15:00	20:30
8	16:41	22:20	03:42	11:48	17:17	22:45	04:16	09:45	15:15	20:44
9	16:57	22:35	03:57	12:04	17:33	23:01	04:32	10:02	15:31	21:00
10	17:20	22:59	04:21	12:27	17:57	23:25	04:57	10:25	15:55	21:25
11	17:53	23:32	04:55	13:03	18:32	00:00	05:32	11:01	16:30	22:00
12	18:11	23:50	05:12	13:21	18:51	00:18	05:50	11:19	16:49	22:18
13	18:30	00:09	05:31	13:41	19:10	00:38	06:09	11:40	17:08	22:38
14	18:46	00:26	05:48	13:58	19:27	00:55	06:27	11:57	17:25	22:55
15	19:01	00:41	06:03	14:14	19:43	01:10	06:42	12:12	17:41	23:10
16	19:32	01:12	06:34	14:46	20:15	01:43	07:14	12:44	18:13	23:42
17	19:59	01:39	07:01	15:14	20:43	02:12	07:43	13:13	18:41	00:11

J.2 Experiment 2

Table J-2: Experiment 2 Scenario 2 Run Log

Time	Client	Type	Runs in October 2012					
			2nd to 3rd	3rd	3rd	3rd	3rd	3rd
			1	2	3	4	5	6
t	HGW	CPU/Memory Monitor Start	23:17	01:54	08:15	09:35	20:50	22:13
t+3m	HGW	Aesop Start	23:20	01:57	08:18	09:38	20:53	22:16
t+6m	HAN	SECCO Start	23:23	02:00	08:21	09:41	20:56	22:19
t+9m	client2	File Transfer Start	23:26	02:03	08:24	09:44	20:59	22:22
t+10m	client3	Social Network Web Start	23:27	02:04	08:25	09:45	21:00	22:23
t+11m	client4	VoD, Standard Quality Start	23:28	02:05	08:26	09:46	21:01	22:24
t+12m	client1	Workplace Web Start	23:29	02:06	08:27	09:47	21:02	22:25
t+45m	client2	File Transfer Stop	00:02	02:39	09:00	10:20	21:35	22:58
t+47m	client3	Social Network Web Stop	00:04	02:41	09:02	10:22	21:37	23:00
t+49m	client4	VoD, Standard Quality Stop	00:06	02:43	09:04	10:24	21:39	23:02
t+51m	client1	Workplace Web Stop	00:09	02:45	09:06	10:26	21:41	23:04
t+54m	HAN	SECCO Stop	00:12	02:48	09:09	10:29	21:44	23:07
t+57m	HGW	Aesop Stop	00:15	02:51	09:12	10:32	21:47	23:10
t+60m	HGW	CPU/Memory Monitor Stop	00:17	02:54	09:15	10:35	21:50	23:13

Table J-3: Experiment 2 Scenario 3 Run Log

Time	Client	Type	Runs in October 2012					
			4th	4th	4th	4th	4th	4th
			1	2	3	4	5	6
t	HGW	CPU/Memory Monitor Start	00:14	01:18	02:30	03:31	04:32	06:00
t+2m	HGW	Aesop Start	00:16	01:20	02:32	03:33	04:34	06:02
t+4m	HGW	SECCO Start	00:18	01:22	02:34	03:35	04:36	06:04
t+6m	client2	VoD Children, High Quality Start	00:20	01:24	02:36	03:37	04:38	06:06
t+7m	client3	VoD Movie, High Quality Start	00:21	01:25	02:37	03:38	04:39	06:07
t+8m	client4	Social Network Web Start	00:22	01:26	02:38	03:39	04:40	06:08
t+9m	client1	Workplace Web Start	00:23	01:27	02:39	03:40	04:41	06:09
t+12m	Shaper	Delay Impairment Start	00:26	01:30	02:42	03:43	04:44	06:12
t+42m	Shaper	Delay Impairment Over	00:56	02:00	03:12	04:13	05:14	06:42
t+48m	client2	VoD Children, High Quality Stop	01:02	02:06	03:18	04:19	05:20	06:48
t+50m	client3	VoD Movie, High Quality Stop	01:04	02:08	03:20	04:21	05:22	06:50
t+52m	client4	Social Network Web Stop	01:06	02:10	03:22	04:23	05:24	06:52
t+54m	client1	Workplace Web Stop	01:08	02:12	03:24	04:25	05:26	06:54
t+56m	HGW	SECCO Stop	01:10	02:14	03:26	04:27	05:28	06:56
t+58m	HGW	Aesop Stop	01:12	02:16	03:28	04:29	05:30	06:58
t+60m	HGW	CPU/Memory Monitor Stop	01:14	02:18	03:30	04:31	05:32	07:00

J.3 Experiment 3

Table J-4: Experiment 3 Scenario 2 Run Log

Time	Client	Type	Runs in October 2012					
			6th	6th	6th	6th	6th to 7th	7th
			1	2	3	4	5	6
t	HGW	CPU/Memory Monitor Start	19:47	20:48	21:52	22:54	23:46	00:47
t+3m	HGW	Aesop Start	19:50	20:51	21:55	22:57	23:49	00:50
t+6m	HAN	SECCO Start	19:53	20:54	21:58	23:00	23:52	00:53
t+9m	client2	File Transfer Start	19:56	20:57	22:01	23:03	23:55	00:57
t+10m	client3	Social Network Web Start	19:57	20:58	22:02	23:04	23:56	00:58
t+11m	client4	VoD, Standard Quality Start	19:58	20:59	22:03	23:05	23:57	00:59
t+12m	client1	Workplace Web Start	19:59	21:00	22:04	23:06	23:58	01:00
t+45m	client2	File Transfer Stop	20:32	21:33	22:37	23:37[1]	00:31	01:30[2]
t+47m	client3	Social Network Web Stop	20:34	21:35	22:39	23:37[1]	00:33	01:34
t+49m	client4	VoD, Standard Quality Stop	20:36	21:37	22:41	23:37[1]	00:35	01:36
t+51m	client1	Workplace Web Stop	20:38	21:39	22:43	23:37[1]	00:37	01:38
t+54m	HAN	SECCO Stop	20:41	21:42	22:46	23:37[1]	00:40	01:41
t+57m	HGW	Aesop Stop	20:44	21:45	22:49	23:37[1]	00:43	01:44
t+60m	HGW	CPU/Memory Monitor Stop	20:47	21:48	22:52	23:37[1]	00:46	01:47

Observations:

1. In run 4, Aesop terminated abnormally near the end of the run due to a concurrency exception
2. The file transfer session was terminated 1 minute early in this run.

Table J-5: Experiment 3 Scenario 3 Run Log

Time	Client	Type	Runs in October 2012					
			7th	7th	7th	7th	7th	7th
			1	2	3	4	5	6
t	HGW	CPU/Memory Monitor Start	12:13	13:14	14:15	15:17	16:19	18:48
t+2m	HGW	Aesop Start	12:15	13:16	14:17	15:19	16:21	18:50
t+4m	HGW	SECCO Start	12:17	13:18	14:19	15:21	16:23	18:52
t+6m	client2	VoD Children, High Quality Start	12:19	13:20	14:21	15:23	16:25	18:54
t+7m	client3	VoD Movie, High Quality Start	12:20	13:21	14:22	15:24	16:26	18:55
t+8m	client4	Social Network Web Start	12:21	13:22	14:23	15:25	16:27	18:56
t+9m	client1	Workplace Web Start	12:22	13:23	14:24	15:26	16:28	18:57
t+12m	Shaper	Delay Impairment Start	12:25	13:26	14:27	15:29	16:31	19:00
t+42m	Shaper	Delay Impairment Over	12:55	13:56	14:57	15:59	17:01	19:30
t+48m	client2	VoD Children, High Quality Stop	13:01	14:02	15:03	16:05	17:07	19:36
t+50m	client3	VoD Movie, High Quality Stop	13:03	14:04	15:05	16:07	17:09	19:38
t+52m	client4	Social Network Web Stop	13:05	14:06	15:07	16:09	17:11	19:40
t+54m	client1	Workplace Web Stop	13:07	14:08	15:09	16:11	17:13	19:40
t+56m	HGW	SECCO Stop	13:09	14:10	15:11	16:13	17:15	19:44
t+58m	HGW	Aesop Stop	13:11	14:12	15:13	16:15	17:17	19:46
t+60m	HGW	CPU/Memory Monitor Stop	13:13	14:14	15:15	16:17	17:19	19:48

Appendix K
Samples of Collected Results

K.1 Samples of Simulation Results

```
1,/tmp/terminal_reports/20120106_043219.046-TR[081][810][008100].xml,1407,61,1346,2012-01-
      06 04:32:20.502
2,/tmp/terminal_reports/20120106_043219.046-TR[197][1970][019700].xml,470,34,436,2012-01-06
      04:32:20.976
3,/tmp/terminal_reports/20120106_043221.619-TR[006][060][000600].xml,641,30,611,2012-01-06
      04:32:22.617
4,/tmp/terminal_reports/20120106_043219.880-TR[100][1000][010000].xml,425,30,395,2012-01-06
      04:32:23.042
5,/tmp/terminal_reports/20120106_043219.470-TR[082][820][008200].xml,392,29,363,2012-01-06
      04:32:23.434
6,/tmp/terminal_reports/20120106_043220.756-TR[034][340][003400].xml,500,30,470,2012-01-06
      04:32:23.934
7,/tmp/terminal_reports/20120106_043222.427-TR[072][720][007200].xml,408,27,381,2012-01-06
      04:32:25.344
8,/tmp/terminal_reports/20120106_043223.515-TR[105][1050][010500].xml,341,25,316,2012-01-06
      04:32:25.685
9,/tmp/terminal_reports/20120106_043222.339-TR[196][1960][019600].xml,292,20,272,2012-01-06
      04:32:25.977
10,/tmp/terminal_reports/20120106_043224.819-TR[088][880][008800].xml,559,19,540,2012-01-06
      04:32:26.536
11,/tmp/terminal_reports/20120106_043224.373-TR[152][1520][015200].xml,308,27,281,2012-01-
      06 04:32:26.844
12,/tmp/terminal_reports/20120106_043223.692-TR[087][870][008700].xml,359,19,340,2012-01-06
      04:32:27.203
13,/tmp/terminal_reports/20120106_043222.970-TR[193][1930][019300].xml,525,24,501,2012-01-
      06 04:32:27.728
14,/tmp/terminal_reports/20120106_043222.643-TR[045][450][004500].xml,350,21,329,2012-01-06
      04:32:28.078
15,/tmp/terminal_reports/20120106_043222.723-TR[181][1810][018100].xml,383,19,364,2012-01-
      06 04:32:28.461
16,/tmp/terminal_reports/20120106_043224.370-TR[007][070][000700].xml,441,24,417,2012-01-06
      04:32:28.903
17,/tmp/terminal_reports/20120106_043222.023-TR[037][370][003700].xml,509,22,487,2012-01-06
      04:32:29.412
18,/tmp/terminal_reports/20120106_043222.511-TR[078][780][007800].xml,674,20,654,2012-01-06
      04:32:30.086
19,/tmp/terminal_reports/20120106_043221.985-TR[160][1600][016000].xml,308,21,287,2012-01-
      06 04:32:30.395
20,/tmp/terminal_reports/20120106_043222.089-TR[030][300][003000].xml,533,20,513,2012-01-06
      04:32:30.928
21,/tmp/terminal_reports/20120106_043224.130-TR[052][520][005200].xml,308,20,288,2012-01-06
      04:32:31.237
22,/tmp/terminal_reports/20120106_043228.710-TR[121][1210][012100].xml,356,15,341,2012-01-
      06 04:32:32.594
23,/tmp/terminal_reports/20120106_043227.622-TR[101][1010][010100].xml,409,9,400,2012-01-06
      04:32:33.004
24,/tmp/terminal_reports/20120106_043231.138-TR[029][290][002900].xml,325,11,314,2012-01-06
      04:32:33.329
25,/tmp/terminal_reports/20120106_043229.608-TR[171][1710][017100].xml,300,11,289,2012-01-
      06 04:32:33.63
26,/tmp/terminal_reports/20120106_043229.329-TR[060][600][006000].xml,550,14,536,2012-01-06
      04:32:34.18
27,/tmp/terminal_reports/20120106_043231.476-TR[080][800][008000].xml,341,13,328,2012-01-06
      04:32:34.521
28,/tmp/terminal_reports/20120106_043231.067-TR[031][310][003100].xml,450,14,436,2012-01-06
      04:32:34.972
29,/tmp/terminal_reports/20120106_043229.647-TR[184][1840][018400].xml,316,12,304,2012-01-
      06 04:32:35.288
30,/tmp/terminal_reports/20120106_043227.767-TR[180][1800][018000].xml,466,12,454,2012-01-
      06 04:32:35.755
```

Listing K-1: Experiment 1 Run 7 Test 9 Monitoring Log, first 30 records

```
Kicking off 200 IPTV sessions
Terminal reporting delay period of 7563ms for [0][0][0]
Terminal reporting delay period of 58346ms for [2][20][200]
Terminal reporting delay period of 11688ms for [1][10][100]
Terminal reporting delay period of 54257ms for [3][30][300]
Terminal reporting delay period of 6924ms for [4][40][400]
Terminal reporting delay period of 14988ms for [5][50][500]
Terminal reporting delay period of 3108ms for [6][60][600]
Terminal reporting delay period of 5863ms for [7][70][700]
Terminal reporting delay period of 18865ms for [8][80][800]
Terminal reporting delay period of 38219ms for [9][90][900]
Terminal reporting delay period of 21580ms for [10][100][1000]
Terminal reporting delay period of 14405ms for [11][110][1100]
Terminal reporting delay period of 23954ms for [12][120][1200]
Terminal reporting delay period of 29939ms for [13][130][1300]
Terminal reporting delay period of 8112ms for [14][140][1400]
Terminal reporting delay period of 52815ms for [15][150][1500]
Terminal reporting delay period of 49714ms for [16][160][1600]
Terminal reporting delay period of 41506ms for [17][170][1700]
Terminal reporting delay period of 22716ms for [18][180][1800]
Terminal reporting delay period of 34808ms for [19][190][1900]
Terminal reporting delay period of 30575ms for [20][200][2000]
Terminal reporting delay period of 51936ms for [21][210][2100]
Terminal reporting delay period of 30096ms for [22][220][2200]
Terminal reporting delay period of 23631ms for [23][230][2300]
Terminal reporting delay period of 14191ms for [24][240][2400]
Terminal reporting delay period of 44165ms for [25][250][2500]
Terminal reporting delay period of 51792ms for [26][260][2600]
Terminal reporting delay period of 10382ms for [27][270][2700]
```

Listing K-2: Experiment 1 Run 7 Test 9 Simulation Log, first 30 records

K.2 Samples of Test Bed Results

```
liam$ psql -U postgres -h localhost -d aesop_e3s3run2
psql (9.1.4)
Type "help" for help.

aesop_e3s3run2=# copy (select * from web_sessions) to stdout with csv header;
id,address,service,user_name,priority,terminated,throttled,t_limit_down,t_limit_up,first_ur
      l
1349612590629_WEB_client1,159.107.219.145,Web
      Browsing,AlexHan,1,t,f,0,0,http://workplacetestwebsite/workplace/?page_id=110
1349612530836_WEB_client4,159.107.219.148,Web
      Browsing,JoeHan,3,t,f,0,0,http://socialnetworktestwebsite/~socialnetwork/?p=74
```

Listing K-3: Experiment 3 Scenario 3 Run 2 Web Sessions

```
aesop_e3s3run2=# copy (select * from web_metrics limit 30) to stdout with csv header;
start_date,service,session_id,compliance,period,duration,bytes_up,bytes_down,user_name,conn
      ection_id,report_type,requests,ongoing,finished,cached,aborts,errors,rltm,rltsd,rsm,r
      ssd,status,throttled
2012-10-07 13:29:08,Web Browsing,1349612530836_WEB_client4,99.742,52928,430516,0,133216,Joe
      Han,terminal_004_client4,PERIODIC,0,1,1,0,1,0,60155,0,102153898,0,(Aborted),f
2012-10-07 13:22:07,Web Browsing,1349612530836_WEB_client4,100,5607,5607,9188,18519243,Joe
      Han,terminal_004_client4,STARTED,25,2,23,0,4,0,1974,4404,2758284,7688179,200,f
2012-10-07 13:22:17,Web
      Browsing,1349612530836_WEB_client4,100,16599,22206,3681,14927193,Joe
      Han,terminal_004_client4,PERIODIC,5,2,5,0,0,0,3525,8393,5070029,12411299,200,f
2012-10-07 13:29:05,Web
      Browsing,1349612590629_WEB_client1,28.074,10341,366255,0,859244,Alex
      Han,terminal_001_client1,PERIODIC,0,2,8,0,0,0,7124,4842,164494,120117,200,f
2012-10-07 13:22:27,Web Browsing,1349612530836_WEB_client4,100,4386,26592,576,4761424,Joe
      Han,terminal_004_client4,PERIODIC,2,1,3,0,0,0,7003,12027,10134963,17553924,200,f
2012-10-07 13:22:37,Web Browsing,1349612530836_WEB_client4,100,8686,35278,3562,10336089,Joe
      Han,terminal_004_client4,PERIODIC,10,2,9,0,0,0,3948,10793,4638245,13026809,200,f
2012-10-07 13:22:47,Web Browsing,1349612530836_WEB_client4,100,0,35278,0,14031120,Joe
      Han,terminal_004_client4,PERIODIC,0,2,0,0,0,0,28202,0,36877675,0,200,f
2012-10-07 13:29:18,Web
      Browsing,1349612530836_WEB_client4,66.335,12678,443194,5377,66113,Joe
      Han,terminal_004_client4,PERIODIC,15,8,8,0,0,0,1206,721,7074,17210,*,f
2012-10-07 13:22:57,Web Browsing,1349612530836_WEB_client4,100,26059,61337,0,12710544,Joe
      Han,terminal_004_client4,PERIODIC,0,1,1,0,0,0,26162,0,36877675,0,200,f
2012-10-07 13:23:07,Web Browsing,1349612530836_WEB_client4,100,8745,70082,3874,3222381,Joe
      Han,terminal_004_client4,PERIODIC,11,2,10,0,0,0,3645,9983,3018068,8444316,200,f
2012-10-07 13:23:04,Web Browsing,1349612590629_WEB_client1,100,6829,6829,16635,928699,Alex
      Han,terminal_001_client1,STARTED,50,1,49,0,1,0,221,295,31212,49779,200,f
2012-10-07 13:23:17,Web Browsing,1349612530836_WEB_client4,100,0,70082,0,12106728,Joe
      Han,terminal_004_client4,PERIODIC,0,2,0,0,0,0,21475,0,23916254,0,200,f
2012-10-07 13:23:14,Web
      Browsing,1349612590629_WEB_client1,100,9816,16645,25672,3460487,Alex
      Han,terminal_001_client1,PERIODIC,75,2,74,0,0,0,399,419,68107,97116,200,f
2012-10-07 13:23:27,Web Browsing,1349612530836_WEB_client4,100,19757,89839,0,8786464,Joe
      Han,terminal_004_client4,PERIODIC,0,1,1,0,0,0,19811,0,23916254,0,200,f
2012-10-07 13:23:24,Web
      Browsing,1349612590629_WEB_client1,100,10730,27375,30008,3683509,Alex
      Han,terminal_001_client1,PERIODIC,88,9,81,0,0,0,453,476,80088,105769,200,f
2012-10-07 13:26:37,Web Browsing,1349612530836_WEB_client4,95.957,0,254819,0,7083616,Joe
      Han,terminal_004_client4,PERIODIC,0,2,0,0,0,0,62528,0,55699356,0,200,f
2012-10-07 13:26:35,Web
      Browsing,1349612590629_WEB_client1,98.28,10544,217565,23791,2871208,Alex
      Han,terminal_001_client1,PERIODIC,70,9,71,0,0,0,814,1235,78557,104171,200,f
2012-10-07 13:26:47,Web Browsing,1349612530836_WEB_client4,92.16,0,254819,0,8021920,Joe
      Han,terminal_004_client4,PERIODIC,0,2,0,0,0,0,65104,0,55699356,0,200,f
2012-10-07 13:26:45,Web
      Browsing,1349612590629_WEB_client1,100,9954,227519,17582,2275792,Alex
      Han,terminal_001_client1,PERIODIC,52,2,59,0,0,0,508,437,70616,96515,200,f
2012-10-07 13:27:55,Web
      Browsing,1349612590629_WEB_client1,75.973,10183,296400,12635,2170073,Alex
      Han,terminal_001_client1,PERIODIC,37,2,37,0,0,0,1053,949,89970,113486,*,f
2012-10-07 13:27:47,Web Browsing,1349612530836_WEB_client4,100,0,334052,0,6472560,Joe
      Han,terminal_004_client4,PERIODIC,0,2,0,0,0,0,39713,0,20174343,0,200,f
2012-10-07 13:27:45,Web
      Browsing,1349612590629_WEB_client1,100,9320,286217,20165,2756113,Alex
      Han,terminal_001_client1,PERIODIC,59,2,58,0,0,0,479,471,67373,96529,*,f
2012-10-07 13:27:37,Web Browsing,1349612530836_WEB_client4,100,250,334052,0,8661709,Joe
      Han,terminal_004_client4,PERIODIC,0,2,1,0,0,0,28253,23831,10596130,13545638,200,f
2012-10-07 13:27:35,Web
      Browsing,1349612590629_WEB_client1,100,10544,276897,12545,913886,Alex
      Han,terminal_001_client1,PERIODIC,37,1,39,0,0,0,684,652,38638,52921,200,f
2012-10-07 13:28:47,Web Browsing,1349612530836_WEB_client4,39.464,0,377588,0,7881464,Joe
      Han,terminal_004_client4,PERIODIC,0,2,0,0,0,0,152036,0,102153898,0,200,f
2012-10-07 13:28:45,Web
      Browsing,1349612590629_WEB_client1,100,10363,348022,13335,716709,Alex
      Han,terminal_001_client1,PERIODIC,40,2,40,0,0,0,600,425,32268,50169,200,f
2012-10-07 13:28:37,Web Browsing,1349612530836_WEB_client4,37.458,0,377588,0,7432584,Joe
      Han,terminal_004_client4,PERIODIC,0,2,0,0,0,0,160181,0,102153898,0,200,f
2012-10-07 13:28:35,Web
      Browsing,1349612590629_WEB_client1,100,10080,337659,17211,2369645,Alex
      Han,terminal_001_client1,PERIODIC,51,2,51,0,0,0,687,685,76352,105735,*,f
2012-10-07 13:30:35,Web
      Browsing,1349612590629_WEB_client1,100,10113,457706,26891,3246036,Alex
      Han,terminal_001_client1,PERIODIC,79,2,79,0,0,0,358,439,61875,89534,302,f
2012-10-07 13:39:55,Web Browsing,1349612590629_WEB_client1,2.865,8919,1016497,0,694174,Alex
      Han,terminal_001_client1,PERIODIC,0,2,4,0,0,0,69796,23789,187226,131156,200,f
```

Listing K-4: Experiment 3 Scenario 3 Run 2 Web Metrics, first 30 records

```
2012-10-07 13:42:18,686 [TRLoaderThread-723] INFO
     com.ericsson.aesop.terminalreport.TRLoaderThread - TRLoaderThread-723
     starting . . .
2012-10-07 13:42:18,925 [TRLoaderThread-724] INFO
     com.ericsson.aesop.terminalreport.TRLoaderThread - TRLoaderThread-724
     starting . . .
2012-10-07 13:42:19,727 [TRLoaderThread-723] INFO
     com.ericsson.aesop.terminalreport.TRLoaderThread - TRLoaderThread-723 ran
     for 1041ms
2012-10-07 13:42:19,953 [TRLoaderThread-724] INFO
     com.ericsson.aesop.terminalreport.TRLoaderThread - TRLoaderThread-724 ran
     for 1028ms
2012-10-07 13:42:21,934 [AnalysePeriodicOperation] INFO
     com.ericsson.aesop.model.AnalysePeriodicOperation - analysis run started . .
     .
2012-10-07 13:42:22,346 [TRLoaderThread-725] INFO
     com.ericsson.aesop.terminalreport.TRLoaderThread - TRLoaderThread-725
     starting . . .
2012-10-07 13:42:22,964 [OptimisePeriodicOperation] INFO
     com.ericsson.aesop.model.OptimisePeriodicOperation - optimization run
     started . . .
2012-10-07 13:42:22,964 [AnalysePeriodicOperation] INFO
     com.ericsson.aesop.model.AnalysePeriodicOperation - analysis run completed,
     time taken:1030
2012-10-07 13:42:22,964 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation -
     checkAndApplyThrottling()-->
2012-10-07 13:42:22,964 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation - precheckAndGetWindow()-
     ->
2012-10-07 13:42:23,074 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation -
     precheckAndGetWindow():snapshot buckets span window, continuing . . .
2012-10-07 13:42:23,362 [TRLoaderThread-725] INFO
     com.ericsson.aesop.terminalreport.TRLoaderThread - TRLoaderThread-725 ran
     for 1016ms
2012-10-07 13:42:23,384 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation -
     precheckAndGetWindow():no new/deleted sessions in window, continuing . . .
2012-10-07 13:42:23,388 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation -
     precheckAndGetWindow():no throttling action in window, continung . . .
2012-10-07 13:42:23,468 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation -
     checkAndApplyThrottling(): non-compliant sessions in all snapshot buckets,
     continuing . . .
2012-10-07 13:42:23,497 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation -
     checkAndApplyThrottling():applying throttling to session:
     http://aesop/ServiceAnalysisABoxDynamic.owl#SSM_Session_1349612403551_VOD_cl
     ient2 in bucket 2012-10-
     07T12:41:40Z^^http://www.w3.org/2001/XMLSchema#dateTime . . .
2012-10-07 13:42:23,497 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.throttling.ThrottlingHandler - applyThrottling()-->
2012-10-07 13:42:23,497 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.throttling.ThrottlingProcess - ThrottlingProcess()-
     >[Ljava.lang.String;@f07d505
2012-10-07 13:42:23,497 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.throttling.ThrottlingProcess - starting process:
     [bin/throttle_apply.sh, 159.107.219.146, 146, 2000, 50, 2]
2012-10-07 13:42:23,541 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.throttling.ThrottlingProcess - process finished:
     result=2000,50
2012-10-07 13:42:23,541 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.throttling.ThrottlingHandler - applyThrottling():
     throttling applied
2012-10-07 13:42:23,541 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.throttling.ThrottlingHandler - applyThrottling()<--
2012-10-07 13:42:23,550 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation -
     checkAndApplyThrottling():throttled session:
     http://aesop/ServiceAnalysisABoxDynamic.owl#SSM_Session_1349612403551_VOD_cl
     ient2 in bucket 2012-10-
```

```
     07T12:41:40Z^^http://www.w3.org/2001/XMLSchema#dateTime,ThrottlingTargetSess
     ion [id=1349612403551_VOD_client2, priority=2, bucketBeginning=Sun Oct 07
     13:41:40 IST 2012, compliance=18.333333333333336, serviceName=Video on
     Demand, hostAddress=159.107.219.146, userId=SamHan, bytesDownstream=1165173,
     bytesUpstream=0, limitDownstream=2000000, limitUpstream=50000,
     appliedLimitDownstream=2000000, appliedLimitUpstream=50000]
2012-10-07 13:42:23,550 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation -
     checkAndApplyThrottling()<--
2012-10-07 13:42:23,550 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation -
     checkAndReleaseThrottling()-->
2012-10-07 13:42:23,550 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation - precheckAndGetWindow()-
     ->
2012-10-07 13:42:23,651 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation -
     precheckAndGetWindow():snapshot buckets span window, continuing . . .
2012-10-07 13:42:24,111 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation -
     precheckAndGetWindow():no new/deleted sessions in window, continuing . . .
2012-10-07 13:42:24,115 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation -
     precheckAndGetWindow()<--throttling action already applied in window, stop
2012-10-07 13:42:24,115 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation -
     checkAndReleaseThrottling()<--pre-checks precluded optimization, stop
2012-10-07 13:42:24,115 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation -
     releaseThrottlingOnTerminatedSessions()-->
2012-10-07 13:42:25,819 [TRLoaderThread-726] INFO
     com.ericsson.aesop.terminalreport.TRLoaderThread - TRLoaderThread-726
     starting . . .
2012-10-07 13:42:25,903 [TRLoaderThread-727] INFO
     com.ericsson.aesop.terminalreport.TRLoaderThread - TRLoaderThread-727
     starting . . .
2012-10-07 13:42:26,220 [OptimisePeriodicOperation] DEBUG
     com.ericsson.aesop.model.OptimisePeriodicOperation -
     releaseThrottlingOnTerminatedSessions()<--no throttled terminated sessions
     found, stop
2012-10-07 13:42:26,220 [OptimisePeriodicOperation] INFO
     com.ericsson.aesop.model.OptimisePeriodicOperation - optimization run
     completed, time taken:3256
2012-10-07 13:42:26,896 [TRLoaderThread-726] INFO
     com.ericsson.aesop.terminalreport.TRLoaderThread - TRLoaderThread-726 ran
     for 1077ms
2012-10-07 13:42:27,047 [TRLoaderThread-727] INFO
     com.ericsson.aesop.terminalreport.TRLoaderThread - TRLoaderThread-727 ran
     for 1144ms
```

Listing K-5: Experiment 3 Scenario 3 Run 2 Aesop Performance Log Fragment

The log fragment above shows the timings for monitoring of five terminal reports, which were handled by *TrLoaderThread* 723, 724, 725, 726, and 727 respectively. It also shows timings for an analysis run that started at 13:42:21.934 and completed at 13:42:22.964. An optimization planning and execution run started at 13:42:22.964 and finished at 13:42:26.220. That optimization run executed a throttling action that limited a Video on Demand session being run by the user *Sam Han*.

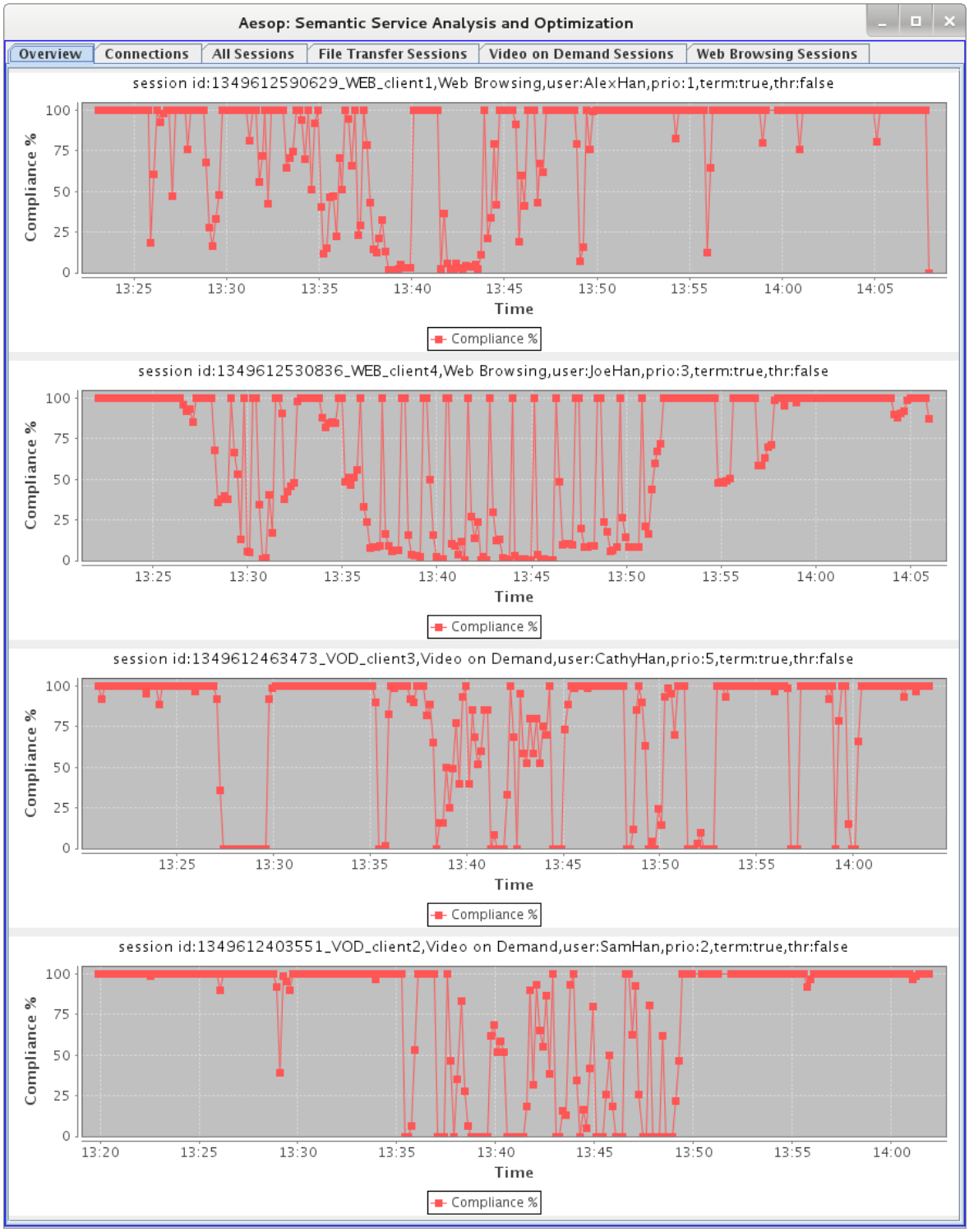


Figure K-1: Screenshot of the Aesop client at the end of Experiment 3 Scenario 3 Run 2

```
<?xml version="1.0" encoding="UTF-8"?>
<TerminalReport xmlns="http://aesop/TerminalReport">
  <ID>terminal_004_2012-10-07T12_42_18.680</ID>
  <StartTime>2012-10-07T12:42:08.674</StartTime>
  <Duration>10006</Duration>
  <ReportTrigger>Periodic Terminal Report</ReportTrigger>
  <Terminal>
    <ID>terminal_004</ID>
    <Name>client4</Name>
    <Type>tablet</Type>
    <Manufacturer>HP</Manufacturer>
    <Model>HP Compaq tc4200</Model>
    <SerialNumber>CND542150Z</SerialNumber>
    <Location>
      <Latitude>53.429174</Latitude>
      <Longitude>-8.02062</Longitude>
      <Altitude>151.0</Altitude>
    </Location>
    <Connection>
      <ID>client4</ID>
      <Interface>p2p1</Interface>
      <ConnectivityService>ZestcomDSLServicePackageGold</ConnectivityService>
      <Subscription>Sub_SamHanZestcomDSLGO</Subscription>
      <Address>159.107.219.148</Address>
      <MACAddress>00:0f:b0:7a:70:78</MACAddress>
      <Metrics>
        <ConnectionRequests>0</ConnectionRequests>
        <Terminations>0</Terminations>
        <PacketsUploaded>114</PacketsUploaded>
        <PacketsDownloaded>133</PacketsDownloaded>
        <BytesUploaded>9046</BytesUploaded>
        <BytesDownloaded>132508</BytesDownloaded>
        <PacketLossRatio>0.0</PacketLossRatio>
        <PacketErrorRatio>0.0</PacketErrorRatio>
        <Latency>7604.375</Latency>
        <Jitter>0.017</Jitter>
      </Metrics>
    </Connection>
    <Session>
      <ID>1349612530836_WEB_client4</ID>
      <ConnectionID>client4</ConnectionID>
      <ReportType>Periodic</ReportType>
      <User>
        <ID>JoeHan</ID>
      </User>
      <Metrics>
        <StartTime>2012-10-07T12:22:03.803</StartTime>
        <Duration>1203961</Duration>
        <ReportDuration>0</ReportDuration>
        <AuthenticationExecuted>false</AuthenticationExecuted>
        <Established>true</Established>
        <EstablishmentDuration>345</EstablishmentDuration>
        <DataTransferDuration>1203616</DataTransferDuration>
        <BytesUploaded>312</BytesUploaded>
        <BytesDownloaded>182</BytesDownloaded>
        <WebBrowsingMetrics>
          <RequestCount>1</RequestCount>
          <OngoingCount>8</OngoingCount>
          <FinishCount>1</FinishCount>
          <CacheCount>0</CacheCount>
          <AbortCount>0</AbortCount>
          <ErrorCount>0</ErrorCount>
          <RequestLoadTimeMean>3358</RequestLoadTimeMean>
          <RequestLoadTimeStdDev>0</RequestLoadTimeStdDev>
          <RequestSizeMean>182</RequestSizeMean>
          <RequestSizeStdDev>0</RequestSizeStdDev>
          <LastStatus>200</LastStatus>
        </WebBrowsingMetrics>
      </Metrics>
      <WebBrowsingSession>
        <FirstURL>http://socialnetworktestwebsite/~socialnetwork/?p=74</FirstURL>
      </WebBrowsingSession>
    </Session>
  </Terminal>
</TerminalReport>
```

Listing K-6: Experiment 3 Scenario 3 Run 2 Terminal Report

This is the terminal report monitored by *TrLoaderThread* 723. It gives details on a Web Browsing session being run by the user *Joe Han* on client 4.

```
12:41:01 AM     all      4.68      0.00      0.27      0.21      0.00     94.84
12:41:11 AM     all      4.84      0.00      0.36      0.20      0.00     94.60
12:41:21 AM     all      4.60      0.00      0.26      0.19      0.00     94.95
12:41:31 AM     all      4.63      0.00      0.26      0.18      0.00     94.93
12:41:41 AM     all      4.79      0.00      0.32      0.35      0.00     94.54
12:41:51 AM     all      4.78      0.00      0.26      0.15      0.00     94.81
12:42:01 AM     all      4.70      0.00      0.26      0.20      0.00     94.84
12:42:11 AM     all      3.83      0.00      0.26      0.19      0.00     95.72
12:42:21 AM     all      4.33      0.00      0.30      0.28      0.00     95.10
12:42:31 AM     all      4.38      0.00      0.24      0.13      0.00     95.26
12:42:41 AM     all      4.52      0.00      0.36      0.14      0.00     94.98
12:42:51 AM     all      4.39      0.00      0.24      0.16      0.00     95.21
12:43:01 AM     all      4.78      0.00      0.36      0.16      0.00     94.69
12:43:11 AM     all      2.82      0.00      0.29      0.21      0.00     96.68
12:43:21 AM     all      0.13      0.00      0.15      0.15      0.00     99.57
12:43:31 AM     all      0.35      0.00      0.15      0.05      0.00     99.45
12:43:41 AM     all      0.22      0.00      0.17      0.09      0.00     99.51
12:43:51 AM     all      0.14      0.00      0.09      0.13      0.00     99.65
12:44:01 AM     all      0.22      0.00      0.15      0.09      0.00     99.54
```

Listing K-7: Experiment 3 Scenario 3 Run 2 CPU Usage from 12:41 to 12:44 (in UTC)

```
12:41:01 AM  19007484   5666412     22.97    330412   1465816   5208340     10.12
     4620432    583612
12:41:11 AM  19007296   5666600     22.97    330416   1465860   5208504     10.12
     4620628    583624
12:41:21 AM  19008536   5665360     22.96    330420   1465864   5207800     10.12
     4619428    583612
12:41:31 AM  19008536   5665360     22.96    330420   1465884   5207672     10.12
     4619696    583604
12:41:41 AM  19008024   5665872     22.96    330424   1465904   5207040     10.12
     4619476    583612
12:41:51 AM  19008148   5665748     22.96    330428   1465908   5208704     10.12
     4619424    583612
12:42:01 AM  19008980   5664916     22.96    330428   1465940   5209320     10.13
     4618996    583612
12:42:11 AM  19008964   5664932     22.96    330432   1465980   5209320     10.13
     4619784    583612
12:42:21 AM  19002768   5671128     22.98    330436   1465996   5206656     10.12
     4625256    583604
12:42:31 AM  19002768   5671128     22.98    330440   1466016   5206528     10.12
     4625260    583604
12:42:41 AM  19002768   5671128     22.98    330444   1466032   5206716     10.12
     4625352    583608
12:42:51 AM  19003512   5670384     22.98    330448   1466048   5206012     10.12
     4624208    583612
12:43:01 AM  19322380   5351516     21.69    330452   1462288   4809236      9.35
     4311136    579836
12:43:11 AM  21868596   2805300     11.37    330460   1462244   1925388      3.74
     1771100    579780
12:43:21 AM  21868868   2805028     11.37    330464   1462244   1925380      3.74
     1771332    579816
12:43:31 AM  21869116   2804780     11.37    330464   1462248   1925380      3.74
     1771156    579800
12:43:41 AM  21868744   2805152     11.37    330464   1462252   1925380      3.74
     1771588    579836
12:43:51 AM  21869240   2804656     11.37    330464   1462252   1925372      3.74
     1771324    579796
12:44:01 AM  21869272   2804624     11.37    330464   1462252   1925364      3.74
     1771072    579792
```

Listing K-8: Experiment 3 Scenario 3 Run 2 Memory Usage from 12:41 to 12:44 (in UTC)

Note: The CPU and Memory usage logs in Listing K-7 and Listing K-8 cover the same time period as Listing K-5. The hour difference in time stamps is due to a 1 hour difference between UTC and Irish daylight saving time.

Appendix L
Examples of Analysis Scripts and Queries

```
#! /bin/bash

set -v

result_home=$HOME/git/ssao-main/results/testbed

rm -fr $result_home/processed/*

#
# Create database for aggregate data
#
psql -h localhost -U postgres -f $result_home/bin/define_result_db.sql -v
     db_id=aesop_aggregate

#
# Create aggregate tables
#
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e2s2_ft
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e2s2_vod
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e2s2_webs
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e2s2_webw
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e3s2_ft
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e3s2_vod
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e3s2_webs
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e3s2_webw
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e2s3_vodc
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e2s3_vodt
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e2s3_webs
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e2s3_webw
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e3s3_vodc
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e3s3_vodt
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e3s3_webs
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e3s3_webw

#
# Create scenario aggregate tables
#
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e2s2
```

```
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e2s3
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e3s2
psql -h localhost -U aesop -d aesop_aggregate -f
     $result_home/bin/define_aggregate_table.sql -v table_id=e3s3

$result_home/bin/import_all.sh e2s2
$result_home/bin/import_all.sh e2s3
$result_home/bin/import_all.sh e3s2
$result_home/bin/import_all.sh e3s3

$result_home/bin/plot_all.sh e2s2
$result_home/bin/plot_all.sh e2s3
$result_home/bin/plot_all.sh e3s2
$result_home/bin/plot_all.sh e3s3

#
# Plot load, analysis, optimization, CPU, and memory
#
cd $result_home/processed
gnuplot $result_home/bin/plot_load_anal_e2.gnu
gnuplot $result_home/bin/plot_cpu_mem_e2.gnu
gnuplot $result_home/bin/plot_load_anal_opti_e3.gnu
gnuplot $result_home/bin/plot_cpu_mem_e3.gnu
cd $result_home
rm $result_home/processed/summary_*.csv

#
# Plot overall aggregated statistics
#
cd $result_home/processed
gnuplot $result_home/bin/plot_compliance.gnu
cd $result_home
rm $result_home/processed/agg_*.csv

#
# Output overall scenario statistics
#
psql --quiet -h localhost -U aesop -d aesop_aggregate -f bin/effectiveness.sql -v
     scenario=s2 | grep -v '^Tot' > $result_home/processed/eff_s2.csv
psql --quiet -h localhost -U aesop -d aesop_aggregate -f bin/effectiveness.sql -v
     scenario=s3 | grep -v '^Tot' > $result_home/processed/eff_s3.csv

#
# Plot overall scenario statistics
#
cd $result_home/processed
gnuplot $result_home/bin/plot_effectiveness.gnu
cd $result_home

rm -fr $result_home/processed/eff_*.csv
rm -f $result_home/processed/congestion.csv
rm -f $result_home/processed/impairment.csv
```

Listing L-1: The process_all.sh script for Experiment 2 and Experiment 3

This script kicks off processing of all result data for Experiments 2 and 3. It analyses the result data, produces summary tables, and generates plots.

```
#! /opt/local/bin/perl

use strict;
use Time::Piece;

if ($#ARGV != 1) {
  print "usage: parse_sysstat input_dir start_tag\n";
  exit;
}

my $sysstat_file_dir=$ARGV[0];
my $start_tag=$ARGV[1];

opendir (SYSSTAT_FILE_DIR, $sysstat_file_dir) or die $!;

my @output_lines;

while (my $file = readdir(SYSSTAT_FILE_DIR)) {
  if ($file !~ m/^$start_tag/) {
    next;
  }

  process_sysstat_file($sysstat_file_dir . "/" . $file);
}

my @sorted_lines = sort @output_lines;

print map { "$_ \n" } @sorted_lines;

sub process_sysstat_file
  {
    open(SYSSTAT_FILE, @_[0]) or die @_[0] . ":" . $!;
    my @lines= <SYSSTAT_FILE>;
    close(SYSSTAT_FILE);

    chomp @lines;

    # First line has date in second field in US format
    my $date = $lines[0];
    $date =~ s/^.*\t([0-1][0-9]\/[0-3][0-9]\/[0-9][0-9][0-9][0-9]).*/$1/;

    for (my $i = 3; $i < @lines - 1; $i++) {
      my @line_split = split '\s+', $lines[$i];
      my $date_time = $date . " " . $line_split[0] . " " . $line_split[1];

      my $date_val = Time::Piece->strptime($date_time, "%m/%d/%Y %I:%M:%S %p");

      my $new_line = $date_val->strftime("%Y-%m-%d %H:%M:%S");

      for (my $j = 2; $j < @line_split; $j++) {
     $new_line = $new_line . ',' . $line_split[$j];
      }
      push @output_lines, $new_line;
    }
  }
```

Listing L-2: The parse_sysstat.pl script for Experiment 2 and Experiment 3

This script parses the output of the Linux "sysstat" command, producing comma-separated output that can be imported into a SQL database.

```
DROP TABLE IF EXISTS aggregate_:scenario;

CREATE TABLE aggregate_:scenario
(
  minute integer,
  session_tag text,
  priority integer,
  e2_met_count integer,
  e3_met_count integer,
  e3_thr_count integer,
  compliance_before double precision,
  compliance_after double precision,
  throttled boolean
 );
ALTER TABLE aggregate_:scenario
  OWNER TO aesop;

INSERT INTO aggregate_:scenario
SELECT
  e2.minute,
  e2.session_tag,
  e2.priority,
  e2.met_count AS e2_met_count,
  e3.met_count AS e2_met_count,
  e3.thr_count AS e3_thr_count,
  e2.avg_comp  AS e2_avg_comp,
  e3.avg_comp  AS e3_avg_comp,
  (CASE e3.thr_count >= e3.met_count / 2 WHEN true THEN true ELSE false END) AS
     throttled
FROM
  (SELECT
    CAST(metric_time AS INT) AS minute,
    regexp_replace(session_id, '^[0-9]*_(...)_.*([0-9])$', E'\\1_client\\2') as
     session_tag,
    priority,
    COUNT(*) AS met_count,
    ROUND(CAST (AVG(compliance) AS NUMERIC), 3) AS avg_comp
    FROM e2:scenario
    GROUP BY minute, session_tag, priority
    ORDER BY minute, session_tag, priority
  )
  AS e2,
  (SELECT
    CAST(metric_time AS INT) AS minute,
    regexp_replace(session_id, '^[0-9]*_(...)_.*([0-9])$', E'\\1_client\\2') as
     session_tag,
    priority,
    COUNT(*) AS met_count,
    COUNT(CASE throttled WHEN true THEN 1 ELSE null END) AS thr_count,
    ROUND(CAST (AVG(compliance) AS NUMERIC), 3) AS avg_comp
    FROM e3:scenario
    GROUP BY minute, session_tag, priority
    ORDER BY minute, session_tag, priority
  ) AS e3
WHERE e2.minute      = e3.minute      AND
      e2.session_tag = e3.session_tag AND
      e2.priority    = e3.priority;

DROP TABLE IF EXISTS :scenario;
CREATE TABLE :scenario
(
  minute integer,
  avg_comp_before double precision,
  avg_comp_after  double precision,
  effectiveness   double precision
 );
ALTER TABLE :scenario
  OWNER TO aesop;

INSERT INTO :scenario
  SELECT
    aggregate_:scenario.minute,
    ROUND(CAST (AVG(compliance_before) AS NUMERIC), 3) AS avg_comp_before,
```

```
    ROUND(CAST (AVG(compliance_after)  AS NUMERIC), 3) AS avg_comp_after
  FROM aggregate_:scenario
  WHERE aggregate_:scenario.throttled = false
  GROUP BY aggregate_:scenario.minute
  ORDER BY aggregate_:scenario.minute;

UPDATE :scenario
SET effectiveness = round(cast ((avg_comp_after - avg_comp_before) * 100 /
     avg_comp_before AS NUMERIC), 3);

COPY (
  SELECT * from :scenario
) TO STDOUT WITH CSV;

COPY (
  SELECT
    'Total',
    ROUND(CAST (AVG(avg_comp_before) AS NUMERIC), 3) AS tot_avg_before,
    ROUND(CAST (AVG(avg_comp_after)  AS NUMERIC), 3) AS tot_avg_after,
    ROUND(CAST (AVG(effectiveness)   AS NUMERIC), 3) AS tot_effectiveness
  FROM :scenario
) TO STDOUT WITH CSV;

COPY (
  SELECT
    'Tot below 50',
    ROUND(CAST (AVG(avg_comp_before) AS NUMERIC), 3) AS tot_avg_before,
    ROUND(CAST (AVG(avg_comp_after)  AS NUMERIC), 3) AS tot_avg_after,
    ROUND(CAST (AVG(effectiveness)   AS NUMERIC), 3) AS tot_effectiveness
  FROM :scenario
  WHERE avg_comp_before < 50.0
) TO STDOUT WITH CSV;
```

Listing L-3: The effectiveness.sql SQL script for Experiment 3

The SQL queries in this SQL script calculate the effectiveness of optimization for each minute of an experimental run by comparing equivalent sessions in Experiment 2 and Experiment 3. This script implements the pseudo code shown in Listing 6-1.

```
set terminal postscript eps enhanced color
set size 1.0,1.0
set datafile separator ","
set style data linespoints
set grid xtics ytics

set xrange [0:60]

set output "effectiveness.ps"

set multiplot layout 2,1
set key at 53,370 horizontal
set size 1.0,0.49
set xtics 2 format ""
set ylabel "%" offset 1.5,0
set yrange [-20:340]
set ytics 50
set origin 0.0,0.49
set title "Scenario 2" offset -32,-3
plot "eff_s2.csv" using 1:4 lt 1 lc 1 title "Effectiveness",\
     "congestion.csv" using 1:(($2-20)*4.25) lt 1 lc 2 lw 2 with lines title
     "Sess Start:Stop, % Bandwidth Avail"

set key off
set ylabel "%" offset 1.5,0
set yrange [-20:340]
set ytics 50
set origin -0.005,0.0
set size 1.003,0.52
set xtics format "% g"
set title "Scenario 3" offset -32,-3
plot "eff_s3.csv" using 1:4 lt 1 lc 1 title "Effectiveness",\
     "impairment.csv" using 1:($2*18.88888) lt 1 lc 2 lw 2 with lines title "%
     Bandwidth Avail."
```

Listing L-4: The plot_effectiveness.gnu gnuplot script for Experiment 3

This script uses the “gnuplot” utility to produce the plot in Figure 6-25.

Appendix M
Parameters Collected in Experiments

M.1 Experiments in Simulated Environment

Table M-1: Parameters Collected in Experiments in Simulated Environment

Parameter	Units	Description
Test Number	1:17	The test number, identifies the number of terminals reporting for a test, see Table I-1
Report Number	>0	The number of this terminal report in this test
Terminal ID	String	The ID of the terminal that created the report
Session ID	String	The ID of the session for this
Report ID	String	The ID of this report
Generate Time	Time stamp	The time at which the terminal report was generated
Store Time	Time stamp	The time at which knowledge from the terminal report was stored to the semantic store
Intercept Time	Milli-seconds	The amount of time taken to lift knowledge from the terminal report into the ontology

M.2 Experiments in Test Bed Environment

Table M-2: Parameters Collected on Sessions

Service	Name	Units	Description
All	Session ID	String	The ID of the service session
All	Address	String	The IP address of the connection carrying this session
All	Service ID	String	The ID of the service: Video on Demand, Web Browsing, or File Transfer
All	User ID	String	The user running the session
All	Priority	1:10	The priority of the session, deduced by Aesop from the user ID and Service ID using a SWRL rule over the priority knowledge that appears in Table 6-5
All	Terminated	0:1	Flag indicating if the session has terminated or not
All	Throttled	0:1	Flag indicating if the session has been throttled or not
All	Throttle Limit Down	Bytes per second	The downstream throttling limit applied to the session
All	Throttle Limit Up	Bytes per second	The upstream throttling limit applied to the session
File Transfer	File Size	Bytes	Size of the file being transferred
File Transfer	URL	String	The URL of the remote location of the file being transferred
File Transfer	Host	String	The name of the remote host being used on the file transfer operation
File Transfer	Uploading	True or False	Flag indicating if a file transfer is an upload or download, used by Aesop to determine whether the upload or download throughput should be used for compliance analysis
Video on Demand	URL	String	The URL of the remote location of video being played
Video on Demand	Quality Level	Standard or High	Flag indicating if a Video on Demand session is standard quality of high quality, used by Aesop to which Video on Demand service expectation level should be used for compliance analysis
Web Browsing	First URL	String	The first URL used on this Web Browsing session

Table M-3: Parameters Collected on Session Metrics

Service	Name	Units	Description
All	Timestamp	Time	The timestamp for the metric, terminal reports were issued at 10s intervals
All	Service ID	String	The ID of the service: Video on Demand, Web Browsing, or File Transfer
All	Session ID	String	The ID of the service session
All	Compliance	%	The compliance with the session expectations as calculated by the Aesop analysis operation
All	Period	Milli-seconds	The amount of time that this metric covers
All	Duration	Milli-seconds	The amount of time the session has been running
File Transfer	Bytes Uploaded	Kilobytes	The upload throughput in this session in the 10s of time covered by this metric
File Transfer	Bytes Downloaded	Kilobytes	The download throughput in this session in the 10s of time covered by this metric
All	User ID	String	The user running the session
All	Connection ID	String	The connection ID of the connection on which this service is running
All	Report Type	String	The type of report from which metrics were read: Started, Periodic, or Terminated
Video on Demand	MOS	1:5	Mean Opinion Score of the quality of this Video on Demand session in the 10s of time covered by this metric
Video on Demand	Freezes	>= 0	Number of freezes in this Video on Demand session in the 10s of time covered by this metric
Video on Demand	Errors	>=0	Number of errors on this Video on Demand session in the 10s of time covered by this metric
Video on Demand	Warnings	>= 0	Number of warnings on this Video on Demand session in the 10s of time covered by this metric
Web Browsing	Requests	>=0	Total number of requests in this Web Browsing session in the 10s of time covered by this metric
Web Browsing	Requests Ongoing	>=0	Number of requests still ongoing in this Web Browsing session at the end of the 10s of time covered by this metric
Web Browsing	Requests finished	>=0	Number of requests that finished in this Web Browsing session during the 10s of time covered by this metric

Service	Name	Units	Description
Web Browsing	Cached Requests	>=0	Number of requests read from cache in this Web Browsing session in the 10s of time covered by this metric
Web Browsing	Aborted Requests	>=0	Number of requests that aborted in this Web Browsing session during the 10s of time covered by this metric
Web Browsing	Error Requests	>=0	Number of requests that terminated in an error state in this Web Browsing session during the 10s of time covered by this metric
Web Browsing	Average Request Load Time	Milli-seconds	Average request load time of requests handled by the web browsing session in the 10s of time covered by this metric
Web Browsing	Standard Deviation on Request Load Time	Milli-seconds	Standard deviation on request load time of requests handled by the web browsing session in the 10s of time covered by this metric
Web Browsing	Average Request Size	Bytes	Average request size of requests handled by the web browsing session in the 10s of time covered by this metric
Web Browsing	Standard Deviation on Request Size	Bytes	Standard deviation on request size of requests handled by the web browsing session in the 10s of time covered by this metric
Web Browsing	Last Status	String	The HTTP status of the last request completed in the web browsing session in the 10s of time covered by this metric

Table M-4: Parameters Collected on Throttling Actions

Parameter	Units	Description
Throttling Time	Timestamp	The time at which the throttling action was carried out
Host	String	The name of the host on which throttling was carried out
Address	String	The IP address of the host on which throttling was carried out
Applied	0:1	True if throttling was activated, false if throttling was released
Queue	String	The netfilter queue used on this throttling action
Throttle Limit Down	Bytes per second	The downstream throttling limit used on this throttling action
Throttle Limit Up	Bytes per second	The upstream throttling limit used on this throttling action
Priority	1:10	The priority value used on this throttling action

Table M-5: Parameters Collected on Connections

Parameter	Units	Description
Connection ID	String	The connection ID
Address	String	The IP address of the connection
Interface	String	The hardware interface name of the connection
MAC Address	String	The MAC address of the connection
Connectivity Service	String	The connectivity service of the service provider that is providing this connection
Subscriber Name	String	The name of the subscriber that has subscribed for service on this connection

Table M-6: Other Parameters Collected

Parameter	Units	Description
CPU	%	CPU usage, recorded every 10s
Memory	Kilobytes	Memory usage, recorded every 10s
Load Time	Milli-seconds	The length of time taken by Aesop to lift knowledge from a terminal report and store it in the Knowledge Base
Analysis Time	Milli-seconds	The length of time taken by Aesop to analyse the compliance of all sessions in updated snapshots, set every 10s
Optimization Time	Milli-seconds	The length of time taken by Aesop to optimize the compliance of the set of running sessions over a group of snapshot windows, set every 10s
Run Start Time	Timestamp	The start time of the experimental run
Run Stop Time	Timestamp	The stop time of the experimental run

Appendix N
Experimental Results

N.1 Results for Experiment 1

Table N-1: Terminal and Terminal Report Counts for tests of Experiment 1

Test	Terminals Active	Terminal Reports Generated
1	6	660
2	10	1100
3	20	2200
4	30	3300
5	40	4400
6	50	5500
7	60	6600
14	80	8800
8	100	11000
13	125	13750
11	150	16500
12	175	19250
9	200	22000
17	225	24750
16	250	27500
15	275	30250
10	300	33000

Table N-2: Statistical Analysis of Results for Experiment 1

Test	Generate to Store Time (ms)			Processing Time (ms)			Intercept Time (ms)		
	Average	Stddev	Error Margin	Average	Stddev	Error Margin	Average	Stddev	Error Margin
1	921	438	44	327	204	20	15	9	1
2	883	428	33	304	168	13	12	8	1
3	917	420	23	291	131	7	10	7	0
4	969	440	20	286	116	5	8	6	0
5	1048	535	21	286	140	5	7	5	0
6	1182	848	29	286	147	5	6	5	0
7	1304	730	23	290	122	4	6	5	0
14	1558	928	25	295	124	3	5	4	0
8	2037	1360	33	300	116	3	5	4	0
13	3045	2118	47	315	126	3	5	4	0
11	6285	4913	99	327	152	3	4	3	0
12	20194	9805	182	339	142	3	4	3	0
9	61519	37480	651	345	146	3	4	3	0
17	111973	71700	1174	351	145	2	4	3	0
16	163120	110557	1717	356	145	2	4	5	0
15	213693	145789	2159	359	139	2	4	4	0
10	268226	179277	2542	363	142	2	3	3	0

In table above, error margins were determined with a confidence level of 99%.

N.2 Results for Experiment 2

N.2.1 Results on Session Users and Priorities

Table N-3: Scenario 2 Sessions

Run	Service	User	Priority		Run	Service	User	Priority	
			Actual	Expected				Actual	Expected
1	FT	Joe	10	10	4	FT	Joe	10	10
1	VOD	Alex	2	2	4	VOD	Alex	2	2
1	Web	Cathy	3	3	4	Web	Cathy	3	3
1	Web	Sam	1	1	4	Web	Sam	1	1
2	FT	Joe	10	10	5	FT	Joe	10	10
2	VOD	Alex	2	2	5	VOD	Alex	2	2
2	Web	Cathy	3	3	5	Web	Cathy	3	3
2	Web	Sam	1	1	5	Web	Sam	1	1
3	FT	Joe	10	10	6	FT	Joe	10	10
3	VOD	Alex	2	2	6	VOD	Alex	2	2
3	Web	Cathy	3	3	6	Web	Cathy	3	3
3	Web	Sam	1	1	6	Web	Sam	1	1

Table N-4: Scenario 3 Sessions

Run	Service	User	Priority		Run	Service	User	Priority	
			Actual	Expected				Actual	Expected
1	VOD	Cathy	5	5	4	VOD	Cathy	5	5
1	VOD	Sam	2	2	4	VOD	Sam	2	2
1	Web	Joe	3	3	4	Web	Joe	3	3
1	Web	Alex	1	1	4	Web	Alex	1	1
2	VOD	Cathy	5	5	5	VOD	Cathy	5	5
2	VOD	Sam	2	2	5	VOD	Sam	2	2
2	Web	Joe	3	3	5	Web	Joe	3	3
2	Web	Alex	1	1	5	Web	Alex	1	1
3	VOD	Cathy	5	5	6	VOD	Cathy	5	5
3	VOD	Sam	2	2	6	VOD	Sam	2	2
3	Web	Joe	3	3	6	Web	Joe	3	3
3	Web	Alex	1	1	6	Web	Alex	1	1

N.2.2 Results on Accuracy of Aesop

Table N-5: Aesop Monitoring Accuracy

Scenario and Run	Terminal Reports	Missed	Loading Incorrect	Accuracy	
				Handling	Monitoring
S2 Run1	0	N/A	N/A	N/A	N/A
S2 Run2	897	0	0	100.00%	100.00%
S2 Run3	902	2	0	99.78%	100.00%
S2 Run4	902	0	0	100.00%	100.00%
S2 Run5	901	0	0	100.00%	100.00%
S2 Run6	901	0	0	100.00%	100.00%
S3 Run1	1046	0	0	100.00%	100.00%
S3 Run2	1046	0	0	100.00%	100.00%
S3 Run3	1046	1	0	99.90%	100.00%
S3 Run4	1046	0	0	100.00%	100.00%
S3 Run5	1044	0	0	100.00%	100.00%
S3 Run6	1046	0	0	100.00%	100.00%
Total	**10777**	**3**	**0**	**99.97%**	**100.00%**

Note: Terminal reports were not collected for Scenario 2 Run 1.

Table N-6: Aesop Analysis Accuracy

Scenario and Run	Snapshots	Not Analyzed	Analysis Incorrect	Accuracy	
				Processing	Analysis
S2 Run1	N/A	N/A	N/A	N/A	N/A
S2 Run2	897	0	0	100.00%	100.00%
S2 Run3	900	0	0	100.00%	100.00%
S2 Run4	902	4	0	99.56%	100.00%
S2 Run5	901	0	0	100.00%	100.00%
S2 Run6	901	5	0	99.45%	100.00%
S3 Run1	1046	7	0	99.33%	100.00%
S3 Run2	1046	0	0	100.00%	100.00%
S3 Run3	1045	4	0	99.62%	100.00%
S3 Run4	1046	3	0	99.71%	100.00%
S3 Run5	1044	0	0	100.00%	100.00%
S3 Run6	1046	4	0	99.62%	100.00%
Total	**10774**	**27**	**0**	**99.75%**	**100.00%**

N.2.3 Statistical Analysis of Aesop Performance Parameters

In the following tables, error margins were determined with a confidence level of 99%.

Table N-7: Statistical Analysis of Scenario 2 Monitoring Times

Min	Read-ings	Milliseconds			Min	Read-ings	Milliseconds		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
6	143	395.16	64.35	13.86	31	180	933.63	177.16	34.02
7	180	355.21	7.20	1.38	32	180	954.92	180.41	34.64
8	180	357.46	9.34	1.79	33	180	974.61	186.21	35.75
9	180	414.68	91.26	17.52	34	180	988.93	188.62	36.22
10	180	468.49	118.99	22.85	35	180	1000.66	189.66	36.41
11	180	538.09	132.34	25.41	36	180	1011.67	187.58	36.02
12	180	583.57	117.13	22.49	37	180	1037.79	196.69	37.77
13	180	619.57	110.72	21.26	38	180	1049.63	201.05	38.60
14	180	644.82	116.70	22.41	39	180	1055.49	196.65	37.76
15	180	670.48	122.37	23.50	40	180	1090.94	207.67	39.87
16	180	687.64	126.42	24.27	41	180	1106.74	213.29	40.95
17	180	711.84	133.99	25.73	42	180	1116.91	214.43	41.17
18	180	720.20	135.06	25.93	43	180	1143.16	224.00	43.01
19	180	729.49	135.68	26.05	44	180	1157.46	224.25	43.06
20	180	751.79	138.78	26.65	45	180	1105.55	272.98	52.41
21	180	773.66	144.29	27.70	46	180	1096.06	282.35	54.21
22	180	786.47	147.58	28.34	47	180	1011.48	266.54	51.18
23	180	805.76	150.66	28.93	48	180	1005.63	273.18	52.45
24	180	822.71	154.82	29.73	49	179	908.27	226.79	43.67
25	180	836.07	153.87	29.54	50	180	919.92	222.98	42.81
26	180	857.17	162.25	31.15	51	179	812.56	97.68	18.81
27	179	872.59	169.85	32.70	52	180	796.18	29.43	5.65
28	180	883.27	162.78	31.25	53	180	798.48	31.75	6.10
29	180	911.03	171.88	33.00	54	1	774.00	0	0
30	180	924.08	178.14	34.20					

Table N-8: Statistical Analysis of Scenario 2 Analysis Times

Min	Read-ings	Milliseconds			Min	Read-ings	Milliseconds		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
3	35	120.54	40.72	17.73	31	36	417.28	56.03	24.06
4	36	136.75	5.03	2.16	32	36	396.47	75.17	32.27
5	36	136.67	3.77	1.62	33	36	385.17	67.36	28.92
6	36	172.39	33.78	14.50	34	36	401.17	66.25	28.44
7	36	190.81	31.97	13.73	35	36	405.19	65.17	27.98
8	36	193.67	33.55	14.41	36	36	399.19	50.02	21.47
9	36	198.31	42.62	18.30	37	36	364.72	65.98	28.33
10	36	210.78	29.82	12.80	38	36	389.47	68.15	29.26
11	36	240.58	45.31	19.45	39	36	420.14	74.72	32.08
12	36	280.42	33.43	14.35	40	36	360.50	67.64	29.04
13	36	290.11	32.76	14.07	41	36	384.72	72.42	31.09
14	36	294.89	61.25	26.30	42	36	436.00	76.63	32.90
15	36	291.69	49.15	21.10	43	36	394.78	78.38	33.65
16	36	315.17	35.58	15.27	44	36	411.36	85.71	36.80
17	36	307.78	51.69	22.19	45	36	421.69	88.33	37.92
18	36	331.75	37.39	16.05	46	36	429.31	83.34	35.78
19	36	345.44	55.02	23.62	47	36	395.19	74.79	32.11
20	36	323.03	64.11	27.53	48	36	409.92	105.22	45.18
21	36	324.78	54.67	23.47	49	36	416.08	59.53	25.56
22	36	363.64	52.10	22.37	50	36	363.83	63.63	27.32
23	36	353.56	46.43	19.93	51	36	446.44	100.04	42.95
24	36	358.28	59.68	25.62	52	36	476.53	76.84	32.99
25	36	360.08	54.52	23.41	53	36	503.00	56.52	24.26
26	36	358.31	52.10	22.37	54	36	339.11	67.88	29.14
27	36	383.08	60.74	26.08	55	36	293.14	27.46	11.79
28	36	404.39	68.69	29.49	56	36	299.72	19.69	8.45
29	36	354.83	61.93	26.59	57	5	300.20	2.28	2.63
30	36	382.36	63.30	27.18					

Table N-9: Statistical Analysis of Scenario 2 CPU Usage

Min	Read-ings	CPU %			Min	Read-ings	CPU %		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
0	36	0.46	0.55	0.24	30	36	8.81	0.19	0.08
1	36	0.26	0.16	0.07	31	36	8.87	0.26	0.11
2	36	0.34	0.29	0.13	32	36	8.99	0.28	0.12
3	36	4.75	4.40	1.89	33	36	9.13	0.22	0.09
4	36	1.19	0.17	0.07	34	36	9.27	0.24	0.10
5	36	1.13	0.30	0.13	35	36	9.47	0.33	0.14
6	36	5.89	3.21	1.38	36	36	9.55	0.50	0.21
7	36	4.93	0.58	0.25	37	36	9.58	0.19	0.08
8	36	4.40	0.35	0.15	38	36	9.72	0.19	0.08
9	36	5.30	0.55	0.24	39	36	9.75	0.30	0.13
10	36	5.56	0.33	0.14	40	36	9.79	0.33	0.14
11	36	6.07	0.22	0.09	41	36	10.13	0.24	0.10
12	36	6.40	0.28	0.12	42	36	10.20	0.41	0.18
13	36	6.62	0.26	0.11	43	35	10.28	0.36	0.16
14	36	7.05	1.48	0.64	44	36	10.55	0.30	0.13
15	36	6.94	0.24	0.10	45	36	10.84	1.72	0.74
16	36	7.09	0.25	0.11	46	36	10.20	0.63	0.27
17	36	7.23	0.20	0.09	47	36	9.74	0.55	0.24
18	36	7.31	0.26	0.11	48	36	9.60	0.15	0.07
19	36	7.45	0.30	0.13	49	36	8.91	0.56	0.24
20	36	7.40	0.30	0.13	50	36	8.96	0.40	0.17
21	36	7.55	0.18	0.08	51	36	8.99	0.96	0.41
22	36	8.09	1.16	0.50	52	36	8.18	0.69	0.30
23	36	7.86	0.26	0.11	53	36	8.13	0.24	0.10
24	36	8.07	0.20	0.09	54	36	3.47	2.36	1.01
25	36	8.34	0.40	0.17	55	36	2.19	0.29	0.13
26	36	8.28	0.26	0.11	56	36	2.20	0.21	0.09
27	36	8.44	0.24	0.11	57	36	0.84	0.93	0.40
28	36	8.49	0.25	0.11	58	36	0.31	0.23	0.10
29	36	8.59	0.26	0.11	59	36	0.28	0.18	0.08

Table N-10: Statistical Analysis of Scenario 2 Memory Usage

Min	Read-ings	Memory GB			Min	Read-ings	Memory GB		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
0	36	1.85	0.17	0.07	30	36	3.51	0.37	0.16
1	36	1.85	0.17	0.07	31	36	3.55	0.23	0.10
2	36	1.85	0.17	0.07	32	36	3.46	0.15	0.06
3	36	2.17	0.21	0.09	33	36	3.53	0.23	0.10
4	36	2.27	0.17	0.07	34	36	3.53	0.33	0.14
5	36	2.27	0.17	0.07	35	36	3.59	0.26	0.11
6	36	2.43	0.24	0.10	36	36	3.55	0.30	0.13
7	36	2.52	0.32	0.14	37	36	3.75	0.27	0.12
8	36	2.53	0.33	0.14	38	36	4.09	0.47	0.20
9	36	2.63	0.33	0.14	39	36	4.12	0.38	0.16
10	36	2.70	0.30	0.13	40	36	3.95	0.35	0.15
11	36	2.82	0.26	0.11	41	36	4.08	0.37	0.16
12	36	2.92	0.23	0.10	42	36	4.14	0.35	0.15
13	36	2.84	0.22	0.10	43	35	4.20	0.27	0.12
14	36	2.80	0.26	0.11	44	36	4.06	0.29	0.12
15	36	2.88	0.28	0.12	45	36	3.94	0.26	0.11
16	36	2.93	0.19	0.08	46	36	4.08	0.30	0.13
17	36	2.90	0.20	0.09	47	36	3.96	0.30	0.13
18	36	3.02	0.18	0.08	48	36	4.14	0.43	0.18
19	36	2.98	0.19	0.08	49	36	4.31	0.29	0.13
20	36	2.99	0.26	0.11	50	36	4.24	0.31	0.13
21	36	3.04	0.26	0.11	51	36	4.47	0.25	0.11
22	36	3.17	0.25	0.11	52	36	4.61	0.18	0.08
23	36	3.18	0.19	0.08	53	36	4.67	0.16	0.07
24	36	3.07	0.16	0.07	54	36	4.47	0.15	0.06
25	36	3.15	0.25	0.11	55	36	4.47	0.15	0.07
26	36	3.17	0.33	0.14	56	36	4.47	0.15	0.07
27	36	3.36	0.25	0.11	57	36	2.27	0.89	0.38
28	36	3.48	0.30	0.13	58	36	1.88	0.16	0.07
29	36	3.44	0.33	0.14	59	36	1.88	0.16	0.07

Table N-11: Statistical Analysis of Scenario 3 Monitoring Times

Min	Read-ings	Milliseconds			Min	Read-ings	Milliseconds		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
4	143	395.66	61.05	13.15	31	180	970.90	179.53	34.47
5	180	357.17	14.58	2.80	32	180	983.13	180.55	34.67
6	180	410.80	100.18	19.23	33	180	1002.31	185.04	35.53
7	180	466.28	119.32	22.91	34	180	1020.66	190.44	36.56
8	180	520.14	119.91	23.02	35	180	1035.25	190.10	36.50
9	180	585.60	111.77	21.46	36	180	1045.10	191.54	36.78
10	180	616.76	117.42	22.54	37	180	1066.54	196.36	37.70
11	180	637.58	121.83	23.39	38	180	1087.40	204.82	39.33
12	180	655.80	121.72	23.37	39	179	1105.26	206.38	39.74
13	180	674.52	127.26	24.43	40	180	1122.41	210.87	40.49
14	180	698.55	131.03	25.16	41	180	1144.61	218.83	42.02
15	180	712.91	134.44	25.81	42	180	1149.98	212.43	40.79
16	180	729.37	136.26	26.16	43	180	1173.66	220.28	42.29
17	180	740.08	141.06	27.09	44	180	1189.31	221.10	42.45
18	180	759.27	147.20	28.26	45	180	1203.16	224.44	43.09
19	180	781.37	149.20	28.65	46	179	1218.24	227.22	43.75
20	180	796.52	153.67	29.51	47	180	1248.98	239.41	45.97
21	180	810.89	153.32	29.44	48	179	1165.93	268.86	51.77
22	180	826.44	157.59	30.26	49	180	1162.63	276.71	53.13
23	180	841.81	161.66	31.04	50	181	1089.41	285.62	54.69
24	180	850.39	154.45	29.66	51	180	1069.92	284.06	54.54
25	180	867.41	160.41	30.80	52	181	988.37	252.04	48.26
26	180	879.86	160.45	30.81	53	180	967.20	235.42	45.20
27	180	902.08	167.47	32.16	54	180	863.17	110.13	21.14
28	180	922.29	172.90	33.20	55	180	861.02	44.89	8.62
29	180	931.89	170.43	32.72	56	1	918.00	0	0
30	180	953.06	175.77	33.75					

Table N-12: Statistical Analysis of Scenario 3 Analysis Times

Min	Read-ings	Milliseconds			Min	Read-ings	Milliseconds		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
2	36	123.28	41.29	17.73	31	36	394.72	61.43	26.37
3	36	137.89	4.46	1.92	32	36	403.33	67.05	28.79
4	36	170.00	27.00	11.59	33	36	405.67	71.87	30.86
5	36	183.83	17.32	7.44	34	36	389.47	56.68	24.34
6	36	206.11	24.96	10.72	35	36	383.17	62.94	27.02
7	36	225.81	40.18	17.25	36	36	400.42	83.06	35.66
8	36	218.81	30.39	13.05	37	36	401.58	82.90	35.59
9	36	254.44	43.87	18.83	38	36	385.47	71.50	30.70
10	36	272.19	41.67	17.89	39	36	425.92	108.25	46.47
11	36	287.06	43.18	18.54	40	36	419.47	81.76	35.10
12	36	285.81	38.41	16.49	41	36	390.36	66.19	28.42
13	36	310.14	31.78	13.65	42	36	424.53	84.87	36.44
14	36	316.17	48.63	20.88	43	36	417.33	89.42	38.39
15	36	307.64	48.84	20.97	44	36	418.86	79.98	34.34
16	36	318.94	45.24	19.42	45	36	433.89	83.57	35.88
17	36	360.58	41.88	17.98	46	36	431.11	93.83	40.28
18	36	343.72	58.62	25.17	47	36	430.75	84.19	36.15
19	36	328.78	60.71	26.06	48	36	431.08	104.56	44.89
20	36	356.31	57.85	24.84	49	36	433.94	95.03	40.80
21	36	351.08	43.93	18.86	50	36	402.08	70.89	30.44
22	36	357.53	52.30	22.46	51	36	437.92	100.66	43.22
23	36	365.67	50.59	21.72	52	36	403.61	69.38	29.79
24	36	354.06	72.62	31.18	53	36	405.53	80.37	34.51
25	36	363.19	59.51	25.55	54	36	435.97	88.04	37.80
26	36	396.19	56.78	24.38	55	36	387.00	84.61	36.33
27	36	382.44	61.99	26.61	56	36	328.06	43.93	18.86
28	36	369.22	66.27	28.45	57	36	309.92	24.74	10.62
29	36	399.83	59.33	25.47	58	6	286.50	45.50	47.85
30	36	388.22	61.41	26.36					

Table N-13: Statistical Analysis of Scenario 3 CPU Usage

Min	Read-ings	CPU %			Min	Read-ings	Memory %		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
0	36	0.34	0.32	0.14	30	36	8.95	0.27	0.12
1	36	0.38	0.20	0.09	31	36	9.35	0.95	0.41
2	36	5.66	6.11	2.62	32	36	9.40	0.96	0.41
3	36	1.21	0.16	0.07	33	36	9.44	0.32	0.14
4	36	5.85	2.97	1.27	34	36	9.79	0.28	0.12
5	36	4.88	0.58	0.25	35	36	9.58	0.32	0.14
6	36	5.37	0.57	0.24	36	36	9.92	0.16	0.07
7	36	5.41	0.34	0.14	37	36	10.05	0.30	0.13
8	36	6.12	0.41	0.17	38	36	10.16	0.32	0.14
9	36	6.78	0.32	0.14	39	36	10.51	0.21	0.09
10	36	7.16	0.29	0.12	40	36	10.39	0.33	0.14
11	36	7.15	0.28	0.12	41	36	10.75	0.32	0.14
12	36	7.42	1.21	0.52	42	36	10.84	0.35	0.15
13	36	7.31	0.22	0.09	43	36	11.04	0.41	0.17
14	36	7.54	0.33	0.14	44	36	11.23	0.25	0.11
15	36	7.52	0.32	0.14	45	36	11.18	0.34	0.15
16	36	7.69	0.26	0.11	46	36	11.32	0.33	0.14
17	36	7.74	0.29	0.12	47	36	11.58	0.45	0.19
18	36	7.67	0.31	0.13	48	36	11.25	0.58	0.25
19	36	7.89	0.20	0.09	49	36	11.16	0.53	0.23
20	36	8.06	0.24	0.10	50	36	10.82	0.58	0.25
21	36	7.88	0.30	0.13	51	36	10.61	0.40	0.17
22	36	8.21	0.31	0.13	52	36	9.71	0.63	0.27
23	36	8.35	0.33	0.14	53	36	9.42	0.22	0.09
24	36	8.22	0.33	0.14	54	36	9.09	0.86	0.37
25	36	8.31	0.39	0.17	55	36	8.55	0.59	0.25
26	36	8.36	0.22	0.09	56	36	3.59	2.42	1.04
27	36	8.44	0.31	0.13	57	36	2.35	0.25	0.11
28	36	8.66	0.26	0.11	58	36	0.79	0.94	0.40
29	36	8.71	0.22	0.09	59	36	0.25	0.10	0.04

Table N-14: Statistical Analysis of Scenario 3 Memory Usage

Min	Read-ings	Memory GB			Min	Read-ings	Memory GB		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
0	36	2.14	0.04	0.02	30	36	3.72	0.26	0.11
1	36	2.14	0.04	0.02	31	36	3.69	0.26	0.11
2	36	2.50	0.13	0.06	32	36	3.85	0.27	0.12
3	36	2.57	0.02	0.01	33	36	3.83	0.33	0.14
4	36	2.72	0.10	0.04	34	36	3.83	0.38	0.16
5	36	2.77	0.08	0.04	35	36	4.13	0.56	0.24
6	36	2.77	0.05	0.02	36	36	4.24	0.49	0.21
7	36	2.84	0.11	0.05	37	36	4.37	0.34	0.15
8	36	2.97	0.17	0.07	38	36	4.20	0.37	0.16
9	36	2.96	0.14	0.06	39	36	4.25	0.28	0.12
10	36	2.96	0.08	0.04	40	36	4.42	0.27	0.11
11	36	2.93	0.06	0.03	41	36	4.44	0.33	0.14
12	36	2.98	0.07	0.03	42	36	4.58	0.28	0.12
13	36	3.09	0.17	0.07	43	36	4.55	0.22	0.09
14	36	3.07	0.15	0.07	44	36	4.44	0.28	0.12
15	36	3.12	0.16	0.07	45	36	4.48	0.23	0.10
16	36	3.29	0.17	0.07	46	36	4.42	0.25	0.11
17	36	3.37	0.20	0.09	47	36	4.44	0.32	0.14
18	36	3.34	0.16	0.07	48	36	4.52	0.28	0.12
19	36	3.32	0.12	0.05	49	36	4.58	0.28	0.12
20	36	3.44	0.28	0.12	50	36	4.57	0.35	0.15
21	36	3.36	0.26	0.11	51	36	4.74	0.22	0.09
22	36	3.34	0.21	0.09	52	36	4.60	0.27	0.12
23	36	3.38	0.21	0.09	53	36	4.72	0.27	0.12
24	36	3.41	0.34	0.14	54	36	4.61	0.26	0.11
25	36	3.57	0.26	0.11	55	36	4.59	0.25	0.11
26	36	3.66	0.17	0.07	56	36	4.66	0.26	0.11
27	36	3.57	0.14	0.06	57	36	4.69	0.26	0.11
28	36	3.51	0.15	0.07	58	36	2.54	0.86	0.37
29	36	3.77	0.23	0.10	59	36	2.16	0.04	0.02

N.2.4 Statistical Analysis of Session Compliance by Scenario and Type

In the following tables, error margins were determined with a confidence level of 99%.

Table N-15: Statistical Analysis of Scenario 2 File Transfer Session Metrics

Min	TRs	Compliance			Min	TRs	Compliance		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
9	24	100.00	0.00	0.00	28	36	33.30	17.95	7.71
10	36	93.48	12.68	5.45	29	36	33.04	24.92	10.70
11	36	66.33	21.41	9.19	30	36	47.53	26.24	11.27
12	36	38.83	15.93	6.84	31	36	31.96	16.88	7.25
13	36	34.47	12.80	5.50	32	36	27.56	14.93	6.41
14	36	38.26	16.33	7.01	33	36	28.43	14.04	6.03
15	36	34.29	11.33	4.86	34	36	26.92	13.03	5.59
16	36	31.89	18.56	7.97	35	36	20.46	15.54	6.67
17	36	36.11	10.69	4.59	36	36	28.47	16.06	6.89
18	36	30.64	14.43	6.19	37	36	22.59	12.84	5.51
19	36	28.29	14.70	6.31	38	36	31.01	12.10	5.20
20	36	28.33	12.32	5.29	39	36	32.90	17.30	7.43
21	36	23.98	11.79	5.06	40	36	43.18	27.93	11.99
22	36	27.92	10.85	4.66	41	36	37.37	18.70	8.03
23	36	24.61	12.11	5.20	42	36	31.08	18.39	7.90
24	36	27.66	12.07	5.18	43	36	30.99	16.12	6.92
25	36	34.12	12.99	5.58	44	36	30.03	15.51	6.66
26	36	33.56	20.29	8.71	45	18	33.55	12.08	7.34
27	36	26.99	11.98	5.14					

Table N-16: Statistical Analysis of Scenario 2 Video on Demand Session Metrics

Min	TRs	Compliance			Min	TRs	Compliance		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
11	24	49.40	44.20	23.24	31	36	84.60	31.77	13.64
12	36	55.43	44.23	18.99	32	36	83.58	31.98	13.73
13	36	75.25	36.97	15.87	33	36	90.37	24.88	10.68
14	36	65.37	42.59	18.29	34	36	93.61	18.99	8.15
15	36	73.09	39.01	16.75	35	36	92.28	22.90	9.83
16	36	65.62	45.39	19.49	36	36	96.20	16.92	7.26
17	36	78.46	35.81	15.37	37	36	79.66	35.62	15.29
18	35	85.75	29.51	12.85	38	36	85.40	32.82	14.09
19	36	92.25	24.18	10.38	39	36	85.31	31.44	13.50
20	35	90.76	27.12	11.81	40	36	80.86	37.01	15.89
21	36	90.86	23.56	10.11	41	36	82.59	37.31	16.02
22	36	96.27	17.16	7.37	42	36	91.14	24.90	10.69
23	36	82.28	35.78	15.36	43	36	81.05	36.94	15.86
24	36	84.51	33.51	14.39	44	36	64.29	42.52	18.25
25	36	85.31	31.98	13.73	45	36	87.65	26.93	11.56
26	36	73.40	39.98	17.16	46	36	96.08	17.84	7.66
27	36	79.38	36.56	15.70	47	36	96.27	17.21	7.39
28	35	58.76	47.39	20.63	48	36	100.00	0.00	0.00
29	36	69.78	43.06	18.49	49	17	100.00	0.00	0.00
30	36	85.03	33.31	14.30					

Table N-17: Statistical Analysis of Scenario 2 Social Networking Web Session Metrics

Min	TRs	Compliance			Min	TRs	Compliance		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
10	18	90.78	13.67	8.30	29	36	32.89	36.65	15.74
11	36	74.30	26.93	11.56	30	36	41.19	34.31	14.73
12	36	57.04	35.35	15.18	31	36	40.10	35.48	15.23
13	36	44.72	31.48	13.52	32	36	40.86	32.41	13.91
14	36	32.17	31.98	13.73	33	36	34.82	37.51	16.10
15	36	33.23	33.36	14.32	34	36	32.38	36.33	15.60
16	36	46.49	35.10	15.07	35	36	23.62	33.48	14.37
17	36	32.84	34.42	14.78	36	36	30.98	35.98	15.45
18	36	36.73	38.58	16.56	37	36	28.31	36.69	15.75
19	36	37.21	33.21	14.26	38	36	35.37	28.95	12.43
20	35	25.69	33.82	14.73	39	36	32.46	31.45	13.50
21	36	29.09	41.63	17.87	40	36	43.53	34.25	14.70
22	36	17.82	30.02	12.89	41	36	47.43	36.01	15.46
23	36	22.86	32.13	13.80	42	36	36.35	35.80	15.37
24	36	40.48	34.35	14.75	43	36	31.19	31.91	13.70
25	36	42.88	38.16	16.38	44	36	39.94	31.73	13.62
26	36	42.10	36.90	15.84	45	36	42.64	41.60	17.86
27	36	43.68	33.35	14.32	46	36	71.04	25.58	10.98
28	36	38.19	32.76	14.06	47	18	40.25	39.32	23.87

Table N-18: Statistical Analysis of Scenario 2 Work Web Session Metrics

Min	TRs	Compliance			Min	TRs	Compliance		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
12	15	51.54	23.87	15.87	32	36	32.20	21.96	9.43
13	34	28.38	13.41	5.92	33	36	26.96	11.76	5.05
14	36	27.84	16.11	6.92	34	36	28.15	18.42	7.91
15	36	25.51	11.77	5.05	35	36	22.17	14.85	6.37
16	36	27.01	13.86	5.95	36	36	33.85	21.90	9.40
17	36	23.68	13.93	5.98	37	36	28.68	20.63	8.86
18	36	22.38	15.39	6.61	38	36	29.64	23.70	10.18
19	36	24.05	13.56	5.82	39	36	26.51	15.66	6.72
20	36	20.53	13.95	5.99	40	36	34.09	18.51	7.95
21	36	19.14	9.11	3.91	41	36	30.86	14.75	6.33
22	36	25.58	11.73	5.03	42	36	33.30	21.45	9.21
23	36	22.51	10.10	4.34	43	36	29.55	16.79	7.21
24	36	25.41	11.17	4.79	44	36	31.27	21.54	9.25
25	36	28.00	11.42	4.90	45	36	43.40	26.29	11.29
26	36	35.00	20.95	8.99	46	36	43.33	26.86	11.53
27	36	34.42	18.59	7.98	47	36	57.28	32.14	13.80
28	36	33.87	21.13	9.07	48	36	81.95	18.42	7.91
29	36	22.02	12.85	5.52	49	36	91.73	15.62	6.71
30	36	33.39	21.72	9.32	50	36	96.91	16.72	7.18
31	36	33.53	19.61	8.42	51	17	99.90	0.42	0.26

Table N-19: Statistical Analysis of Scenario 3 Sam Video on Demand Session Metrics

Min	TRs	Compliance			Min	TRs	Compliance		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
6	22	100.00	0.00	0.00	28	36	47.64	26.88	11.54
7	36	99.86	0.83	0.36	29	36	71.20	31.20	13.40
8	36	99.58	1.75	0.75	30	36	82.57	29.11	12.50
9	36	99.72	1.41	0.60	31	36	62.75	44.18	18.97
10	36	100.00	0.00	0.00	32	36	58.56	42.35	18.18
11	36	99.91	0.56	0.24	33	36	88.24	29.54	12.68
12	36	100.00	0.00	0.00	34	36	97.69	13.61	5.84
13	36	97.89	7.20	3.09	35	36	56.64	41.08	17.64
14	36	100.00	0.00	0.00	36	36	99.63	2.22	0.95
15	36	82.92	25.87	11.11	37	36	99.58	2.50	1.07
16	36	96.94	13.13	5.64	38	36	99.86	0.61	0.26
17	36	97.04	16.67	7.16	39	36	100.00	0.00	0.00
18	36	99.95	0.28	0.12	40	36	100.00	0.00	0.00
19	36	100.00	0.00	0.00	41	36	100.00	0.00	0.00
20	36	95.00	20.36	8.74	42	36	97.22	16.67	7.16
21	36	71.85	39.85	17.11	43	36	99.81	1.11	0.48
22	36	63.98	44.36	19.04	44	36	100.00	0.00	0.00
23	36	58.82	46.40	19.92	45	36	99.40	3.09	1.33
24	36	45.32	39.84	17.11	46	36	100.00	0.00	0.00
25	36	7.64	15.41	6.62	47	36	99.81	1.11	0.48
26	36	46.16	39.47	16.94	48	18	100.00	0.00	0.00
27	36	26.32	31.66	13.59					

Table N-20: Statistical Analysis of Scenario 3 Cathy Video on Demand Session Metrics

Min	TRs	Compliance			Min	TRs	Compliance		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
7	24	99.20	1.86	0.98	29	36	86.37	19.86	8.53
8	36	97.87	5.05	2.17	30	36	86.11	18.26	7.84
9	36	97.55	6.65	2.85	31	36	63.75	46.05	19.77
10	36	99.17	4.23	1.81	32	36	47.01	44.31	19.02
11	36	99.26	3.12	1.34	33	36	79.70	37.88	16.26
12	36	99.72	1.41	0.60	34	36	93.94	23.31	10.01
13	36	97.08	8.29	3.56	35	36	73.84	32.73	14.05
14	36	100.00	0.00	0.00	36	36	99.95	0.28	0.12
15	36	87.55	21.67	9.30	37	36	100.00	0.00	0.00
16	36	97.75	9.58	4.11	38	36	99.58	2.50	1.07
17	36	97.22	16.67	7.16	39	36	99.31	2.91	1.25
18	36	99.58	1.75	0.75	40	36	99.91	0.56	0.24
19	36	99.95	0.28	0.12	41	36	99.68	1.25	0.54
20	36	97.22	16.67	7.16	42	36	96.81	16.74	7.19
21	36	67.06	43.27	18.58	43	36	99.63	2.22	0.95
22	36	63.29	45.02	19.33	44	36	100.00	0.00	0.00
23	36	64.58	44.82	19.24	45	36	99.58	1.84	0.79
24	36	47.64	43.80	18.81	46	36	100.00	0.00	0.00
25	36	38.10	24.99	10.73	47	36	98.98	3.52	1.51
26	36	60.81	38.33	16.46	48	36	99.54	2.51	1.08
27	36	54.26	27.98	12.01	49	36	94.44	23.23	9.97
28	36	81.00	18.32	7.86	50	18	99.91	0.39	0.24

Table N-21: Statistical Analysis of Scenario 3 Social Networking Web Session Metrics

Min	TRs	Compliance			Min	TRs	Compliance		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
8	18	100.00	0.00	0.00	31	36	31.71	36.67	15.74
9	36	100.00	0.00	0.00	32	36	47.97	41.55	17.84
10	36	100.00	0.00	0.00	33	36	74.04	18.63	8.00
11	36	100.00	0.00	0.00	34	36	86.55	9.89	4.24
12	36	90.18	13.97	6.00	35	36	82.23	21.35	9.17
13	36	92.92	8.37	3.59	36	36	93.12	8.84	3.79
14	36	78.57	28.05	12.04	37	36	95.77	5.55	2.38
15	36	48.22	32.43	13.92	38	36	97.75	3.64	1.56
16	36	67.09	29.75	12.77	39	36	100.00	0.00	0.00
17	36	54.93	29.50	12.67	40	36	99.47	3.19	1.37
18	36	67.11	23.17	9.95	41	36	65.16	25.02	10.74
19	36	99.97	0.17	0.07	42	36	92.35	16.84	7.23
20	36	92.83	6.52	2.80	43	36	74.33	22.08	9.48
21	36	88.92	15.24	6.54	44	36	87.79	17.92	7.70
22	36	53.63	31.10	13.35	45	36	96.04	10.68	4.59
23	36	43.14	34.14	14.66	46	36	100.00	0.00	0.00
24	36	40.20	37.11	15.93	47	36	100.00	0.00	0.00
25	36	34.47	42.44	18.22	48	36	100.00	0.00	0.00
26	36	30.62	41.20	17.69	49	36	94.44	23.23	9.97
27	36	29.38	38.53	16.54	50	36	96.47	4.66	2.00
28	36	24.40	36.58	15.70	51	36	97.62	3.93	1.69
29	36	31.04	40.98	17.60	52	18	96.17	7.88	4.79
30	36	37.98	37.59	16.14					

Table N-22: Statistical Analysis of Scenario 3 Work Web Session Metrics

Min	TRs	Compliance			Min	TRs	Compliance		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
9	18	100.00	0.00	0.00	32	36	43.50	30.97	13.30
10	36	99.08	5.53	2.37	33	36	75.85	24.21	10.39
11	36	99.96	0.27	0.11	34	36	84.45	19.29	8.28
12	36	98.01	6.22	2.67	35	36	49.08	33.22	14.26
13	36	92.01	19.43	8.34	36	35	84.87	20.44	8.90
14	36	87.92	19.74	8.48	37	36	96.89	8.20	3.52
15	36	65.91	37.27	16.00	38	36	94.87	13.02	5.59
16	36	72.38	31.04	13.32	39	36	100.00	0.00	0.00
17	36	92.60	19.05	8.18	40	36	96.98	9.90	4.25
18	36	81.28	19.88	8.53	41	36	98.86	5.96	2.56
19	36	83.81	20.54	8.82	42	36	91.57	16.49	7.08
20	36	83.16	20.25	8.69	43	36	98.94	6.35	2.72
21	36	64.61	31.11	13.36	44	36	96.95	11.77	5.05
22	36	44.90	25.17	10.81	45	36	94.24	13.88	5.96
23	36	51.17	28.56	12.26	46	36	98.41	6.37	2.73
24	36	30.18	26.77	11.49	47	36	100.00	0.00	0.00
25	36	11.43	22.31	9.58	48	36	100.00	0.00	0.00
26	36	10.64	17.60	7.55	49	36	94.44	23.23	9.97
27	36	15.55	30.93	13.28	50	36	100.00	0.00	0.00
28	36	20.41	33.80	14.51	51	36	100.00	0.00	0.00
29	36	8.47	11.51	4.94	52	36	100.00	0.00	0.00
30	36	27.43	26.13	11.22	53	36	100.00	0.00	0.00
31	36	39.75	25.37	10.89	54	18	94.44	23.57	14.31

N.3 Results for Experiment 3

N.3.1 Optimization Effectiveness and Statistical Analysis of Session Compliance

In the following tables, error margins were determined with a confidence level of 99%.

Table N-23: Optimization Effectiveness, Scenario 2

Minute	Compliance			Minute	Compliance		
	Before	After	Effectiveness		Before	After	Effectiveness
10	90.775	89.778	-1.098	31	33.534	73.73	119.866
11	61.848	67.701	9.464	32	32.203	85.689	166.09
12	54.669	69.369	26.889	33	26.961	76.377	183.287
13	51.812	58.139	12.211	34	28.146	66.354	135.749
14	46.604	64.155	37.66	35	22.169	72.542	227.223
15	25.509	75.05	194.21	36	33.847	76.627	126.392
16	27.011	76.355	182.681	37	28.678	78.746	174.587
17	23.68	88.164	272.314	38	29.639	81.095	173.609
18	22.376	81.046	262.201	39	26.508	83.337	214.384
19	24.05	79.144	229.081	40	34.091	83.372	144.557
20	20.533	83.639	307.339	41	56.725	62.29	9.81
21	19.141	68.011	255.316	42	33.299	73.565	120.923
22	25.581	77.964	204.773	43	29.554	74.781	153.032
23	22.509	77.957	246.337	44	31.269	79.156	153.145
24	25.405	74.254	192.281	45	43.402	80.316	85.051
25	28.002	82.357	194.111	46	43.327	78.664	81.559
26	34.996	90.337	158.135	47	57.284	81.98	43.112
27	34.421	77.015	123.744	48	81.954	80.456	-1.828
28	33.875	78.575	131.956	49	91.732	92.212	0.523
29	22.016	77.766	253.225	50	96.905	100	3.194
30	33.388	75.394	125.812	51	99.899	100	0.101

Table N-24: Optimization Effectiveness, Scenario 3

Minute	Compliance			Minute	Compliance		
	Before	After	Effectiveness		Before	After	Effectiveness
6	100	96.087	-3.913	31	39.755	77.798	95.694
7	99.531	98.015	-1.523	32	43.499	72.099	65.749
8	99.151	99.167	0.016	33	75.848	85.721	13.017
9	99.317	99.803	0.489	34	84.453	99.815	18.19
10	99.562	99.433	-0.13	35	49.075	63.946	30.303
11	99.781	99.468	-0.314	36	84.868	99.321	17.03
12	96.978	96.865	-0.117	37	98.238	84.837	-13.641
13	94.977	87.148	-8.243	38	97.364	99.213	1.899
14	88.831	91.054	2.503	39	100	88.734	-11.266
15	65.684	67.375	2.574	40	98.816	98.105	-0.72
16	84.661	90.671	7.099	41	88.009	90.473	2.8
17	94.817	98.354	3.73	42	93.716	91.126	-2.764
18	90.619	91.26	0.707	43	91.028	95.423	4.828
19	91.906	89.189	-2.956	44	94.913	94.991	0.082
20	89.079	90.964	2.116	45	96.56	92.629	-4.071
21	68.229	74.806	9.64	46	99.471	99.228	-0.244
22	54.443	58.071	6.664	47	99.699	99.638	-0.061
23	54.992	57.982	5.437	48	99.884	97.431	-2.456
24	30.182	46.316	53.456	49	94.444	90.463	-4.215
25	11.431	14.345	25.492	50	98.793	94.464	-4.382
26	10.644	18.193	70.923	51	98.811	97.947	-0.874
27	15.549	31.532	102.791	52	98.084	98.815	0.745
28	20.405	32.126	57.442	53	100	100	0
29	8.474	16.477	94.442	54	94.444	100	5.883
30	27.427	29.619	7.992				

Table N-25: Statistical Analysis of Scenario 2 File Transfer Session Metrics

Min	TRs	Compliance			Min	TRs	Compliance		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
9	22	100.00	0.00	0.00	28	36	0.98	2.23	0.96
10	36	90.96	16.02	6.88	29	36	3.10	7.99	3.43
11	36	54.23	35.06	15.05	30	36	1.87	2.76	1.18
12	36	9.73	15.97	6.86	31	36	2.21	3.31	1.42
13	36	1.82	2.05	0.88	32	36	2.20	3.19	1.37
14	36	2.45	2.81	1.21	33	32	1.81	3.98	1.81
15	36	1.73	2.77	1.19	34	30	1.87	2.72	1.28
16	36	2.35	2.74	1.18	35	36	2.50	5.33	2.29
17	36	1.71	2.19	0.94	36	36	1.89	5.05	2.17
18	36	2.86	4.20	1.80	37	36	2.74	6.19	2.66
19	36	2.09	1.21	0.52	38	36	2.07	5.12	2.20
20	36	1.30	2.10	0.90	39	36	2.20	4.28	1.84
21	36	1.45	3.98	1.71	40	36	2.59	4.70	2.02
22	36	2.73	4.97	2.13	41	36	1.80	3.61	1.55
23	36	1.98	2.09	0.90	42	36	2.78	5.76	2.47
24	36	2.21	5.39	2.31	43	36	2.22	3.63	1.56
25	36	1.88	2.22	0.95	44	27	1.76	3.35	1.66
26	36	2.23	2.98	1.28	45	12	1.48	2.52	1.87
27	36	2.27	2.81	1.21					

Table N-26: Statistical Analysis of Scenario 2 Video on Demand Session Metrics

Min	TRs	Compliance			Min	TRs	Compliance		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
11	23	58.07	37.68	20.24	31	36	1.76	6.08	2.61
12	36	58.73	43.75	18.79	32	36	7.62	24.73	10.62
13	33	59.76	46.29	20.76	33	36	22.04	40.64	17.45
14	36	55.86	45.86	19.69	34	36	28.64	44.60	19.15
15	36	39.63	47.56	20.42	35	36	22.25	42.15	18.10
16	36	22.22	42.16	18.10	36	36	11.76	30.95	13.29
17	36	16.67	37.80	16.23	37	36	1.54	7.45	3.20
18	36	16.67	37.80	16.23	38	36	0.00	0.00	0.00
19	36	16.67	37.80	16.23	39	36	7.19	23.48	10.08
20	36	16.67	37.80	16.23	40	36	13.95	29.76	12.78
21	36	19.44	40.14	17.23	41	36	47.62	48.79	20.95
22	36	23.83	42.27	18.15	42	36	38.77	46.97	20.17
23	36	15.59	35.61	15.29	43	35	20.60	40.34	17.57
24	36	19.44	40.14	17.23	44	30	15.89	36.34	17.09
25	36	19.48	38.65	16.59	45	30	13.22	34.29	16.13
26	36	21.11	37.52	16.11	46	30	20.00	40.68	19.13
27	36	17.65	37.62	16.15	47	30	9.07	25.78	12.12
28	36	20.22	39.11	16.79	48	30	15.37	35.10	16.51
29	36	9.44	28.48	12.23	49	14	44.60	50.19	34.55
30	36	6.98	23.74	10.19					

Table N-27: Statistical Analysis of Scenario 2 Social Networking Web Session Metrics

Min	TRs	Compliance			Min	TRs	Compliance		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
10	18	89.78	18.45	11.20	29	36	28.41	38.97	16.73
11	36	77.33	27.41	11.77	30	36	25.11	34.83	14.95
12	36	68.06	33.17	14.24	31	36	25.28	33.86	14.54
13	36	50.52	29.29	12.58	32	36	28.21	39.19	16.82
14	36	38.33	36.25	15.56	33	36	35.85	44.35	19.04
15	36	17.66	25.49	10.94	34	36	24.16	38.46	16.51
16	36	34.86	38.81	16.66	35	36	34.86	43.65	18.74
17	36	30.03	36.45	15.65	36	36	21.92	37.37	16.04
18	36	35.66	42.14	18.09	37	36	20.11	39.81	17.09
19	36	42.65	45.49	19.53	38	36	35.93	36.15	15.52
20	36	27.70	40.68	17.47	39	36	29.35	33.49	14.38
21	36	36.23	47.18	20.25	40	36	48.00	38.89	16.70
22	36	19.92	35.49	15.24	41	36	33.87	36.67	15.74
23	36	22.24	39.25	16.85	42	36	30.03	35.90	15.41
24	36	45.32	40.48	17.38	43	36	30.19	35.95	15.43
25	36	37.95	38.26	16.43	44	30	25.98	34.30	16.13
26	36	38.27	36.32	15.59	45	30	24.88	38.29	18.01
27	36	33.29	34.33	14.74	46	30	21.20	34.09	16.03
28	36	31.71	37.57	16.13	47	15	45.90	43.50	28.93

Table N-28: Statistical Analysis of Scenario 2 Work Web Session Metrics

Min	TRs	Compliance			Min	TRs	Compliance		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
12	18	81.31	19.62	11.91	32	36	85.69	19.74	8.48
13	36	56.51	29.15	12.52	33	36	76.38	22.98	9.87
14	36	72.45	28.27	12.14	34	36	66.35	26.31	11.29
15	36	75.05	23.50	10.09	35	36	72.54	20.84	8.95
16	36	76.36	22.18	9.52	36	36	76.63	20.22	8.68
17	36	88.16	19.04	8.17	37	36	78.75	22.09	9.48
18	36	81.05	22.84	9.81	38	36	81.09	21.61	9.28
19	36	79.14	22.63	9.72	39	36	83.34	21.46	9.21
20	36	83.64	22.21	9.53	40	36	83.37	22.85	9.81
21	36	68.01	27.79	11.93	41	36	76.96	23.44	10.06
22	36	77.96	22.82	9.80	42	36	73.57	20.55	8.82
23	36	77.96	23.07	9.91	43	36	74.78	23.84	10.24
24	36	74.25	26.93	11.56	44	31	79.16	25.56	11.83
25	36	82.36	19.35	8.31	45	30	80.32	22.26	10.47
26	36	90.34	13.77	5.91	46	30	78.66	23.87	11.23
27	36	77.01	21.82	9.37	47	30	81.98	20.98	9.87
28	36	78.58	24.05	10.32	48	30	80.46	22.26	10.47
29	36	77.77	24.99	10.73	49	30	92.21	17.55	8.26
30	36	75.39	26.93	11.56	50	30	100.00	0.00	0.00
31	36	73.73	23.67	10.16	51	15	100.00	0.00	0.00

Table N-29: Statistical Analysis of Scenario 3 Priority 2 Video on Demand Session Metrics

Min	TRs	Compliance			Min	TRs	Compliance		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
6	23	96.09	14.00	7.52	28	37	41.71	25.64	10.86
7	36	99.26	4.44	1.91	29	35	58.00	39.56	17.22
8	36	99.95	0.28	0.12	30	37	34.53	40.34	17.08
9	36	99.63	1.39	0.60	31	35	32.81	29.55	12.87
10	36	99.95	0.28	0.12	32	37	26.17	28.54	12.09
11	36	98.89	3.36	1.44	33	35	64.07	34.34	14.95
12	36	99.49	1.90	0.82	34	37	52.64	38.92	16.48
13	36	97.13	8.19	3.52	35	35	40.52	41.46	18.05
14	36	100.00	0.00	0.00	36	37	76.51	36.68	15.53
15	36	88.15	18.99	8.15	37	35	72.00	39.52	17.21
16	36	95.83	17.59	7.55	38	37	99.59	1.54	0.65
17	36	97.22	16.67	7.16	39	35	100.00	0.00	0.00
18	36	96.99	16.66	7.15	40	37	99.73	1.64	0.70
19	36	96.27	17.46	7.50	41	35	99.57	1.87	0.81
20	36	96.76	16.74	7.19	42	37	99.37	2.02	0.85
21	36	79.24	39.10	16.79	43	35	99.95	0.28	0.12
22	36	59.03	44.89	19.27	44	36	99.54	1.41	0.61
23	36	51.81	45.68	19.61	45	35	99.19	4.23	1.84
24	36	35.16	39.72	17.05	46	37	100.00	0.00	0.00
25	35	10.81	15.24	6.63	47	35	99.76	0.82	0.36
26	37	40.43	31.57	13.37	48	18	100.00	0.00	0.00
27	35	12.62	27.08	11.79					

Table N-30: Statistical Analysis of Scenario 3 Priority 5 Video on Demand Session Metrics

Min	TRs	Compliance			Min	TRs	Compliance		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
7	24	96.77	13.80	7.25	29	36	57.57	26.72	11.47
8	36	97.55	6.59	2.83	30	36	50.97	35.12	15.08
9	36	99.58	1.46	0.63	31	36	41.92	45.01	19.33
10	36	97.78	5.76	2.47	32	36	92.45	23.26	9.99
11	36	98.98	3.14	1.35	33	36	91.53	27.99	12.02
12	36	99.91	0.56	0.24	34	36	75.14	40.08	17.21
13	36	70.35	42.34	18.18	35	36	66.69	35.42	15.21
14	36	12.59	31.39	13.48	36	36	39.65	41.16	17.67
15	36	0.07	0.42	0.18	37	36	79.31	31.72	13.62
16	36	80.39	35.26	15.14	38	36	51.39	43.80	18.81
17	36	100.00	0.00	0.00	39	36	62.57	42.44	18.22
18	36	97.36	12.88	5.53	40	36	89.19	26.89	11.55
19	36	94.72	21.18	9.09	41	36	55.86	39.46	16.94
20	36	96.34	16.94	7.27	42	36	79.56	32.44	13.93
21	36	89.17	20.32	8.73	43	36	47.27	48.70	20.91
22	36	87.01	24.07	10.34	44	36	83.33	37.80	16.23
23	36	95.88	9.12	3.91	45	36	79.54	31.65	13.59
24	36	74.12	27.79	11.93	46	36	62.92	42.28	18.15
25	36	48.17	17.34	7.45	47	36	99.68	1.94	0.84
26	36	69.86	24.43	10.49	48	36	89.72	25.63	11.00
27	36	53.89	29.41	12.63	49	36	80.30	33.29	14.29
28	36	34.75	37.61	16.15	50	18	90.56	24.93	15.14

Table N-31: Statistical Analysis of Scenario 3 Social Networking Web Session Metrics

Min	TRs	Compliance			Min	TRs	Compliance		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
8	15	100.00	0.00	0.00	31	36	26.66	41.94	18.01
9	35	100.00	0.00	0.00	32	36	29.18	40.79	17.51
10	36	100.00	0.00	0.00	33	36	38.61	38.86	16.68
11	36	100.00	0.00	0.00	34	36	30.09	36.83	15.81
12	36	94.46	10.15	4.36	35	36	28.51	31.96	13.72
13	36	94.76	6.94	2.98	36	36	26.28	31.51	13.53
14	36	81.34	27.98	12.01	37	36	37.97	35.99	15.45
15	36	47.60	34.83	14.95	38	36	48.32	43.40	18.63
16	36	44.89	34.64	14.87	39	36	67.43	40.46	17.37
17	36	32.15	43.80	18.80	40	36	98.14	5.90	2.53
18	36	61.01	24.95	10.71	41	36	78.22	25.69	11.03
19	36	92.23	19.14	8.22	42	36	83.57	22.27	9.56
20	36	85.60	28.59	12.27	43	36	86.73	17.34	7.44
21	36	66.81	32.09	13.78	44	36	85.44	17.36	7.45
22	36	43.56	32.45	13.93	45	36	85.64	19.14	8.22
23	36	27.05	34.08	14.63	46	36	98.17	7.45	3.20
24	36	29.57	38.29	16.44	47	36	100.00	0.00	0.00
25	34	43.27	46.56	20.57	48	36	100.00	0.00	0.00
26	35	24.26	36.40	15.85	49	33	96.97	17.41	7.81
27	36	27.71	36.78	15.79	50	25	92.83	9.90	5.10
28	36	29.38	36.98	15.88	51	34	96.48	4.76	2.10
29	36	26.84	38.35	16.46	52	18	97.73	6.78	4.11
30	36	24.39	38.90	16.70					

Table N-32: Statistical Analysis of Scenario 3 Work Web Session Metrics

Min	TRs	Compliance			Min	TRs	Compliance		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
9	15	100.00	0.00	0.00	32	36	72.10	29.71	12.75
10	32	100.00	0.00	0.00	33	36	85.72	20.65	8.87
11	35	100.00	0.00	0.00	34	36	99.82	1.11	0.48
12	36	93.60	17.20	7.38	35	36	63.95	39.81	17.09
13	36	86.35	23.65	10.15	36	36	99.32	3.97	1.70
14	36	91.82	14.00	6.01	37	35	97.67	9.96	4.34
15	36	66.37	35.93	15.43	38	35	98.83	5.65	2.46
16	36	85.51	25.83	11.09	39	36	98.78	6.05	2.60
17	36	99.49	3.09	1.33	40	36	96.44	11.29	4.85
18	36	85.53	17.89	7.68	41	36	93.63	14.28	6.13
19	36	82.11	23.17	9.95	42	36	90.44	20.05	8.61
20	36	85.17	21.52	9.24	43	35	99.59	1.83	0.80
21	36	70.37	30.53	13.11	44	36	100.00	0.00	0.00
22	36	57.11	29.69	12.75	45	34	93.06	15.54	6.87
23	36	64.16	29.35	12.60	46	35	99.51	2.90	1.26
24	36	46.32	32.58	13.99	47	36	99.11	4.16	1.79
25	36	14.35	30.98	13.30	48	35	100.00	0.00	0.00
26	36	18.19	30.43	13.07	49	34	94.12	23.88	10.55
27	36	31.53	44.16	18.96	50	24	100.00	0.00	0.00
28	36	32.13	43.24	18.57	51	33	99.42	3.34	1.50
29	36	16.48	30.54	13.11	52	36	99.90	0.58	0.25
30	36	29.62	29.53	12.68	53	36	100.00	0.00	0.00
31	36	77.80	24.76	10.63	54	18	100.00	0.00	0.00

N.3.2 Throttling Actions and Consistency of Aesop Optimization

Table N-33: Throttling in Scenario 2 Sessions

Run	Min	Service	User	Priority	Action	Reason	Limit Bits/s	
							Down	Up
1	12	FT	Joe Han	10	Applied	Optimization	100000	10000
1	13	Web	Cathy Han	3	Applied	Optimization	500000	30000
1	14	VOD	Alex Han	2	Applied	Optimization	2000000	50000
1	22	VOD	Alex Han	2	Removed	Optimization	2000000	50000
1	23	VOD	Alex Han	2	Applied	Optimization	2000000	50000
1	41	VOD	Alex Han	2	Removed	Optimization	2000000	50000
1	43	VOD	Alex Han	2	Applied	Optimization	2000000	50000
1	45	FT	Joe Han	10	Removed	Terminated	100000	10000
1	47	Web	Cathy Han	3	Removed	Terminated	500000	30000
1	49	VOD	Alex Han	2	Removed	Terminated	2000000	50000
2	11	FT	Joe Han	10	Applied	Optimization	100000	10000
2	14	Web	Cathy Han	3	Applied	Optimization	500000	30000
2	15	VOD	Alex Han	2	Applied	Optimization	2000000	50000
2	25	VOD	Alex Han	2	Removed	Optimization	2000000	50000
2	26	VOD	Alex Han	2	Applied	Optimization	2000000	50000
2	43	VOD	Alex Han	2	Removed	Optimization	2000000	50000
2	45	VOD	Alex Han	2	Applied	Optimization	2000000	50000
2	45	FT	Joe Han	10	Removed	Terminated	100000	10000
2	47	Web	Cathy Han	3	Removed	Terminated	500000	30000
3	11	FT	Joe Han	10	Applied	Optimization	100000	10000
3	12	Web	Cathy Han	3	Applied	Optimization	500000	30000
3	14	VOD	Alex Han	2	Applied	Optimization	2000000	50000
3	32	VOD	Alex Han	2	Removed	Optimization	2000000	50000
3	35	VOD	Alex Han	2	Applied	Optimization	2000000	50000
3	40	VOD	Alex Han	2	Removed	Optimization	2000000	50000
3	41	VOD	Alex Han	2	Applied	Optimization	2000000	50000
3	47	Web	Cathy Han	3	Removed	Terminated	500000	30000
3	50	VOD	Alex Han	2	Removed	Terminated	2000000	50000
4	11	FT	Joe Han	10	Applied	Optimization	100000	10000
4	11	Web	Cathy Han	3	Applied	Optimization	500000	30000
4	20	Web	Cathy Han	3	Removed	Optimization	500000	30000
4	21	Web	Cathy Han	3	Applied	Optimization	500000	30000
4	22	VOD	Alex Han	2	Applied	Optimization	2000000	50000
4	39	VOD	Alex Han	2	Removed	Optimization	2000000	50000

Run	Min	Service	User	Priority	Action	Reason	Limit Bits/s	
							Down	Up
4	42	VOD	Alex Han	2	Applied	Optimization	2000000	50000
5	11	FT	Joe Han	10	Applied	Optimization	100000	10000
5	12	Web	Cathy Han	3	Applied	Optimization	500000	30000
5	16	VOD	Alex Han	2	Applied	Optimization	2000000	50000
5	23	VOD	Alex Han	2	Removed	Optimization	2000000	50000
5	25	VOD	Alex Han	2	Applied	Optimization	2000000	50000
5	40	VOD	Alex Han	2	Removed	Optimization	2000000	50000
5	42	VOD	Alex Han	2	Applied	Optimization	2000000	50000
5	45	FT	Joe Han	10	Removed	Terminated	100000	10000
5	47	Web	Cathy Han	3	Removed	Terminated	500000	30000
5	49	VOD	Alex Han	2	Removed	Terminated	2000000	50000
6	12	Web	Cathy Han	3	Applied	Optimization	500000	30000
6	12	FT	Joe Han	10	Applied	Optimization	100000	10000
6	13	VOD	Alex Han	2	Applied	Optimization	2000000	50000
6	25	VOD	Alex Han	2	Removed	Optimization	2000000	50000
6	28	VOD	Alex Han	2	Applied	Optimization	2000000	50000
6	32	VOD	Alex Han	2	Removed	Optimization	2000000	50000
6	36	VOD	Alex Han	2	Applied	Optimization	2000000	50000
6	44	FT	Joe Han	10	Removed	Terminated	100000	10000
6	46	VOD	Alex Han	2	Removed	Optimization	2000000	50000
6	47	Web	Cathy Han	3	Removed	Terminated	500000	30000

Note: Aesop removes throttling from throttled sessions if they terminate.

Table N-34: Throttling in Scenario 3 Sessions

Run	Minute	Service	User	Priority	Action	Reason	Limit Bits/s	
							Down	Up
1	14	VOD	Cathy Han	5	Applied	Optimization	1000000	35000
1	15	Web	Joe Han	3	Applied	Optimization	500000	20000
1	17	Web	Joe Han	3	Removed	Optimization	500000	20000
1	21	Web	Joe Han	3	Applied	Optimization	500000	20000
1	22	VOD	Sam Han	2	Applied	Optimization	2000000	50000
1	34	VOD	Sam Han	2	Removed	Optimization	2000000	50000
1	35	VOD	Sam Han	2	Applied	Optimization	2000000	50000
1	37	VOD	Sam Han	2	Removed	Optimization	2000000	50000
1	39	Web	Joe Han	3	Removed	Optimization	500000	20000
1	41	VOD	Cathy Han	5	Removed	Optimization	1000000	35000
1	42	VOD	Cathy Han	5	Applied	Optimization	1000000	35000
1	49	VOD	Cathy Han	5	Removed	Optimization	1000000	35000
2	13	VOD	Cathy Han	5	Applied	Optimization	1000000	35000
2	15	Web	Joe Han	3	Applied	Optimization	500000	20000
2	17	Web	Joe Han	3	Removed	Optimization	500000	20000
2	21	Web	Joe Han	3	Applied	Optimization	500000	20000
2	24	VOD	Sam Han	2	Applied	Optimization	2000000	50000
2	27	VOD	Sam Han	2	Removed	Optimization	2000000	50000
2	28	VOD	Sam Han	2	Applied	Optimization	2000000	50000
2	34	VOD	Sam Han	2	Removed	Optimization	2000000	50000
2	37	Web	Joe Han	3	Removed	Optimization	500000	20000
2	39	VOD	Cathy Han	5	Removed	Optimization	1000000	35000
2	41	VOD	Cathy Han	5	Applied	Optimization	1000000	35000
2	46	VOD	Cathy Han	5	Removed	Optimization	1000000	35000
3	12	VOD	Cathy Han	5	Applied	Optimization	1000000	35000
3	15	Web	Joe Han	3	Applied	Optimization	500000	20000
3	17	Web	Joe Han	3	Removed	Optimization	500000	20000
3	21	Web	Joe Han	3	Applied	Optimization	500000	20000
3	23	VOD	Sam Han	2	Applied	Optimization	2000000	50000
3	34	VOD	Sam Han	2	Removed	Optimization	2000000	50000
3	36	Web	Joe Han	3	Removed	Optimization	500000	20000
3	46	VOD	Cathy Han	5	Removed	Optimization	1000000	35000
4	13	Web	Joe Han	3	Applied	Optimization	500000	20000
4	13	VOD	Cathy Han	5	Applied	Optimization	1000000	35000
4	15	Web	Joe Han	3	Removed	Optimization	500000	20000

Run	Minute	Service	User	Priority	Action	Reason	Limit Bits/s	
							Down	Up
4	16	Web	Joe Han	3	Applied	Optimization	500000	20000
4	17	Web	Joe Han	3	Removed	Optimization	500000	20000
4	19	Web	Joe Han	3	Applied	Optimization	500000	20000
4	21	Web	Joe Han	3	Removed	Optimization	500000	20000
4	22	Web	Joe Han	3	Applied	Optimization	500000	20000
4	23	VOD	Sam Han	2	Applied	Optimization	2000000	50000
4	37	VOD	Sam Han	2	Removed	Optimization	2000000	50000
4	39	Web	Joe Han	3	Removed	Optimization	500000	20000
4	41	VOD	Cathy Han	5	Removed	Optimization	1000000	35000
4	44	VOD	Cathy Han	5	Applied	Optimization	1000000	35000
4	49	VOD	Cathy Han	5	Removed	Optimization	1000000	35000
5	13	VOD	Cathy Han	5	Applied	Optimization	1000000	35000
5	15	Web	Joe Han	3	Applied	Optimization	500000	20000
5	17	Web	Joe Han	3	Removed	Optimization	500000	20000
5	20	Web	Joe Han	3	Applied	Optimization	500000	20000
5	22	VOD	Sam Han	2	Applied	Optimization	2000000	50000
5	35	VOD	Sam Han	2	Removed	Optimization	2000000	50000
5	36	VOD	Cathy Han	5	Removed	Optimization	1000000	35000
5	38	Web	Joe Han	3	Removed	Optimization	500000	20000
5	41	VOD	Cathy Han	5	Applied	Optimization	1000000	35000
5	46	VOD	Cathy Han	5	Removed	Optimization	1000000	35000
6	13	VOD	Cathy Han	5	Applied	Optimization	1000000	35000
6	16	Web	Joe Han	3	Applied	Optimization	500000	20000
6	17	Web	Joe Han	3	Removed	Optimization	500000	20000
6	21	Web	Joe Han	3	Applied	Optimization	500000	20000
6	23	VOD	Sam Han	2	Applied	Optimization	2000000	50000
6	34	VOD	Sam Han	2	Removed	Optimization	2000000	50000
6	35	VOD	Sam Han	2	Applied	Optimization	2000000	50000
6	37	VOD	Sam Han	2	Removed	Optimization	2000000	50000
6	39	Web	Joe Han	3	Removed	Optimization	500000	20000
6	46	VOD	Cathy Han	5	Removed	Optimization	1000000	35000

Table N-35: Throttling Spreads during Scenario 2 Session Starts, Minute 0-22

Type	Priority	Runs	Min	Q1	Median	Q3	Max	Range
File Transfer	10	5	10.675	11.342	11.342	11.675	11.683	1.008
Web Social	3	5	12.425	12.508	12.517	13.008	14.100	1.675
Video on Demand	2	5	13.350	14.017	14.350	15.017	16.017	2.667

Table N-36: Throttling Spreads during Scenario 2 Session Terminations, Minute 44-60

Type	Priority	Runs	Min	Q1	Median	Q3	Max	Range
File Transfer	10	5	44.367	44.983	45.183	45.300	45.433	1.067
Web Social	3	5	47.100	47.108	47.108	47.108	47.167	0.067
Video on Demand	2	5	46.175	48.800	48.958	49.133	49.617	3.442

Table N-37: Throttling Spreads during Scenario 3 Impairment Increase, Minute 12-30

Type	Priority	Runs	Min	Q1	Median	Q3	Max	Range
Video on Demand	5	6	12.517	12.683	12.767	13.106	13.733	1.217
Web Social	3	6	13.517	15.392	15.517	15.560	15.692	2.175
Video on Demand	2	6	21.850	22.435	23.021	23.394	24.025	2.175

Table N-38: Throttling Spreads during Scenario 3 Impairment Decrease, Minute 30-60

Type	Priority	Runs	Min	Q1	Median	Q3	Max	Range
File Transfer	5	6	45.783	45.944	46.117	48.158	48.833	3.050
Web Social	3	6	36.275	36.992	38.225	38.415	38.783	2.508
Video on Demand	2	6	34.275	34.325	35.392	36.421	36.608	2.333

N.3.3 Results on Session Users and Priorities

Table N-39: Scenario 2 Sessions

Run	Service	User	Priority		Run	Service	User	Priority	
			Actual	Expected				Actual	Expected
1	FT	Joe	10	10	4	FT	Joe	10	10
1	VOD	Alex	2	2	4	VOD	Alex	2	2
1	Web	Cathy	3	3	4	Web	Cathy	3	3
1	Web	Sam	1	1	4	Web	Sam	1	1
2	FT	Joe	10	10	5	FT	Joe	10	10
2	VOD	Alex	2	2	5	VOD	Alex	2	2
2	Web	Cathy	3	3	5	Web	Cathy	3	3
2	Web	Sam	1	1	5	Web	Sam	1	1
3	FT	Joe	10	10	6	FT	Joe	10	10
3	VOD	Alex	2	2	6	VOD	Alex	2	2
3	Web	Cathy	3	3	6	Web	Cathy	3	3
3	Web	Sam	1	1	6	Web	Sam	1	1

Table N-40: Scenario 3 Sessions

Run	Service	User	Priority		Run	Service	User	Priority	
			Actual	Expected				Actual	Expected
1	VOD	Cathy	5	5	4	VOD	Cathy	5	5
1	VOD	Sam	2	2	4	VOD	Sam	2	2
1	Web	Joe	3	3	4	Web	Joe	3	3
1	Web	Alex	1	1	4	Web	Alex	1	1
2	VOD	Cathy	5	5	5	VOD	Cathy	5	5
2	VOD	Sam	2	2	5	VOD	Sam	2	2
2	Web	Joe	3	3	5	Web	Joe	3	3
2	Web	Alex	1	1	5	Web	Alex	1	1
3	VOD	Cathy	5	5	6	VOD	Cathy	5	5
3	VOD	Sam	2	2	6	VOD	Sam	2	2
3	Web	Joe	3	3	6	Web	Joe	3	3
3	Web	Alex	1	1	6	Web	Alex	1	1

N.3.4 Results on Accuracy of Aesop

Table N-41: Aesop Monitoring Accuracy

Scenario and Run	Terminal Reports	Missed	Loading Incorrect	Accuracy	
				Handling	Monitoring
S2 Run1	902	0	N/A	99.89%	100.00%
S2 Run2	901	1	0	98.89%	100.00%
S2 Run3	891	0	0	100.00%	100.00%
S2 Run4	814	19	0	97.67%	100.00%
S2 Run5	902	1	0	99.89%	100.00%
S2 Run6	895	3	0	99.66%	100.00%
S3 Run1	1036	0	0	100.00%	100.00%
S3 Run2	1046	24	0	97.71%	100.00%
S3 Run3	1046	0	0	100.00%	100.00%
S3 Run4	1046	19	0	98.18%	100.00%
S3 Run5	1046	3	0	99.71%	100.00%
S3 Run6	1046	2	0	99.81%	100.00%
Total	**11571**	**72**	**0**	**99.38%**	**100.00%**

Table N-42: Aesop Analysis Accuracy

Scenario and Run	Snapshots	Not Analyzed	Analysis Incorrect	Accuracy	
				Processing	Analysis
S2 Run1	902	1	0	99.89%	100.00%
S2 Run2	900	9	0	99.00%	100.00%
S2 Run3	891	11	0	98.77%	100.00%
S2 Run4	795	7	0	99.12%	100.00%
S2 Run5	901	8	0	99.11%	100.00%
S2 Run6	892	4	0	99.55%	100.00%
S3 Run1	1036	23	0	97.78%	100.00%
S3 Run2	1022	1	0	99.90%	100.00%
S3 Run3	1046	7	0	99.33%	100.00%
S3 Run4	1027	3	0	99.71%	100.00%
S3 Run5	1043	0	0	100.00%	100.00%
S3 Run6	1044	3	0	99.71%	100.00%
Total	**11499**	**77**	**0**	**99.33%**	**100.00%**

Table N-43: Aesop Optimization Accuracy

Scenario and Run	Throttling Actions	Correct Actions	Accuracy	Terminated Sessions	Restarted Sessions
S2 Run1	10	10	100	2	0
S2 Run2	10	10	100	2	0
S2 Run3	10	10	100	2	0
S2 Run4	8	8	100	1	0
S2 Run5	10	10	100	2	0
S2 Run6	10	10	100	1	0
S3 Run1	12	12	100	0	0
S3 Run2	12	12	100	0	0
S3 Run3	8	8	100	0	0
S3 Run4	14	14	100	0	0
S3 Run5	10	10	100	1	1
S3 Run6	10	10	100	0	0
Total	124	124	100	11	1

N.3.5 Statistical Analysis of Aesop Performance Parameters

In the following tables, error margins were determined with a confidence level of 99%.

Table N-44: Statistical Analysis of Scenario 2 Monitoring Times

Min	Read-ings	Milliseconds			Min	Read-ings	Milliseconds		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
6	146	408.14	69.91	14.90	31	157	1767.69	1746.48	359.05
7	180	369.22	39.68	7.62	32	209	2344.67	2187.72	389.82
8	180	369.83	36.70	7.05	33	171	1234.78	799.21	157.44
9	180	421.86	100.31	19.26	34	181	2454.09	2831.78	542.21
10	180	473.61	118.66	22.78	35	173	1686.72	1556.31	304.80
11	175	545.26	134.00	26.09	36	163	1573.60	1335.94	269.55
12	172	662.33	269.59	52.95	37	198	2366.35	2424.25	443.80
13	183	786.26	402.80	76.70	38	194	2190.06	1676.06	309.98
14	173	763.81	365.34	71.55	39	149	1360.89	972.47	205.22
15	170	939.09	686.16	135.57	40	165	1663.44	1235.35	247.74
16	156	952.94	665.46	137.25	41	213	2636.94	2729.82	481.83
17	160	1057.31	759.07	154.59	42	191	4025.40	5240.06	976.71
18	213	2732.57	2780.80	490.82	43	131	1555.11	1069.51	240.71
19	176	1476.44	1254.09	243.51	44	143	3324.52	3616.46	779.04
20	146	1093.75	825.41	175.97	45	161	2894.51	2508.87	509.34
21	184	1803.26	1857.09	352.67	46	133	1783.65	1429.88	319.39
22	197	2179.09	2067.97	379.54	47	164	1824.62	1489.87	299.69
23	203	2272.47	2352.26	425.29	48	185	3542.59	4086.76	774.00
24	179	1281.27	895.36	172.39	49	147	1034.81	659.82	140.19
25	157	1124.71	801.87	164.85	50	158	994.81	575.35	117.91
26	177	1450.45	1042.07	201.77	51	150	783.91	132.15	27.79
27	169	1653.52	1633.12	323.61	52	150	767.26	74.65	15.70
28	190	2458.24	2442.65	456.49	53	150	778.29	85.15	17.91
29	201	2331.52	2247.27	408.32	54	6	760.67	76.91	80.88
30	173	1823.33	1772.47	347.14					

Table N-45: Statistical Analysis of Scenario 2 Analysis Times

Min	Read-ings	Milliseconds			Min	Read-ings	Milliseconds		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
3	36	95.58	32.05	13.76	31	36	651.61	798.80	342.95
4	36	104.03	7.23	3.11	32	36	800.61	860.13	369.28
5	36	106.81	3.19	1.37	33	36	567.31	290.83	124.86
6	36	163.11	41.12	17.65	34	36	773.72	1158.56	497.41
7	36	183.11	19.41	8.33	35	36	647.78	669.36	287.38
8	36	185.19	10.23	4.39	36	36	606.83	446.39	191.65
9	36	189.86	17.66	7.58	37	36	868.86	1183.42	508.08
10	36	196.00	15.40	6.61	38	36	759.53	839.13	360.27
11	36	244.39	49.96	21.45	39	36	526.75	276.87	118.87
12	36	318.39	87.74	37.67	40	36	711.06	695.98	298.81
13	36	389.33	185.03	79.44	41	36	828.92	1076.12	462.02
14	36	437.86	157.15	67.47	42	34	639.79	610.98	269.92
15	36	478.31	204.96	88.00	43	31	1368.45	3714.62	1718.62
16	36	501.75	363.03	155.86	44	29	553.86	342.01	163.60
17	36	482.75	222.79	95.65	45	31	1302.68	2307.61	1067.65
18	36	921.92	1625.74	697.98	46	30	757.10	730.55	343.58
19	36	609.22	693.42	297.71	47	30	690.10	704.45	331.31
20	36	427.03	308.78	132.57	48	30	1065.63	1967.09	925.14
21	36	588.36	531.61	228.24	49	30	507.27	310.19	145.88
22	36	751.97	1148.38	493.04	50	30	524.07	265.86	125.04
23	36	688.61	647.94	278.18	51	30	459.90	70.69	33.24
24	36	598.56	393.45	168.92	52	30	435.67	74.35	34.97
25	36	522.17	420.01	180.32	53	30	459.10	92.81	43.65
26	36	576.67	413.53	177.54	54	30	425.20	58.58	27.55
27	36	616.61	524.74	225.29	55	30	415.60	18.49	8.70
28	36	831.22	1387.51	595.70	56	30	399.50	20.88	9.82
29	36	756.78	858.35	368.52	57	5	373.20	75.06	86.48
30	36	648.06	807.02	346.48					

Table N-46: Statistical Analysis of Scenario 2 Optimization Times

Min	Read-ings	Milliseconds			Min	Read-ings	Milliseconds		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
3	36	436.11	619.70	266.06	31	36	3116.42	588.04	252.47
4	36	116.94	4.67	2.00	32	36	3461.64	730.66	313.70
5	36	113.00	2.34	1.01	33	36	3209.36	510.95	219.37
6	36	237.58	90.29	38.77	34	36	3578.03	1551.70	666.20
7	36	346.69	30.84	13.24	35	36	3279.89	510.05	218.98
8	36	393.06	26.63	11.43	36	36	3205.08	544.24	233.66
9	36	472.72	42.33	18.17	37	36	3669.64	743.23	319.10
10	36	593.36	50.37	21.63	38	36	3270.92	459.49	197.28
11	36	780.61	79.33	34.06	39	36	3378.72	526.87	226.20
12	36	1004.50	112.33	48.23	40	36	3252.42	478.52	205.44
13	36	1323.06	174.52	74.93	41	35	3568.80	1471.21	640.60
14	36	1560.25	107.45	46.13	42	35	3717.71	2070.07	901.36
15	36	1779.47	198.08	85.04	43	31	3716.97	1196.33	553.50
16	36	2094.14	237.06	101.78	44	29	3334.72	742.93	355.38
17	36	2130.50	189.42	81.32	45	31	4124.19	1381.53	639.18
18	36	2955.69	1147.47	492.65	46	30	3242.00	650.50	305.94
19	36	3004.14	616.97	264.89	47	30	3188.73	735.02	345.69
20	36	2720.72	338.07	145.14	48	30	3935.00	2006.91	943.87
21	36	3074.42	646.02	277.36	49	30	3137.73	452.68	212.90
22	36	3198.97	756.36	324.73	50	30	2887.33	219.13	103.06
23	36	3318.08	696.86	299.19	51	30	2637.70	222.59	104.68
24	36	3034.81	439.04	188.49	52	30	2585.83	220.21	103.57
25	36	2859.72	259.75	111.52	53	30	2510.63	229.82	108.09
26	36	2858.50	303.61	130.35	54	31	2207.10	159.88	73.97
27	36	3205.36	861.50	369.87	55	30	1906.63	93.22	43.84
28	35	3326.29	1063.83	463.21	56	30	1723.07	67.45	31.72
29	37	3741.54	975.44	413.09	57	2	1487.50	72.83	132.66
30	36	3045.22	483.52	207.59					

Table N-47: Statistical Analysis of Scenario 2 CPU Usage

Min	Read-ings	CPU %			Min	Read-ings	CPU %		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
0	36	0.35	0.34	0.15	30	36	13.29	5.82	2.50
1	36	0.29	0.19	0.08	31	36	14.20	6.42	2.76
2	36	0.27	0.12	0.05	32	36	16.56	8.60	3.69
3	36	5.77	5.23	2.24	33	36	13.80	4.38	1.88
4	36	1.39	0.17	0.07	34	36	15.72	8.97	3.85
5	36	1.34	0.35	0.15	35	36	14.52	6.18	2.65
6	36	6.62	3.18	1.37	36	36	14.13	5.92	2.54
7	36	5.87	0.80	0.35	37	36	16.73	8.45	3.63
8	36	5.21	0.30	0.13	38	36	15.95	7.85	3.37
9	36	6.16	0.66	0.28	39	36	13.68	4.53	1.94
10	36	6.59	0.29	0.12	40	36	15.39	6.94	2.98
11	36	7.50	1.30	0.56	41	36	17.68	9.74	4.18
12	36	7.72	0.99	0.43	42	36	16.32	8.27	3.55
13	36	9.16	2.78	1.19	43	30	17.61	10.14	4.77
14	35	9.52	2.04	0.89	44	30	14.57	8.74	4.11
15	36	9.79	2.93	1.26	45	30	20.46	11.62	5.47
16	36	10.12	2.85	1.22	46	30	14.33	7.14	3.36
17	36	9.87	3.39	1.46	47	30	15.03	6.28	2.95
18	36	13.49	8.06	3.46	48	30	19.21	13.47	6.34
19	36	13.15	6.46	2.77	49	30	13.32	3.72	1.75
20	36	11.56	4.65	1.99	50	30	13.41	3.20	1.50
21	36	13.18	7.02	3.02	51	30	12.25	1.17	0.55
22	36	14.05	7.38	3.17	52	30	11.96	0.93	0.44
23	36	14.73	8.45	3.63	53	30	11.41	0.59	0.28
24	36	13.77	5.23	2.24	54	30	6.60	2.40	1.13
25	36	12.09	3.98	1.71	55	30	5.18	0.34	0.16
26	36	12.71	5.14	2.21	56	30	4.87	0.28	0.13
27	36	13.26	6.37	2.73	57	30	1.47	1.89	0.89
28	36	16.88	11.17	4.80	58	30	0.39	0.40	0.19
29	36	16.01	9.07	3.89	59	30	0.27	0.16	0.08

Table N-48: Statistical Analysis of Scenario 2 Memory Usage

Min	Read-ings	Memory GB			Min	Read-ings	Memory GB		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
0	36	2.70	0.06	0.03	30	36	5.58	0.09	0.04
1	36	2.70	0.06	0.03	31	36	5.59	0.09	0.04
2	36	2.70	0.06	0.03	32	36	5.59	0.09	0.04
3	36	3.05	0.13	0.06	33	36	5.61	0.11	0.05
4	36	3.11	0.06	0.03	34	36	5.60	0.10	0.04
5	36	3.10	0.06	0.03	35	36	5.60	0.09	0.04
6	36	3.25	0.12	0.05	36	36	5.60	0.08	0.04
7	36	3.40	0.12	0.05	37	36	5.60	0.08	0.04
8	36	3.54	0.15	0.06	38	36	5.60	0.08	0.04
9	36	3.64	0.20	0.09	39	36	5.61	0.08	0.03
10	36	3.83	0.23	0.10	40	36	5.61	0.08	0.03
11	36	4.05	0.37	0.16	41	36	5.62	0.09	0.04
12	36	4.15	0.27	0.12	42	36	5.63	0.09	0.04
13	36	4.27	0.22	0.10	43	30	5.68	0.09	0.04
14	35	5.00	0.22	0.10	44	30	5.69	0.09	0.04
15	36	5.29	0.11	0.05	45	30	5.69	0.08	0.04
16	36	5.34	0.08	0.04	46	30	5.69	0.08	0.04
17	36	5.36	0.07	0.03	47	30	5.70	0.08	0.04
18	36	5.40	0.07	0.03	48	30	5.72	0.08	0.04
19	36	5.44	0.08	0.03	49	30	5.73	0.08	0.04
20	36	5.46	0.07	0.03	50	30	5.75	0.09	0.04
21	36	5.47	0.07	0.03	51	30	5.76	0.09	0.04
22	36	5.48	0.08	0.04	52	30	5.79	0.10	0.05
23	36	5.51	0.06	0.03	53	30	5.82	0.09	0.04
24	36	5.53	0.05	0.02	54	30	5.66	0.08	0.04
25	36	5.53	0.05	0.02	55	30	5.64	0.08	0.04
26	36	5.54	0.05	0.02	56	30	5.64	0.10	0.04
27	36	5.54	0.05	0.02	57	30	3.17	1.00	0.47
28	36	5.55	0.06	0.02	58	30	2.73	0.08	0.04
29	36	5.58	0.08	0.04	59	30	2.73	0.08	0.04

Table N-49: Statistical Analysis of Scenario 3 Monitoring Times

Min	Read-ings	Milliseconds			Min	Read-ings	Milliseconds		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
4	147	409.04	73.93	15.71	31	175	1013.38	224.50	43.72
5	180	364.46	21.90	4.21	32	173	1036.16	228.40	44.73
6	180	412.66	95.11	18.26	33	191	1990.22	2612.61	486.97
7	180	463.43	110.33	21.18	34	180	1056.01	209.13	40.15
8	180	517.38	116.16	22.30	35	169	1055.12	199.25	39.48
9	180	586.01	117.23	22.51	36	188	1244.13	671.70	126.20
10	180	616.94	118.31	22.72	37	184	1086.62	217.96	41.39
11	179	634.70	112.80	21.72	38	180	1122.94	218.23	41.90
12	180	655.94	117.10	22.48	39	180	1143.06	223.97	43.00
13	175	678.57	119.90	23.35	40	180	1164.39	227.54	43.69
14	163	743.23	252.16	50.88	41	179	1218.33	295.04	56.81
15	201	979.86	706.94	128.45	42	177	1184.16	231.44	44.81
16	181	723.78	126.69	24.26	43	181	1211.61	243.58	46.64
17	180	749.66	135.65	26.05	44	172	1225.78	298.21	58.57
18	180	762.97	137.39	26.38	45	189	1430.26	885.12	165.85
19	180	778.62	137.91	26.48	46	180	1249.76	241.94	46.45
20	180	795.16	141.10	27.09	47	180	1257.56	253.06	48.59
21	180	821.48	157.50	30.24	48	180	1171.41	262.87	50.47
22	178	828.41	155.71	30.06	49	180	1168.11	271.26	52.08
23	182	856.26	172.35	32.91	50	180	1086.80	270.45	51.93
24	180	857.72	159.40	30.61	51	180	1084.91	281.52	54.05
25	180	879.05	158.89	30.51	52	181	1010.81	279.52	53.52
26	180	888.09	174.24	33.46	53	180	1005.58	296.26	56.88
27	173	903.29	170.67	33.42	54	180	885.51	158.20	30.38
28	187	967.58	230.76	43.47	55	180	870.85	89.68	17.22
29	180	974.08	207.02	39.75	56	5	809.80	47.27	54.46
30	177	971.24	185.69	35.95					

Table N-50: Statistical Analysis of Scenario 3 Analysis Times

Min	Read-ings	Milliseconds			Min	Read-ings	Milliseconds		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
2	36	96.61	33.15	14.23	31	36	467.25	138.40	59.42
3	36	106.42	7.63	3.28	32	36	521.22	143.56	61.64
4	36	163.56	39.44	16.93	33	36	635.44	819.23	351.72
5	36	195.58	26.62	11.43	34	36	473.42	141.11	60.58
6	36	215.39	29.67	12.74	35	36	491.64	122.14	52.44
7	36	232.75	32.09	13.78	36	36	551.94	242.04	103.92
8	36	230.00	34.21	14.69	37	36	502.11	159.52	68.49
9	36	256.83	39.93	17.14	38	36	471.08	128.56	55.19
10	36	294.11	42.51	18.25	39	36	499.97	138.51	59.47
11	36	335.17	35.49	15.24	40	36	450.53	96.41	41.39
12	36	319.67	43.98	18.88	41	36	470.22	155.97	66.96
13	36	338.94	77.02	33.07	42	36	478.11	181.60	77.97
14	36	409.75	150.95	64.81	43	36	447.25	115.88	49.75
15	36	482.17	420.59	180.57	44	36	446.47	89.98	38.63
16	36	423.53	62.75	26.94	45	36	495.31	296.72	127.39
17	36	418.56	74.28	31.89	46	36	453.25	94.88	40.73
18	36	358.06	61.84	26.55	47	36	445.25	85.24	36.60
19	36	395.06	67.52	28.99	48	36	454.75	109.37	46.96
20	36	412.28	77.16	33.13	49	36	472.19	133.70	57.40
21	36	433.06	101.05	43.39	50	36	480.58	99.29	42.63
22	36	440.97	86.54	37.15	51	36	473.11	112.63	48.36
23	36	410.50	109.10	46.84	52	36	484.64	93.89	40.31
24	36	432.86	80.23	34.45	53	36	480.61	135.89	58.34
25	36	407.00	63.03	27.06	54	36	467.25	105.91	45.47
26	36	397.44	49.95	21.44	55	36	450.22	103.06	44.25
27	36	412.44	72.29	31.04	56	36	421.31	66.94	28.74
28	36	494.28	189.84	81.51	57	36	401.25	51.15	21.96
29	36	479.44	128.59	55.21	58	5	377.20	76.97	88.67
30	36	406.31	54.76	23.51					

Table N-51: Statistical Analysis of Scenario 2 Optimization Times

Min	Read-ings	Milliseconds			Min	Read-ings	Milliseconds		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
2	36	445.25	641.68	275.50	31	36	3137.44	132.70	56.97
3	36	117.97	4.38	1.88	32	36	3165.72	186.90	80.24
4	36	238.64	80.54	34.58	33	36	3476.75	1384.11	594.25
5	36	340.83	22.64	9.72	34	36	3483.64	510.16	219.03
6	36	411.58	33.85	14.53	35	36	3220.86	147.29	63.24
7	36	524.36	59.42	25.51	36	36	3340.36	306.55	131.61
8	36	673.03	56.20	24.13	37	36	3336.11	178.05	76.44
9	36	856.56	73.34	31.49	38	36	3368.00	150.73	64.71
10	36	1107.67	79.17	33.99	39	36	3376.06	163.31	70.11
11	36	1343.25	86.29	37.05	40	36	3428.08	163.35	70.13
12	35	1560.71	79.19	34.48	41	36	3461.06	162.87	69.92
13	36	1798.72	84.57	36.31	42	36	3505.53	144.73	62.14
14	36	2024.47	166.08	71.30	43	36	3549.58	162.09	69.59
15	36	2437.97	273.75	117.53	44	36	3481.92	180.15	77.35
16	36	2396.22	82.10	35.25	45	36	3550.25	220.11	94.50
17	36	2531.03	98.42	42.25	46	36	3525.44	148.07	63.57
18	36	2659.08	83.69	35.93	47	36	3491.78	115.84	49.73
19	36	2767.75	87.54	37.58	48	36	3381.94	114.16	49.01
20	36	2838.44	115.43	49.56	49	36	3398.50	139.87	60.05
21	36	2890.17	124.66	53.52	50	36	3257.83	128.84	55.31
22	36	2926.33	136.65	58.67	51	36	3265.75	153.14	65.75
23	36	2985.00	121.35	52.10	52	36	3059.81	133.78	57.44
24	36	3022.72	113.20	48.60	53	36	3015.94	114.21	49.03
25	36	3020.42	103.83	44.58	54	36	2852.81	108.98	46.79
26	36	3019.47	98.47	42.28	55	36	2777.39	128.84	55.31
27	36	3043.94	95.03	40.80	56	36	2507.11	141.56	60.78
28	36	3095.22	152.59	65.51	57	36	2201.64	104.42	44.83
29	36	3108.00	136.33	58.53	58	5	2010.80	51.35	59.16
30	36	3133.97	91.92	39.46					

Table N-52: Statistical Analysis of Scenario 3 CPU Usage

Min	Read-ings	CPU %			Min	Read-ings	CPU %		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
0	36	0.36	0.28	0.12	30	36	12.95	0.90	0.38
1	36	0.28	0.16	0.07	31	36	14.27	3.60	1.55
2	36	5.78	6.36	2.73	32	36	13.94	2.16	0.93
3	36	1.48	0.20	0.09	33	36	14.91	7.10	3.05
4	36	6.60	3.51	1.51	34	36	15.14	3.40	1.46
5	36	5.60	0.50	0.21	35	36	14.02	1.12	0.48
6	36	6.12	0.53	0.23	36	36	15.14	2.79	1.20
7	36	6.37	0.32	0.14	37	36	15.31	1.75	0.75
8	36	7.26	0.44	0.19	38	36	15.50	2.17	0.93
9	36	8.07	0.39	0.17	39	36	16.48	2.87	1.23
10	36	8.64	0.35	0.15	40	36	16.45	2.33	1.00
11	36	9.13	0.33	0.14	41	36	16.43	3.45	1.48
12	36	9.38	0.27	0.12	42	36	17.06	2.92	1.25
13	36	9.89	0.40	0.17	43	36	16.82	1.62	0.70
14	36	10.66	2.77	1.19	44	36	16.63	2.61	1.12
15	36	12.92	3.56	1.53	45	36	18.59	5.36	2.30
16	36	11.59	2.13	0.92	46	36	16.83	1.26	0.54
17	36	11.70	1.50	0.64	47	36	16.83	1.20	0.51
18	36	12.16	1.55	0.66	48	36	16.23	1.10	0.47
19	36	12.86	2.15	0.92	49	36	17.96	5.01	2.15
20	36	13.38	3.67	1.58	50	36	16.32	3.19	1.37
21	36	12.64	1.22	0.52	51	36	15.75	1.56	0.67
22	36	13.45	2.44	1.05	52	36	14.34	1.34	0.58
23	36	12.45	0.75	0.32	53	36	13.78	0.64	0.27
24	36	13.21	2.08	0.89	54	36	13.40	0.91	0.39
25	36	12.75	1.42	0.61	55	36	12.62	0.61	0.26
26	36	12.73	0.46	0.20	56	36	8.01	3.59	1.54
27	36	12.65	0.55	0.24	57	36	6.01	1.73	0.74
28	36	13.24	1.49	0.64	58	36	1.81	2.39	1.03
29	36	12.99	1.32	0.57	59	36	0.27	0.12	0.05

Table N-53: Statistical Analysis of Scenario 3 Memory Usage

Min	Read-ings	Memory GB			Min	Read-ings	Memory GB		
		Average	Stddev	Error Margin			Average	Stddev	Error Margin
0	36	2.92	0.05	0.02	30	36	5.62	0.12	0.05
1	36	2.92	0.05	0.02	31	36	5.64	0.10	0.04
2	36	3.26	0.15	0.06	32	36	5.68	0.07	0.03
3	36	3.33	0.05	0.02	33	36	5.70	0.08	0.03
4	36	3.48	0.12	0.05	34	36	5.72	0.09	0.04
5	36	3.64	0.06	0.03	35	36	5.73	0.10	0.04
6	36	3.62	0.06	0.02	36	36	5.74	0.10	0.04
7	36	3.67	0.10	0.04	37	36	5.75	0.10	0.04
8	36	3.92	0.14	0.06	38	36	5.75	0.11	0.05
9	36	4.23	0.20	0.09	39	36	5.76	0.11	0.05
10	36	4.30	0.24	0.10	40	36	5.77	0.12	0.05
11	36	4.39	0.20	0.09	41	36	5.78	0.13	0.05
12	36	4.86	0.35	0.15	42	36	5.79	0.13	0.06
13	36	5.42	0.17	0.07	43	36	5.79	0.13	0.05
14	36	5.51	0.13	0.06	44	36	5.80	0.13	0.06
15	36	5.52	0.07	0.03	45	36	5.80	0.13	0.06
16	36	5.58	0.07	0.03	46	36	5.80	0.13	0.06
17	36	5.59	0.07	0.03	47	36	5.82	0.14	0.06
18	36	5.61	0.08	0.03	48	36	5.82	0.14	0.06
19	36	5.63	0.08	0.04	49	36	5.84	0.13	0.06
20	36	5.64	0.09	0.04	50	36	5.87	0.14	0.06
21	36	5.66	0.09	0.04	51	36	5.88	0.15	0.06
22	36	5.67	0.09	0.04	52	36	5.88	0.13	0.06
23	36	5.67	0.09	0.04	53	36	5.88	0.12	0.05
24	36	5.67	0.09	0.04	54	36	5.89	0.12	0.05
25	36	5.67	0.09	0.04	55	36	5.88	0.11	0.05
26	36	5.67	0.08	0.04	56	36	5.71	0.11	0.05
27	36	5.68	0.08	0.03	57	36	5.72	0.12	0.05
28	36	5.68	0.09	0.04	58	36	3.37	0.94	0.41
29	36	5.62	0.12	0.05	59	36	2.95	0.05	0.02