

A Multi-Faceted Model of Trust that is Personalisable and Specialisable

A thesis submitted to the
University of Dublin, Trinity College
for the degree of
Doctor of Philosophy

Karl Quinn
Knowledge and Data Engineering Group,
School of Computer Science & Statistics,
Trinity College,
Dublin

Submitted September 2006

Declaration

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

Karl Quinn
September 2006

Permission to lend or copy

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

Karl Quinn
September 2006

ACKNOWLEDGEMENTS

Firstly, I would like to thank Vincent Wade for the opportunity given to me to undertake a PhD in KDEG and under his guidance. In addition, I have learnt much more from Vinny than the ability to research and develop over the past three years, for which I am very grateful and will use into the future.

Declan O’Sullivan has been the foundation of my supervision for the past three years. Declan’s supervision has been the key to the steady progress, focused research, and success of my work over the PhD timeframe. I believe that setting my research standards to Declan’s benchmark standard will greatly enhance all my future success in whatever role I partake.

Dave Lewis has had a massive impact on the direction and success of my PhD research. Dave’s way of thinking has been crucial to the core research work of my PhD and I have personally benefited, and will benefit in the future, from it.

Most of my time in Trinity has been spent in KDEG. Outside of research I have had enormous influence from members of KDEG, post-graduate or staff member, in many aspects of life including research, lecturing, supervision, social, and sports.

In my personal life I would like to extend my thanks and appreciation to my close friends of many years; Barra, Adrian, Nick, Tony (Linda & Saga), Trevor, and Terry.

My family has had the most important impact on my life and I would like to offer a very special thanks to both my brothers, Gerard and Daniel. To my Mum and Dad, Adrienne and Thomas, who put a lifetime of effort and love into my upbringing and who always supported me both financially and emotionally I offer my gratitude for all your love, advice, and support.

ABSTRACT

Trust is a term that is open to a wide range of subjective interpretations, and it has therefore been argued that “trust is a fashionable but overloaded term with lots of intertwined meanings” [Gollmann, 2005]. To date, many varied synonyms for trust have been used to describe trust, which has led to a wide range of definitions for trust. With this wide and varied range of synonyms and definitions for trust it has come to pass that there is no real consensus as to the meaning of trust. Current state of the art in the area of trust management tends to use a single synonym, or definition, in their use of trust. For example, eBay uses a reputation based feedback system. Such a single synonym approach can only provide a generic, non-personalised trust management solution.

This thesis proposes a multi-faceted model of trust that is personalisable and specialisable. A multi-faceted approach can be used to provide a personalised model of trust that has the ability to capture an individual’s subjective view of trust and, at the same time, also capture the wide variety of subjective views of trust that are exhibited by individuals over a large and broad population. Such personalisation is currently not found within trust management research in computer science. Personalisation of this type within trust is a means to enhance trust management by providing a tailored and bespoke model of trust. The model of trust is also specialisable towards multiple application domains in order to reflect a domain’s classes, properties, relationships, and attributes. In this way trust management is not only personalised to the user but is it also specialised to the application domain.

To evaluate and validate this approach to modelling trust, several experiments were conducted and detailed analysis of the results is presented. In addition, a trust management service, called *myTrust*, has been implemented and combined with an advanced policy based management system to illustrate dynamic and flexible trust management across several diverse application domains.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	iii
ABSTRACT.....	iv
TABLE OF FIGURES.....	xi
ABBREVIATIONS	xv
1 INTRODUCTION.....	1
1.1 Motivation.....	1
1.2 Research Question.....	3
1.3 Objective and Goals	3
1.4 Technical Approach	4
1.5 Contribution	5
1.6 Overview of Thesis	7
2 STATE OF THE ART	8
2.1 Introduction.....	8
2.2 Literature Review: Definitions, Ideas, and Views of Trust.....	8
2.2.1 Dictionary	8
2.2.2 Psychology	9
2.2.3 Sociology	10
2.2.4 Computer Science.....	11
2.3 Survey of Trust Management Systems	14
2.3.1 Survey Categorisation and Comparison Framework	14
2.3.1.1 <i>Survey Categorisation</i>	14
2.3.1.2 <i>Comparison Framework</i>	15
2.3.2 Credential Based Trust Management Systems	17
2.3.2.1 <i>X.509 and PGP</i>	17
2.3.2.2 <i>PolicyMaker and KeyNote</i>	18
2.3.2.3 <i>REFEREE</i>	19
2.3.2.4 <i>IBM Trust Establishment Framework</i>	20
2.3.2.5 <i>Other Notable Credential Based Approaches</i>	21
2.3.2.6 <i>Overall Analysis of Credential based Trust Management</i>	21
2.3.3 Internal Trust Management Systems	22
2.3.3.1 <i>SULTAN</i>	23
2.3.3.2 <i>OpenPrivacy and Sierra</i>	25
2.3.3.3 <i>TRELLIS</i>	27
2.3.3.4 <i>Fidelis</i>	29
2.3.3.5 <i>FOAF Extended; TrustBot and TrustMail</i>	31
2.3.3.6 <i>Other Notable Internal Approaches</i>	32
2.3.3.7 <i>Overall Analysis of Internal Trust Management</i>	33

2.3.4	Online Community Trust Management Systems	34
2.3.4.1	<i>Advogato</i>	34
2.3.4.2	<i>eBay and Amazon</i>	35
2.3.4.3	<i>Slashdot and Epinions</i>	36
2.3.4.4	<i>Overall Analysis of Online Community Trust Management</i>	38
2.3.5	Online Social Network Trust Management Systems	38
2.3.5.1	<i>FilmTrust</i>	38
2.3.5.2	<i>Analysis of Social Network Trust Management</i>	39
2.3.6	Comparison Framework	40
2.3.6.1	<i>Model of Trust</i>	41
2.3.6.2	<i>Trust Annotation</i>	42
2.3.6.3	<i>Trust Calculation</i>	42
2.3.6.4	<i>Policy</i>	42
2.3.6.5	<i>Trust Architecture</i>	43
2.3.6.6	<i>Trust Representation</i>	43
2.3.6.7	<i>Factor Combinations</i>	43
2.4	Summary	44
3	DESIGN	45
3.1	Introduction	45
3.2	Influences from the State of the Art	46
3.2.1	Influences on Design of Model of Trust	46
3.2.2	Influences on Design of <i>myTrust</i>	48
3.3	Overall Framework	50
3.3.1	General Operation	50
3.4	The Multi-faceted Model of Trust that is Personalisable and Specialisable	52
3.4.1	Upper Ontology	53
3.4.2	Trust Meta-model	56
3.4.3	Personalised Model Design	60
3.4.4	Domain Specific Model Design	65
3.4.4.1	<i>General Engineering Process</i>	67
3.4.4.2	<i>Evidence Based - Web Service Specialisation</i>	68
3.4.4.3	<i>Opinion Based - Instant Messaging Specialisation</i>	70
3.5	Supporting Framework	72
3.5.1	Trust Data Annotation	72
3.5.2	Trust Calculation Algorithm	72
3.5.2.1	<i>Stages of Trust Calculation Algorithm</i>	73
3.5.3	Trust Policy	73
3.5.3.1	<i>myTrust and Policy</i>	74
3.6	Summary	74

4	IMPLEMENTATION	75
4.1	Introduction.....	75
4.2	myTrust Architecture	75
4.2.1	Architecture Overview.....	77
4.2.2	myTrust Application Layer	78
4.2.2.1	Interface Options.....	79
4.2.3	myTrust Service Layer.....	80
4.2.3.1	Web Services Layer.....	80
4.2.3.2	Servlets Layer.....	80
4.2.4	myTrust Business Layer.....	81
4.2.4.1	Trust Calculation Engine	81
4.2.4.2	Recommendation Engine.....	91
4.2.5	myTrust Persistence Layer	93
4.2.5.1	Trust Repository.....	93
4.2.5.2	User Repository.....	94
4.2.5.3	Personalise Repository.....	95
4.2.5.4	Policy Repository	96
4.3	Supporting Technologies	98
4.3.1	Model Representation using OWL	98
4.3.2	Model Storage using OWL and MySQL	98
4.3.3	Model Reasoning using Jena	99
4.3.4	Model Querying using RDQL	99
4.3.5	Model Access using JBoss, EJB's, Servlets, and Web Services.....	100
4.4	Summary	100
5	EVALUATION	101
5.1	Introduction.....	101
5.1.1	Evaluating the Multi-faceted, Personalisable Model of Trust	101
5.1.2	Evaluating the Accuracy of Recommendations.....	102
5.1.3	Evaluating Trust Concept Clarity	102
5.2	Experiment One- Necessity for a Multi-faceted, Personalisable Model of Trust	103
5.2.1	Experiment Goals	104
5.2.2	Hypotheses	104
5.2.3	Experiment Overview	104
5.2.4	Results	106
5.2.5	Key Findings	113
5.2.5.1	Trust Concepts: Usefulness and Concrete or Abstract Categorisation.....	113
5.2.5.2	Meta-Model: Different Strength Relationships	115
5.2.5.3	Necessity for Personalisation.....	116
5.2.5.4	Trust Model Alteration as Risk Increases	117

5.3	Experiment Two- Accuracy of Model of Trust.....	118
5.3.1	Experiment Goals	118
5.3.2	Hypotheses	118
5.3.3	Experiment Overview.....	118
5.3.4	Results	121
5.3.4.1	<i>Seven Day Follow up Survey</i>	123
5.3.4.2	<i>Fourteen Day Follow up Survey</i>	124
5.3.5	Key Findings	124
5.4	Experiment Three- Accuracy of Model of Trust with Additional Information ...	129
5.4.1	Experiment Goals	129
5.4.2	Hypotheses	129
5.4.3	Experiment Overview.....	130
5.4.4	Results	132
5.4.5	Key Findings	134
5.5	Experiment Four- Abstract and Concrete Concepts.....	142
5.5.1	Experiment Goals	142
5.5.2	Hypotheses	142
5.5.3	Experiment Overview.....	142
5.5.4	Results	143
5.5.5	Key Findings	144
5.6	Personalised Model Generation; Experiment One in Comparison with Experiment Two and Three	146
5.7	Summary	151
6	TRIALS.....	153
6.1	Introduction.....	153
6.2	Background Information for Trials One, Two, and Three	154
6.2.1	Community Based Policy Management (CBPM).....	154
6.2.2	PUDECAS Ubiquitous Computing Simulator.....	155
6.2.3	Enhanced Instant Messaging (IM) and Content Based Networking (CBN).....	157
6.3	Overview and Comparison of Trials One, Two, and Three	159
6.4	Trial One - Trustworthy Service Selection	160
6.4.1	Outline of Trial One	160
6.4.2	<i>Architecture and Mechanisms</i>	161
6.4.2.1	<i>Trust Annotation Mechanism</i>	163
6.4.2.2	<i>Trustworthy Service Selection Policy Specification</i>	164
6.4.2.3	<i>Personalisation Mechanism</i>	165
6.4.2.4	<i>Trust Calculation Algorithm and Example</i>	165
6.4.3	Results and Conclusions.....	167
6.5	Trial Two - Access Control.....	168

6.5.1	Outline of Trial Two.....	168
6.5.2	Architecture and Mechanisms	171
6.5.2.1	<i>Trust Annotation Mechanism</i>	172
6.5.2.2	<i>Access Control Policy Specification</i>	173
6.5.2.3	<i>Personalisation Mechanism</i>	174
6.5.2.4	<i>Trust Calculation Algorithm and Example Operation</i>	174
6.5.3	Common Data Sets for Trial Two and Trial Three.....	176
6.5.4	Results and Conclusions.....	177
6.6	Trial Three - Instant Messaging.....	178
6.6.1	Outline of Trial Three.....	178
6.6.2	Architecture and Mechanisms	179
6.6.2.1	<i>Trust Annotation Mechanism</i>	181
6.6.2.2	<i>Instant Messaging Policy Specification</i>	181
6.6.2.3	<i>Personalisation Mechanism</i>	183
6.6.2.4	<i>Trust Calculation Algorithm and Example Operation</i>	183
6.6.3	Results and Conclusions.....	185
6.7	Comparisons of myTrust to Related Work	186
6.8	Summary	189
7	CONCLUSIONS	190
7.1	Objectives and Achievements.....	190
7.2	Contribution	196
7.3	Future Work.....	201
7.4	Final Remarks	203
	BIBLIOGRAPHY	205
	GLOSSARY.....	214
	APPENDICES	216
	APPENDIX I – Trust Ontology Documents.....	216
	Upper Ontology	216
	Meta-Model	219
	Personalised Model.....	221
	Specialised Models	225
	<i>Web Services Domain Specific Model</i>	225
	<i>Instant Messaging Domain Specific Model</i>	253
	APPENDIX II – Research Experiments	256
	Experiment Data Sets.....	256
	Experiment One; Necessity of Personalisable, Multi-Faceted Approach	256
	Experiment Two; Accuracy Survey.....	264
	Experiment Three; Accuracy Survey with Additional Information	286
	Experiment Four; Clarity Survey	320

APPENDIX III – Implementation Code, Trial Data, and Sundry	330
Implementation Code.....	330
<i>Jena MySQL to OWL Converter for Personalised Model of Trust</i>	330
<i>HITS Algorithm for Generating Personalised Models of Trust</i>	341
Trial Data	351
Sundry.....	352
<i>HITS Algorithm</i>	352

TABLE OF FIGURES

Figure 2-1 SULTAN Trust Statement Example	23
Figure 2-2 Structure of a TRELIS <i>unit</i>	27
Figure 2-3 Example of a TRELIS <i>unit</i>	28
Figure 2-4 Example Fidelis Trust Instance	30
Figure 2-5 Example of FOAF Extension	31
Figure 2-6 Comparison Framework Chart.....	40
Figure 3-1 Overall Framework	50
Figure 3-2 Upper Ontology.....	53
Figure 3-3 Upper Ontology Protégé View.....	55
Figure 3-4 Upper Ontology OWL snippet (Reputation Only).....	55
Figure 3-5 Meta-model	56
Figure 3-6 Meta-model Protégé View	59
Figure 3-7 Meta-model OWL snippet.....	59
Figure 3-8 Personalised Model of Trust	60
Figure 3-9 Personalisation Data and HITS Results	62
Figure 3-10 Personalised Model Protégé View	64
Figure 3-11 Instance of a Personalised Model of Trust in Protégé.....	64
Figure 3-12 Partial Instance of a Personalised Model of Trust OWL snippet.....	65
Figure 3-13 Web Service Specific Model (Reliability Only)	69
Figure 3-14 Web Services Domain Model Protégé View	70
Figure 3-15 Instant Messaging Specific Model.....	71
Figure 3-16 Instant Messaging Domain Model Protégé View	71
Figure 4-1 Implemented <i>myTrust</i> Architecture.....	76
Figure 4-2 <i>myTrust</i> Framework and Implementation Relationship.....	78
Figure 4-3 Partial calculateTrustSourceDestination WSDL File.....	80
Figure 4-4 UML Sequence Diagram for trustManager.createUser	82
Figure 4-5 UML Sequence Diagram for trustManager.createUserDestinationTrust...82	
Figure 4-6 UML Sequence Diagram for trustManager.getTrustByUserSource	83
Figure 4-7 UML Sequence Diagram for trustManager.getTrustByUserDestination...83	
Figure 4-8 UML Sequence Diagram for trustManager.getTrustByUserSourceDestination	84

Figure 4-9 UML Sequence Diagram for trustManager.editTrust	85
Figure 4-10 UML Sequence Diagram for trustSearch.getUserSource	86
Figure 4-11 UML Sequence Diagram for trustSearch.getUserDestination	87
Figure 4-12 UML Sequence Diagram for trustSearch.get1degrees	87
Figure 4-13 UML Sequence Diagram for trustSearch.get2degrees	88
Figure 4-14 UML Sequence Diagram for trustCalculate.calculateTrust	89
Figure 4-15 UML Sequence Diagram for personaliseManager.getRankByUserSourceConcept	90
Figure 4-16 UML Sequence Diagram for personaliseManager.getWeightByUserSourceConcept	91
Figure 4-17 UML Sequence Diagram for policyManager.create	92
Figure 4-18 UML Sequence Diagram for policyManager.getPolicyByUserSourceEvent	92
Figure 4-19 trustDB Database Schema	93
Figure 4-20 Example trustData Instance	94
Figure 4-21 userDB Database Schema	94
Figure 4-22 Example userData Instance	95
Figure 4-23 personaliseDB Database Schema	95
Figure 4-24 Example personaliseData Instance	96
Figure 4-25 policyDB Database Schema	96
Figure 4-26 Example policyData Instance	97
Figure 5-1 Participant Age Demographics	106
Figure 5-2 Participant Sex Demographics	106
Figure 5-3 Online Purchase History	106
Figure 5-4 Likert Scales for all Concepts for Scenario One (\$10)	107
Figure 5-5 Likert Scales for all Concepts for Scenario Two (\$100)	108
Figure 5-6 Likert Scales for all Concepts for Scenario Three (\$1000)	109
Figure 5-7 Number One Ranked Trust Concepts by Subjects in \$1000 Scenario	110
Figure 5-8 Abstract Concept Table	111
Figure 5-9 Concept A Influenced By Concept B	111
Figure 5-10 Percentage of Influencing Concepts	111
Figure 5-11 Maximum Percentage of Subjects with Matching Rank, Influence, and Likert Scales	112
Figure 5-12 Likert Scales for all Concepts for All Scenarios	113

Figure 5-13 Trust Meta-Model	115
Figure 5-14 Concepts Influencing <i>concrete</i> Concepts.....	115
Figure 5-15 Concepts Influencing <i>abstract</i> Concepts	116
Figure 5-16 Total Recommendation Accuracy in Experiment one	121
Figure 5-17 Test Subject Community Result Set	123
Figure 5-18 Accuracy of Recommendations for Likert Scales.....	124
Figure 5-19 Accuracy of Recommendations vs. Increasing Trust Requirements	125
Figure 5-20 Number of Trust Recommendations at Required Trust Levels	125
Figure 5-21 Accuracy of Recommendations across Trust and Risk Levels	127
Figure 5-22 'Ask the Audience' Results for <i>family</i> and <i>pencil</i>	130
Figure 5-23 'Provide Guarantees' Results for <i>work colleague</i> and <i>mobile phone</i>	131
Figure 5-24 Total Recommendation Accuracy.....	133
Figure 5-25 Accuracy of Recommendations for Likert Scales.....	134
Figure 5-26 Accuracy of Recommendations vs. Increasing Trust Requirements	134
Figure 5-27 Number of Trust Recommendations at Required Trust Levels	135
Figure 5-28 Accuracy of Recommendations vs. Increasing Trust Requirements	136
Figure 5-29 Selection Points for Additional Information,	138
Figure 5-30 Accuracy of Recommendations across Trust and Risk Levels	139
Figure 5-31 Gains & Losses across Trust and Risk Levels	140
Figure 5-32 Clarity of Trust Concepts	143
Figure 5-33 Concept Rankings via Direct Questioning.....	146
Figure 5-34 Concept Rankings via Personalised Model of Trust.....	146
Figure 5-35 Aggregate Rank for <i>faith</i> via HITS algorithm	147
Figure 5-36 Aggregate Rank for <i>belief</i> via HITS algorithm.....	148
Figure 5-37 Aggregate Rank for <i>reliability</i> via HITS algorithm.....	149
Figure 5-38 Aggregate Rank for <i>credibility</i> via HITS algorithm	149
Figure 5-39 Aggregate Rank for <i>reputation</i> via HITS algorithm.....	150
Figure 6-1 Real World Lloyd Building Digital Photograph	156
Figure 6-2 PUDECAS Simulator Lloyd Building Screenshot.....	156
Figure 6-3 Enhanced Instant Messaging Client.....	158
Figure 6-4 Trust Systems Comparison Chart.....	159
Figure 6-5 Trust Policies for Trial One.....	160
Figure 6-6 User <i>reliability</i> Trust Data for Web Service A, B, and C	161
Figure 6-7 Expected Outcomes for Trial One.....	161

Figure 6-8 Calculation Methodology.....	162
Figure 6-9 Web Service B Protégé based Annotation	163
Figure 6-10 Example Trust Policy for Selection of a Web Services	164
Figure 6-11 User <i>reliability</i> Trust Data for Web Service A, B, and C	165
Figure 6-12 User Trust Concept Rankings	166
Figure 6-13 Actual Results for Trial One	167
Figure 6-14 Trust Policies.....	168
Figure 6-15 Expected Policy Decisions.....	169
Figure 6-16 Overall Architectural for Trial Two.....	171
Figure 6-17 Trust Annotation in Trial Two	172
Figure 6-18 Resource Tree for Lloyd Building	173
Figure 6-19 Policy for 'Small Office within office 111'	173
Figure 6-20 Personalisation Mechanism for Trial Two.....	174
Figure 6-21 HITS Ranked Concepts for User 'kdeg 1'	175
Figure 6-22 Sample Trust Data for Community Member 'kdeg 2'	175
Figure 6-23 Trust Values for Trial One and Two	176
Figure 6-24 Expected vs. Actual Actions	177
Figure 6-25 Location Information Policy	178
Figure 6-26 Expected Policy Decisions.....	179
Figure 6-27 Overall Architectural for Trial Three.....	180
Figure 6-28 Trust Annotation Mechanism in Enhanced IM Application.....	181
Figure 6-29 Policy Specification in Enhanced IM Application.....	182
Figure 6-30 Advanced Policy Specification in Enhanced IM Application.....	182
Figure 6-31 Example Trust Data.....	183
Figure 6-32 Example Rank and Weight Data for 'kdeg 1'	183
Figure 6-33 Band Values for 'kdeg 2'	184
Figure 6-34 Actual vs. Expected Actions	185
Figure 6-35 Comparison Framework Chart with <i>myTrust</i>	186

ABBREVIATIONS

CBN	Context Based Networking
CBPM	Community Based Policy Management
EJB	Enterprise Java Bean
FOAF	Friend-of-a-Friend
HITS	Hypertext Induced Topic Selection
IM	Instant Messaging
PGP	Pretty Good Privacy
PHP	PHP: Hypertext Pre-processor
OWL	Web Ontology Language
OWL-DL	Web Ontology Language – Description Logic
PBNM	Policy Based Network Management
RBAC	Role Based Access Control
RDF	Resource Description Framework
RDQL	RDF Data Query Language
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
WSDL	Web Service Definition Language
XML	Extensible Mark-up Language
XML RPC	Extensible Mark-up Language Remote Procedure Call
XMPP	Extensible Messaging and Presence Protocol

1 INTRODUCTION

1.1 Motivation

Trust is an important factor in real world human society and increasingly so in Internet environments. The Internet reflects many aspects of human society including communication, interaction, and commerce. For example, the online auction house eBay [eBay] enables community members to provide feedback for transactions so that trust can be built in order to provide confidence in future transactions. Trust is a term that is described using many synonyms for trust, or trust concepts. Trust is defined and used in different ways across a broad range of research areas including sociology, physiology, commerce, computer science. In computer science the trust concepts used to describe trust include “reputation” [Golbeck & Hendler, 2004], “confidence” or “faith” [Shadbolt, 2002], “credibility” or “reliability” [Golbeck et al, 2003], “competency” or “honesty” [Grandison & Sloman, 2000], and “belief” [McKnight & Chervany, 1996]. Such a diverse set of trust concepts, found across a single body of research, reflects the real world where a wide and varied range of individual subjectivity in what trust entails seems to exist across a large and broad population.

Trust mechanisms are not commonly deployed in applications widely used in the Internet, for example Instant Messaging, email, social networks, and so on. The author of this thesis speculates that part of the reason for this relates to the models of trust currently being used, and the lack of personalisation within trust models.

To date, trust within Internet environments has been described and defined using one of a number of different trust concepts to form a single-faceted model of trust to support trust based decision making, for example eBay’s reputation based feedback system. In addition, many other trust management systems including REFEREE [Chu et al, 1997], SULTAN [Grandison et al, 2001], Advogato [Levien et al, 1998], and FilmTrust [Golbeck et al, 2006] also use such a single-faceted approach. A single-faceted approach to modelling trust can appeal to some, or many, individuals but it is hypothesised in this thesis that a single-faceted approach cannot capture the wide and varied range of subjective views of trust found across a large and broad population. Therefore, this thesis asserts that a single-faceted approach is an inadequate model of

trust for use in Internet environments where trust management services are provided to individuals within a large population. What is needed is a multi-faceted model of trust that can capture the range of trust concepts, and relationships between these trust concepts, that are used to describe trust. Without a multi-faceted model a wide range of users will not be catered for, which may lead to large scale user dissatisfaction.

Current models of trust may be applicable to the different domains in which they operate but in many models no provisions are made to tailor the model of trust more accurately to other domains. However, websites such as Epinions [Epinions] do provide such specialisation. Specialisation is defined in this thesis as the ability to engineer a model of trust towards a particular domain. For example, a camera on Epinions can be reviewed with respect to camera specific properties such as ‘shutter lag’ and ‘photo quality’. With the aid of specialisation it is possible to create more complex models of trust that can more accurately reflect the trust requirements of different application domains. Thus, specialisation is also required when modelling trust in Internet environments. It is important to note that within current models of trust it is the application developer who specialises a model of trust towards a given application domain, which is a positive feature for websites like Epinions. However, if the end user is also provided with the ability to specialise a model of trust then this may lead to a greater uptake and use of domain specific models by communities in Internet environments. Without either form of specialisation it would not be possible to reflect the different properties, relationships, and classes that are found across multiple application domains, which would result in an undefined, unscoped, and less useful trust management system, and trust management service offerings such as trust based recommendations.

Personalisation is defined in this thesis as the ability of the model of trust to capture an individual’s subjective view of trust and, at the same time, also capture the wide variety of subjective views of trust that are exhibited by individuals over a large and broad population. Personalisation, as defined above, does not exist within models of trust in current Internet environments. This lack of personalisation means that current models of trust are not capturing the subjective nature of trust for its users, which means that trust management systems are not catering for the individual in order to provide meaningful and useful service offerings.

1.2 Research Question

The research question posed in this thesis is *whether a multi-faceted model of trust that is personalisable and specialisable is both necessary and accurate to the user in providing a dynamic and flexible trust based decision support mechanism within Internet environments.*

1.3 Objective and Goals

The main objective of this thesis is to design, develop, and evaluate a multi-faceted model of trust that is personalisable and specialisable, which can be used to enable trust based decision support in Internet environments.

In order to investigate the research question the following goals were derived:

1. Research the state of the art in trust, focusing primarily on models of trust and trust management systems in order to identify whether there is a consensus on what trust is and how trust management operates.
2. Design and develop a multi-faceted model of trust that is personalisable and specialisable.
3. Evaluate the necessity for a multi-faceted model of trust that is personalisable and specialisable.
4. Design and develop a trust management service that has a mechanism for generating personalised models of trust, which provides trust based recommendations. Evaluate the ability of the generation mechanism to produce personalised models of trust that accurately reflect users' ideas of trust. Evaluate the accuracy of trust based recommendations calculated using the developed trust management service.
5. Develop two case studies to illustrate and compare specialisation in an evidence based application domain and in an opinion based application domain.
6. Illustrate the ability of the model of trust to provide dynamic and flexible management by providing a trust management service to a policy based management system as part of a use case scenario.

1.4 Technical Approach

An initial state of the art study in the area of trust was conducted, which concentrated on current approaches to modelling and representing trust and also concentrated on trust management systems. This literature survey aided the understanding of the issues surrounding the modelling and use of trust in the state of the art. The design for a multi-faceted model of trust has influences from the state of the art, which is most visible in the choice of trust concepts incorporated in the model. In chapter 3 (see Section 3.2) details and discusses the full set of state of the art influences.

The multi-faceted model of trust is comprised of an upper ontology, meta-model, specialised models, and also personalised models that are generated on a per user basis. The upper ontology, meta-model, specialised and personalised models are represented in the Web Ontology Language (OWL) [McGuinness & van Harmelen, 2003], which supports semantic reasoning. In addition, OWL provides support for heterogeneity, which makes OWL extensible and interoperable. The upper ontology is an extensible source of trust concepts that can be used to engineer domain specific models of trust and generate personalised models of trust. The meta-model governs the relationships that can exist between trust concepts found in the upper ontology when engineering specialised models and generating personalised models. This separation allows the upper ontology and meta-model to be used independently to enable the design and development of specialised models of trust towards multiple domains and the generation of personalised models of trust on a per user basis. Two specialised models were designed and developed towards different application domains, namely Web Services and Instant Messaging. Over 200 personalised models of trust were generated, independently by test subjects, during experimentation to capture each individual's subjective view of trust.

Four experiments and three trials were designed and developed. The first experiment evaluated the necessity of a multi-faceted, personalised model of trust. This was carried out by survey through an online questionnaire that identified the various ranges of subjectivity found within individual views of trust over a broad population. The second experiment required the specification of policies for regulating access control to a set of objects, including a mobile phone and laptop. The second

experiment evaluated the accuracy of trust based recommendations calculated by the trust management service, which used each test subject's personalised model of trust, along with information comprising annotated trust values and access control policies that each test subject provided directly. The third experiment further evaluates the accuracy of the personalised model of trust by offering each test subject the opportunity to receive additional information. Furthermore, the third experiment examines what effects, if any, such additional information has on trust based decision support systems. Experiment four provides insight into the clarity of the trust concepts as perceived by the test subjects.

The first of the three trials tested a proof of concept system that allowed the user to select Web Services based on the trustworthiness of those Web Services, as rated by many users. The second and third trials enabled the developed trust management service, *myTrust*, to be utilised by a Community Based Policy Management (CBPM) [Feeney, 2004] system in order to provide dynamic and flexible access control within the PUDECAS ubiquitous computing simulator [O'Neill et al, 2006] and separately for a decentralised enhanced Instant Messaging (IM) application [Kenny et al, 2006].

The experiments and trials demonstrate that the use of a multi-faceted model of trust that is personalisable and specialisable enables accurate and automated trust decisions to be made on a user's behalf in Internet environments.

1.5 Contribution

The first contribution of this research is the novel strategy in which modelling trust is accomplished through applying a personalisable and specialisable approach to a multi-faceted model. In this thesis it was found that the trust concept *reputation* ranked highest across a broad population of test subjects, yet seventy one percent of this population think differently and instead a different trust concept is ranked highest in their cases. This demonstrates the wide range of heterogeneity in peoples' use of trust concepts. In addition, two domain specific models of trust investigated and developed in this thesis illustrate the differences in complexity and properties that different application domains can exhibit. The multi-faceted model of trust that is personalisable and specialisable provides a "one model of trust fits all" solution that can be used by trust management systems to provide personalised and specialised

trust models, trust information, and trust based recommendations. Such a model of trust can capture an individual's subjective view of trust and, at the same time, also capture the wide variety of views of trust that are exhibited by individuals over a large and broad population. The author of this thesis believes that providing a more bespoke and useful trust management service in Internet environments will deliver a better and more satisfying user experience and allow for even greater trust service offerings than the non-personalised, single-faceted approaches used by eBay, Advogato, SULTAN, and REFEREE.

In [Jonker et al, 2004], it is stated that trust “theories and models are not often verified experimentally”. Such experiments are necessary in order to evaluate the necessity for a particular model of trust and illustrate that the model is in some way useful. Therefore, the results from the experiments carried out in this thesis are a second contribution as they provide a rich source of information regarding peoples’ attitudes to trust in Internet environments. The results suggest that as risk increases a persons need for, or reliance on, trust also increases, and highlights where recommendations based on automatic trust calculations could be used less and where they should be used more. In addition, it is shown through examination of the clarity, importance, influence, and rank of trust concepts that people find some trust concepts more defined, scoped, and clearly understood than others. This could impact on the selection of a trust concept for use in single-faceted models of trust in Internet environments as better understood and more defined trust concepts might be chosen over ambiguous trust concepts that are not as clearly understood. The results also indicate that there are specific circumstances where trust based recommendations made using the multi-faceted model of trust are satisfactorily accurate, and where they are less accurate. In addition, the results show that the provision of additional information increases the overall accuracy of recommendations. Furthermore, the results identify where additional information will yield increases, and decreases, in the accuracy of trust based recommendations. This suggests that providing accurate trust based recommendations sometimes requires that additional information should be provided, which again impacts on the design of trust management systems for the Internet. Thus, the experiment results can contribute to the specification and design of future trust management systems, irrespective of whether a multi-faceted model is adopted or not.

1.6 Overview of Thesis

Chapter 2 provides a review and analysis of the state of the art in modelling trust, including the various definitions of trust. The state of the art also analyses selected trust management systems.

Chapter 3 presents the design of the multi-faceted model of trust, including the upper ontology, meta-model, multiple specialised models, and personalised models of trust. The designs presented include influences from the state of the art provided in Chapter 2. The design for two models of trust specialised towards Web Services and Instant Messaging are presented. The design of the overall framework in which the multi-faceted, personalisable and specialisable model of trust resides is detailed and discussed. In addition, the mechanism for generating personalised models of trust is detailed.

Chapter 4 describes the implementation of the multi-faceted, personalisable, specialisable model of trust. The implementation of the overall framework that the model of trust operates in, *myTrust*, is also described.

Chapter 5 describes the experiments developed for this research. Four experiments were completed by a broad and diverse range of test subjects. The initial experiment is used to evaluate the argument for a multi-faceted, personalised approach to trust. The second and third experiments evaluate the accuracy and applicability of the personalised models of trust. Experiment four examines the clarity of the trust concepts.

Chapter 6 describes three trials that enabled Internet based applications to use the developed trust management framework in order to illustrate the usefulness of the approach.

Finally, chapter 7 describes how well the objectives of this thesis were achieved, discusses the contributions made, presents salient suggestions for future work, and concludes with some final remarks.

2 STATE OF THE ART

2.1 Introduction

This chapter presents a literature review and survey of trust management systems. The literature review, Section 2.2, is scoped to the various definitions of trust, which are found across many research areas. The survey of trust management systems, Section 2.3, illustrates the various different approaches to trust management that have emerged, ranging from early credential based systems to current online social networking systems. In addition, a comparison framework is presented and the reviewed trust management systems are analysed based on that framework. Section 2.4 provides a short summary of the chapter.

2.2 Literature Review: Definitions, Ideas, and Views of Trust

The definitions, ideas, and views of trust reviewed in this section are sub-grouped into Dictionary, Psychology, Sociology, and Computer Science. The sub-grouping is necessary to illustrate the wide and varied definitions, ideas and notions of trust that are found across a range of research areas.

2.2.1 Dictionary

The Oxford English Dictionary and Cambridge English Dictionary have several definitions for trust, including:

“Confidence in or reliance on some quality or attribute of a person or thing, or the truth of a statement.” (Oxford)

“To have belief or confidence in the honesty, goodness, skill or safety of a person, organization or thing.” (Cambridge)

“The quality of being trustworthy; fidelity, reliability; loyalty, trustiness.” (Oxford)

These definitions suggest that trust is based on a certain quality that a person, or entity, holds. This is quite a common view of trust, where trust is defined using a range of synonyms for trust made specific to some entity. But trust is also defined specific to business and commerce in:

“Confidence in the ability and intention of a buyer to pay at a future time for goods supplied without present payment.” (Oxford)

Once again synonyms for trust, confidence and competency, are used to define the term trust. The synonyms used in the reviewed dictionary definitions include confidence, reliability, belief, honesty, goodness, competence, and loyalty. With such a range of synonyms, provided by various sources, it is possible to assert that trust is subjective to a source and that trust varies from source to source. There are also various different fields of research that have provided definitions and analysis of trust. As Grandison pointed out in [Grandison, 2003] it is McKnight et al [McKnight et al, 1996], Lamsal [Lamsal, 2001], Gerck [Gerck, 97] and Corritore [Corritore et al, 2001] that provide detailed discussion and analysis on trust definitions in philosophy, sociology, psychology, business management, marketing, ergonomics and Human-Computer Interaction. This literature review focuses on the areas of psychology, sociology, and computer science as they are the most cited sources of research work in computer science related trust publications.

2.2.2 Psychology

Deutsch [Deutsch, 1962] definition of trust is frequently referenced, it is as follows:

1. If an individual is confronted with an ambiguous path, a path that can lead to an event perceived to be beneficial (Va+) or to an event perceived to be harmful (Va-),
2. He perceives that the occurrences of (Va+) or (Va-) is contingent on the behaviour of another person,
3. He perceives that the strength of (Va-) to be greater than the strength of (Va+),

4. If he chooses to take an ambiguous path with such properties, I shall say he makes a trusting choice; if he chooses not to take the path, he makes a distrustful choice.

In this definition, Deutsch does not directly use any synonyms for trust and instead states that trust is a perception. In this way trust becomes subjective to the individual. Trust is also specific to another entity, a person, and the decision to trust is made with respect to that person. Note however that [Golembiewski et al, 1975] disagrees with the third assertion where Deutsch states that damage should outweigh benefits.

2.2.3 Sociology

Gambetta [Gambetta, 1998] states that trust is subjective and based on the actions of another entity, an agent. In addition, the actions of the other agent can not be monitored and the other agent's action affects his own actions.

“trust (or, symmetrically, distrust) is a particular level of the subjective probability with which an agent assesses that another agent or group of agents will perform a particular action, both before he can monitor such action (or independently of his capacity ever to be able to monitor it) and in a context in which it affects his own action” [Gambetta, 1988]

Hart's definition of trust [Hart, 1998] states that trust lies in a range of synonyms for belief, somewhere between faith and confidence:

“Trust thus stands in the middle of a continuum of words for belief mixing extremes of blind faith and open-eyed confidence.”[Hart, 1998]

Sociology has also provided research that illustrates what use trust is to society. In [Luhmann, 1979], Luhmann sees trust as a tool for reducing complexity of decision in society. This is achieved through a mechanism where trust provides internal security ahead of taking an action in the face of uncertainty and incomplete information. In [Misztal, 1996], Misztal states that trust makes social life (i) predictable, (ii) creates a sense of community, and (iii) makes it easier for people to work together. Therefore, it can be asserted that trust is a vital part of society.

2.2.4 Computer Science

A seminal contribution to trust research within computer science can be attributed to [Marsh, 1994], where Marsh states that “there are many views of trust”, and in doing so cites the work of Shapiro [Shapiro, 1987] and Barber [Barber, 1983]. Marsh argues that with respect to trust “we are all ‘experts’ on trust, at least our own brand of it, and there is the problem, since, as there are so many different ‘experts,’ each of which could define trust differently, there are as many differing definitions, and thus views, of trust” [Marsh, 1994]. Subjectivity within trust is also echoed by Abdul-Rahman [Abdul-Rahman, 2005] and Dieter Gollmann [Gollmann, 2005].

Like most of the definitions of trust that were presented in the previous sub-sections, computer scientists have had a tendency to use synonyms for trust in their research. The majority of research describes trust using a single-faceted approach, in other words only one synonym for trust is used, whereas the minority of research uses a multi-faceted approach, in other words more than one synonym for trust is used.

In [Golbeck et al, 2003], Golbeck states that she addresses trust as “credibility or reliability in a much more human sense”. In [Golbeck et al, 2004], Golbeck states that “reputation is more a social notion of trust”. Golbeck’s latest research, [Golbeck et al, 2006], has focused on a single-faceted approach that has settled on reputation.

In [Shadbolt, 2002], Shadbolt states that “Trust is also a matter of developing confidence in the decisions of others. It is about having faith in the information and knowledge that they possess. It is about believing that the processes that lead to their decisions are well founded and well informed.” The synonyms confidence, faith, and belief therefore comprise the multi-faceted approach associated with Shadbolt’s research work.

In [Bargh et al, 2002], trust is defined as dependability; where in turn “dependability can be seen as a system property consisting of security, reliability, availability, safety, timeliness, and maintainability attributes”. Furthermore, security is seen as “identification, authentication, confidentiality, integrity, access control, and non-repudiation”. The key aspect to the work of Bargh, from the perspective of the

research work presented in this thesis, is that trust is a term comprised of a set of sub-classes; trust, dependability, security, and so on.

[McKnight & Chervany, 1996] presents a review and analysis of a wide range of trust publications, which are from a diverse set of domains including: management, economics, politics, psychology, and sociology. The research presented in [McKnight & Chervany, 1996] is often quoted in computer science and thus it appears in this subsection. Their research summarises all the views presented in the publications they evaluated, which is that trust is based on six constructs; Trusting Intention, Trusting Behaviour, Trusting Beliefs, System Trust, Dispositional Trust, and Situational Decision to Trust.

Trusting Intention is “the extent to which one party is willing to depend on the other party in a given situation with a feeling of relative security, even though negative consequences are possible”. Trusting Behaviour is “the extent to which one person voluntarily depends on another person in a specific situation with a feeling of relative security, even though negative consequences are possible”. Trusting Beliefs means “the extent to which one believes (and feels confident in believing) that the other person is trustworthy in the situation”. System Trust means “the extent to which one believes that proper impersonal structures are in place to enable one to anticipate a successful future endeavor”. A person has Dispositional Trust to “the extent that s/he has a consistent tendency to trust across a broad spectrum of situations and persons”. Situational Decision to Trust means “the extent to which one intends to depend on a non-specific other party in a given situation”.

Therefore, it could be interpreted from [McKnight & Chervany, 1996] that trust is a relationship, with an associated level of belief that occurs within a context and provides a secure feeling in the face of risk. In addition, people have a disposition towards trust across a spectrum of situations and towards certain people. Furthermore, trust is also based on a particular situation. The term belief is quite useful, from the perspective of this thesis. In the review of literature to date more concrete synonyms for trust have been used, like reputation and reliability, yet belief is somehow more abstract. In terms of designing a model of trust the idea of a separation between a clear and definable set of concepts and more abstract concepts is significant.

In [Grandison & Sloman, 2000], Grandison has defined trust as “the firm belief in the competence of an entity to act dependably, securely, and reliably within a specified context (assuming dependability covers reliability and timeliness)”. He continues to state that “Trust is a complex subject relating to belief in the honesty, truthfulness, competence, reliability etc. of the trusted person or service.” It is interesting to note that distrust is defined as “the lack of firm belief in the competence of an entity to act dependably, securely, and reliably within a specified context.” Therefore, the synonyms belief, competency, honesty, and reliability are key in Grandison’s multi-faceted view of trust. In addition, Grandison states in [Grandison & Sloman, 2000] and [Grandison, 2003] that ‘there is no consensus on the meaning of trust in trust management’. This is a key statement in the context of the research presented in this thesis.

Summary

From the literature review there is clearly no single real consensus, or definition, of trust. Several dictionary definitions have used a wide and varied range of synonyms for trust including confidence, reliability, belief, honesty, goodness, competency, and loyalty. Several of these trust concepts have been utilised by different computer science research projects in describing the term trust. The research work presented in this thesis takes great influence from the assertions of Marsh, Deutsch, Gambetta, and Abdul-Rahman, where trust is stated as being subjective to individuals.

In the next section the survey of trust management systems presents a review of several trust management systems and includes an expansion on some of the research work presented in this section. In addition, the survey of trust management systems reveals crucial aspects to trust management that effect both the design of the multi-faceted model of trust that is personalisable and specialisable model of trust, and also the *myTrust* implementation.

2.3 Survey of Trust Management Systems

The survey of trust management systems is presented in four categories, which mark the evolution of trust management systems over the last decade. This categorisation is explained in Section 2.3.1.1. A comparison framework, presented in Section 2.3.1.2, is used to describe and analyse selected trust management systems.

The trust management systems selected for review were chosen based on citation numbers and how well established each system is in the state of the art. Furthermore, some emerging systems found in the current state of the art were also selected.

2.3.1 Survey Categorisation and Comparison Framework

This sub-section explains the survey categorisation model and the comparison framework used to analyse the selected trust management systems.

2.3.1.1 Survey Categorisation

In analysing the selected trust management systems it became apparent that they could be generally categorised into four categories; (i) credential, (ii) internal, (iii) online community, and (iv) social network based trust management.

In credential based approaches (Section 2.3.2) trust is determined outside of the system by the user and credentials, usually certificates, are subsequently allocated by a user based on trust. Therefore, in general these systems provide no mechanisms for trust annotation or trust calculation in order to determine a level of trust. The category of internal trust management systems (Section 2.3.3) does provide such annotation and calculation mechanisms, which has the effect of transferring the trust decision from outside the system to inside the system. The online community based systems (Section 2.3.4) tend to enable community members to annotate a member, or other entity, with trust information. Multiple sources of trust data, generally received from an unknown community member, can then be used to provide a trust based recommendation, which is made using a trust calculation algorithm. Social network based trust management systems (Section 2.3.5) are an extension to online communities where defined paths of known members may exist from a member who seeks trust data to a provider of trust data. Trust calculation algorithms for social networks may utilise such defined paths in providing trust based recommendations.

2.3.1.2 Comparison Framework

A comparison framework was developed from a set of factors that emerged when reviewing and analysing each trust management system in the state of the art review. The following comparison six headings were used: (i) model of trust, (ii) trust annotation, (iii) trust calculation, (iv) policy, (v) architecture, and (vi) trust representation format.

Each reviewed trust management system has an analysis section that addresses some, or all, of these six comparison factors depending on their applicability. These factors are important to examine as they can be used to describe and analyse a wide range of trust management systems. The model of trust, trust annotation, and trust calculation are core features of many trust management systems, especially those categorised as internal, online community, and social network trust management systems. The presence of policy can indicate that the trust management system provides automated trust services to a user, which is based on the trust values calculated using a trust calculation algorithm. The system architecture and trust representation format are useful in describing the implementation approach of a trust management system.

Model of Trust

This heading relates to the model of trust that the trust management system utilises. In this comparison framework a model of trust can be single-faceted or multi-faceted, and be personalisable, and specialisable. The option of single-faceted or multi-faceted was chosen as it reflects the approaches to the definition of trust found in computer scientist research. From the research it is also asserted that trust is subjective, and so the degree of personalisation provided by the system is a key in comparing trust management systems. Here, personalisation refers to the ability of the model of trust, and thus the trust management system, to capture the individual's subjective views of trust. Specialisable was chosen as a comparison factor as many of the definitions and some trust management systems argue that trust is specific to some area or domain; for example medical procedures. However, the comparison framework makes the distinction between a trust management system that provides a domain specific model of trust that is pre-specialised by a developer, and a trust management system that allows the end user to specialise domain specific models of trust as required.

Trust Annotation

The trust annotation method of a system is also compared. Trust annotation can be opinion based, evidence based, or a combination of both. As seen across the state of the art trust is a human idea that is built on opinion, evidence, or a combination of both. Therefore, a trust management system that can capture evidence and opinion trust data can provide trust based recommendations with the aid of a trust calculation.

Trust Calculation

The trust calculation mechanism can be categorised as either simple or advanced. A simple trust calculation algorithm would refer to an aggregation algorithm that sums people's scores, like in eBay. Whereas, advanced trust calculations use relatively complex mathematical algorithms, like in TidalTrust [Golbeck, 2005]. It must be noted that there is a wide body of research in computer science that studies computational trust, which Abdul-Rahman compares in [Abdul-Rahman, 2005]. However, computational trust is outside the scope of this thesis.

Policy

Policy allows the users of a trust management system to specify rules based on trust values. Trust management systems have previously, and successfully, used policy to give the user a level of empowerment over these trust values in order to automatically regulate the environment in which the trust management system operates. For example, a policy might state that user B can view user A's files, if user A has sufficient trust in user B. Under this heading a comparison is made as to whether policy is used in the trust management system. More specifically, the use of policy can be regarded as internal or separate. If policy is described as internal then the trust management system has its own policy system. If policy is described as separate then this would mean that the trust management system and policy system are separate. It is however also possible for policy to be completely absent in trust management systems.

Architecture

The architecture of the system is classified to signify the way in which trust models, trust data, and trust values are managed, which can be centralised or distributed.

Trust Representation

From the state of the art it is clear that current trust representation formats in research are XML or OWL based. Therefore, the representation mechanism that the system uses is also compared on the basis of XML, OWL or other.

2.3.2 Credential Based Trust Management Systems

Credential based approaches essentially enable management functionality, such as access control, through the use of credentials that have been provided to a user within the system by another user. In the case of access control a user could gain access to a file, or similar, so long as that user has the correct credentials. The process of determining whether a user has enough trust to acquire a necessary credential is external to the trust management system. It is the outcome of such decisions that are managed by credential based trust management systems.

2.3.2.1 X.509 and PGP

X.509 [Adams et al, 1999, X509] is a centralised system that provides certificates which correspond to a Distinguished Name (DN). In essence it enables an entity such as Trinity College Dublin to apply to a Certificate Authority (CA) for an X.509 certificate that is unique and can be used to authenticate Trinity College Dublin in an eCommerce transaction or similar. The CA that issues this certificate has its own certificate that was issued by another CA higher up the CA tree hierarchy. Certificates can then be validated at several levels; student – college – government – international.

Pretty Good Privacy (PGP) [Zimmerman, 1995] is a decentralised approach to certificate based trust management. PGP was originally used to send secure emails with decentralisation achieved through a *web of trust*. In PGP users sign each others public key certificates. The initial confidence is built upon a direct trust relationship between two users, for example Alice and Bob. When Alice and Bob sign each others public key an interconnection is created between the two. Assume that Bob and Carol also sign each others public keys. It is now possible for Carol to authenticate Alice's public key through the web of trust, in other words through the relationship between Carol, Bob, and Alice.

PGP allows users to state how confident they are that a public key matches the owner. Alice can state that she has (i) *undefined*, (ii) *marginal*, or (iii) *complete* confidence in the validity of a public key. These confidence levels can be shared. PGP also allows Carol to state a level of “introducer trust” for Bob; how trusted Bob is in introducing users. Carol can state that Bob is (i) *fully trusted*, (ii) *marginally trusted*, (iii) *untrustworthy*, or (iv) *don't know*. These trust levels are not shared.

PGP allows a user to define how many *marginally trusted* and *completely trusted* signatures are required to make a public key certificate completely valid. Carol creates what is in effect a policy stating that she requires, for example, at least two *completely trusted* signatures to accept a public key. If Alice and Bob are *fully trusted* by Carol and Alice and Bob have *completely trust* in the public key that they know as Dawn's then the policy has been satisfied and Carol will accept Dawn's public key. PGP enables the annotation and utilisation of trust information and uses what is in effect a policy to reconcile such information. However, PGP is used to authenticate a public certificate key with a user and the level of trust in a user is kept secret and not shared. On its own PGP can not be used to regulate access control or provide other management functionality. However, PGP is a precursor to credential based trust management systems.

2.3.2.2 PolicyMaker and KeyNote

PolicyMaker [Blaze et al, 1996, Blaze et al, 1998a] is recognized as the first trust management system, or more correctly stated it is the first trust management system that provides management functionality based on trust. Its successor is KeyNote [Blaze et al, 1998b]. Both these systems provide mechanisms for regulating access control based on certificates. KeyNote builds upon PolicyMaker by adding two design goals, namely standardisation and easy integration into applications.

PolicyMaker and KeyNote grant authorisation without the need to authenticate a user. Instead of the two step process of authentication and authorisation PolicyMaker and KeyNote address the authorisation problem directly. In a two step process the first question asked would be “is this person who they say they are?”, which PGP could reliably answer. The second question would be “does this person have the correct access control permissions for this request?”. Rather than answering both questions

PolicyMaker and KeyNote ask the question “is the key that signed this request authorised to take this action?”. In essence, granted rights are bound to public keys. PolicyMaker uses policies and credentials. Policies are issued by the trust root, which is the ultimate source of authority. Credentials are issued by public keys (signed by the issuer) and are verified before use. Credentials and policies are collectively known as *assertions*. An *assertion* is represented as a pair (f,s) . The term s refers to the source of authority (the issuer). The term f is a program describing the nature of the authority being granted as well as the party(s) to whom authority is granted. Applications that use PolicyMaker collect certificates and translate them to PolicyMaker assertions. The application then invokes the PolicyMaker inference engine by providing a query accompanied by a set of credentials and policies. PolicyMaker then provides a proof that a request complies with policy (granted), or it does not provide such a proof (denied).

In KeyNote less responsibility is assigned to the application in comparison to PolicyMaker. In addition, KeyNote policies and credentials are written in a specific assertion language. An application passes to a KeyNote evaluator a list of policies, credentials, and requester public keys, and an *action environment*. The *action environment* consists of a list of attribute/value pairs, which contains all relevant information to the request and information that necessary for a trust decision. The evaluator returns to the application a string, defined by the application, such as ‘authorised’.

2.3.2.3 REFEREE

The Rule-controlled Environment For Evaluation Of Rules And Everything Else (REFEREE) [Chu et al, 1997] is a system that is based on PolicyMaker and KeyNote that provides trust management for web applications.

REFEREE is an environment for evaluating compliance with policies. A policy language and policy evaluation mechanism is provided for specifying and evaluating trust policy. REFEREE differs from PolicyMaker in that REFEREE permits policies to control credential retrieval and signature verification. Evaluating trust policy on the web may involve dangerous actions (for example retrieving spoofed credentials) and therefore REFEREE uses policy to assert a level of control. For example, a policy can

state that a credential written in one policy language may be executed but a credential written in another policy language might require the language to be vouched for by a trusted party. In REFEREE it is possible to defer, or delegate, trust to a privileged credential. In this way a privileged credential can decide to accept another credential on the user's behalf. REFEREE also allows a user, or organisation, to apply a rating to an entity for a specific attribute. These ratings can be shared and used as part of the policy evaluation mechanism. For example, Alice can state that a music distribution service makes use of the saxophone once in its music files and Bob can state that the same service makes use of the saxophone twice in such files. A policy can be created that states that music services with no more than one use of the saxophone will only be used. It is possible to direct REFEREE to prioritise Alice over Bob, or vice versa.

2.3.2.4 IBM Trust Establishment Framework

The IBM Trust Establishment Framework [Herzberg et al, 2000] is closely related to PolicyMaker and KeyNote in that it is providing management functionality based on trust by using certificates. However, a certificate is mapped to a role using Role Based Access Control (RBAC) [Ferraiolo et al, 1992]. A policy then defines what a role can do. In this way a credential associates an entity with a role and a role can perform certain actions according to a policy.

The three components of the IBM Trust establishment Framework are: (i) Certificates, (ii) Trust Evaluation module, and (iii) Trust Policy Language (TPL) [Ferraiolo et al, 1992]. The Trust Establishment module validates a certificate and maps that certificate to a role. A certificate can state that the owner is a member of an organisation or state that the owner holds a particular position within that organisation. Once mapped to a role the framework can identify what the owner (within the role) is permitted to do. This process uses the TPL, which is XML based. It is the TPL that is used to define what a role is permitted to do.

2.3.2.5 Other Notable Credential Based Approaches

Vigil [Kagal et al, 2002] is a trust management system that is specific to pervasive computing environments. It uses RBAC and an ontology based policy language to represent rules. The system provides trust based on the user's role and where roles can be changed based on a user's actions or context. A system of delegation is used to allow users with no access rights to access a particular resource so long as a user who has the correct access rights delegates this ability.

TrustBuilder [Winslett et al, 2002] is a ubiquitous architecture for scalable trust negotiation, which establishes trust between strangers by gradually disclosing credentials. TrustBuilder can be used where two entities from different security domains need to establish trust. The policy language and compliance checker use IBM Trust Establishment Framework. TrustBuilder also supports Role-based Trust Management Language (RT) [Winsborough et al, 2002] credentials and RT run-time decision engine.

2.3.2.6 Overall Analysis of Credential based Trust Management

PolicyMaker and KeyNote are both formidable trust management systems in which a user makes a decision based on trust (for example can Alice access Bob's files?) and these systems reflect that decision through the use of credentials. However, no trust annotation mechanism or calculation mechanism based on that trust annotation is used. REFEREE, an extension of PolicyMaker, provides such annotation and calculation (aggregation and average) ability. The ability to annotate entities with trust information, make calculations, and subsequently reconcile calculation outcomes against a policy in order to provide management functionality is an idea that is central to the approach that is adopted in this research.

IBM Trust Establishment Framework offers a different approach to trust management based on certificates in comparison with the PolicyMaker and KeyNote approach; membership to roles is the determining factor in the identification of permissions. However, this credential based trust management system does not offer a mechanism for trust annotation or a mechanism for trust calculation that operates in conjunction with the annotation information to provide a trust based recommendation.

The majority of trust management systems reviewed all make extensive use of policy in their operation. Policy is used to specify rules that work in conjunction with the trust management system to provide management functions such as access control. It is IBM's Trust Establishment Framework and Vigil that employ advanced use of policy by using a RBAC approach. In comparison to previous uses of policy in trust management systems a RBAC approach has advantages. Mapping a role to a policy, thus defining permission for that role, can reduce the administrative overheads involved in policy creation as many people can fit into a single role and roles can inherit permissions. In addition, RBAC benefits from the principle of least privilege (reduces damage) and separation of duties (reduces fraud). However, more advanced use of policy is desirable in the research presented in this thesis in order to maximise the potential of the multi-faceted, personalisable, specialisable model of trust.

In the majority of the trust management systems that are based on credentials the decision to trust is made external to the system. However, the credential based PGP and REFEREE include the ability to make a decision based on trust within the trust management system. Therefore, PGP and REFEREE can be considered as approaches that can be categorised as credential based trust management systems and internal trust management systems. PGP and REFEREE provide the ability for trust annotation and trust calculation but they do not provide any element of personalisation within the calculation. REFEREE also provides a level of specialisation within its annotation and calculation, which is an approach that is used in the research work presented in this thesis.

2.3.3 Internal Trust Management Systems

In the previous section the decision to trust was made external to the trust management system. In this section the reviewed trust management systems provide a trust annotation mechanism that allows users to express their trust in other users. In addition, a trust calculation algorithm is also present and can be used to provide an overall assessment of trust, or trust based recommendation based on trust annotation data. Therefore, the decision to trust is no longer external to the trust management system; it has become internal to the trust management system.

2.3.3.1 SULTAN

SULTAN (Simple Universal Logic-oriented Trust Analysis Notation) [Grandison et al, 2001] is a Trust Management Framework that is designed to facilitate the management of trust relationships. It is a collection of specification, analysis and management tools, which are comprised of five components; Specification Editor, Analysis Tool, Risk Service, Monitoring Service, and Trust Consultant.

The Specification Editor is a composite toolkit for creating, storing, retrieving, editing and translating SULTAN Specifications [Grandison et al, 2003]. These specifications are used to specify trust and recommendations statements, which can be either positive or negative. A positive trust statement generally takes the form: UniqueTrustName: **trust** (Tr, Te, As, L) \leftarrow Cs, which translates as Tr (a trustor) trusts Te (a trustee) to perform an action (As) at a trust level (L) if constraints (Cs) is true. An example of such a trust statement, taken from [Grandison et al, 2003], is provided in Figure 2-1.

```
Realtor: trust ( Jenny, Realtor, send_deals(Realtor, Jenny), HighTrust)
← trust (Jenny, Tom, ProvideInfo(Jenny), MediumTrust ) |
  trust (Tom, Realtor, send_deals(Realtor, Tom), MediumTrust );
```

Figure 2-1 SULTAN Trust Statement Example

Figure 2-1 can be interpreted as follows: Jenny trusts Realtor to perform send_deals(Realtor, Jenny) at trust level HighTrust if Jenny trusts Tom to perform ProvideInfo(Jenny) at trust level MediumTrust or if Tom trusts Realtor to perform send_deals(Realtor, Tom) at trust level MediumTrust. In this way SULTAN provides Jenny with the ability to delegate the constraints to Tom, effectively stating that Jenny trusts the Realtor to send deals to her if Tom trusts the Realtor to send deals to him. The levels of trust can be integer based or string based as in this example. These trust specifications are entered into the editor, which compiles, saves, and translates them to Prolog for analysis. A tool also translates Prolog based specifications to Ponder [Damianou et al, 2001] policies.

The SULTAN Analysis Tool allows an administrator to produce simulation analysis and property analysis. Simulation analysis enables the addition of information into the Prolog System. Questions can then be asked through Prolog or the Prolog based SULTAN Analysis Model. Property analysis validates whether specific properties hold on trust and recommendation rules. The Risk Service retrieves risk information and performs risk calculations. Risk information includes any evidence of prior service failure or fraud. A risk value is calculated from a given subject, target, action, and risk identifier. The Monitoring Service provides up to date information for risk, experience, and system state information. As new information is added it is possible for SULTAN to test for potential conflicts, which are reported as they occur. It is the Monitoring Service that makes SULTAN adaptable to capture trust as it changes within the system. The Trust Consultant enables the end user to query the SULTAN Trust Management Framework with questions such as ‘Should I trust target X to perform action A?’. SULTAN will return a tuple form (boolean_answer, justification) for such a query, which might read ‘yes’, ‘authorisation confirmed’.

In his doctoral thesis [Grandison, 2003] Grandison provides a use case scenario in which SULTAN is incorporated into an Internet-based reservation system for the hotel industry that aims to provide increased revenues for partners and hotels, and a more efficient and effective shopping experience for travellers. Initially, the administrator maps the organisation as a set of facts in SULTAN, for example a traveller is a customer who uses the reservation system. Then actions are generated from the service offerings of the company, for example search for and book a hotel room. Trust relationships are then added, for example a traveller is trusted to responsibly book a hotel room. Risk specifications are then made and analysis queries are designed. Specific trust information is then added to test the integration. It states that a traveller, Jane, has created an account and booked a hotel room. The successful completion of a booking results in Jane receiving a positive experience from the reservation system. This new information is added to the SULTAN information database and SULTAN checks for conflicts. If Jane returns to use the reservation system the reservation system can ask SULTAN ‘Can I trust Jane to search and book a hotel room’. SULTAN will take into account the prior positive experience and inform the reservation service that it should grant Jane the ability to search and book a hotel room.

Analysis

An analysis of the SULTAN Trust Management Framework shows that there are many positives aspects to the approach taken in its design and development. One of SULTAN's key elements is its ability to handle and use risk through its Risk Service. As risk increases the possibility to increase the amount of required trust is available, thus capturing the organic ebb and flow of real life environments and adapting to meet the needs of the new environment. SULTAN can also adapt based on new information and the re-evaluation of that information. The application domain in which SULTAN has been integrated uses domain specific properties that are used to describe a user's trust, for example successfully booking a hotel room increased the trust of a user with respect to online reservation system. Such domain specialisation can provide more useful consultation results.

SULTAN also has some notable weaknesses in its design and development, which Grandison notes in his doctoral thesis. No trust calculation mechanism is provided to provide a level of trust. Such a mechanism, in SULTAN, would make the specification process easier for the administrator. There are no facilities for reasoning about experiences in SULTAN; their inclusion would enhance the analysis model.

2.3.3.2 OpenPrivacy and Sierra

OpenPrivacy [Labalme et al, 2001] is an open platform for the creation, sharing, and calculation of reputation based trust. A key element of its approach is that it protects the privacy of the user while providing enhanced services to that user such as item selection or search result filtering. In addition, OpenPrivacy enables the migration of reputation information across disparately managed communities such as eBay, Amazon, and Slashdot.

Privacy is maintained by using a pseudonym for a user, which is called a *nym*. These pseudonyms are represented through public-key pairs. The OpenPrivacy *Nym Service* provides the ability to create 'parent' *nym*s. It is possible to create 'child' and 'grand-child' *nym*s from a parent *nym*. This hierarchical *nym* set cannot be traced back to its 'parent' by a third party, but a 'parent' can provide an anonymous certificate that proves that a 'child' *nym* was created from a 'parent'.

OpenPrivacy's *Reputation Services* "provide a standard opinion and reputation framework that can be used by any community, supporting an unlimited number of mechanisms to create, use and calculate results from accumulated opinions, bias and reputations." [Labalme et al, 2001]. A reference is a URI pointer to a person, place, object, and so on, for example a reference to a work colleague. OpenPrivacy's users provide an opinion with respect to a reference so that many opinions can exist about the work colleague. Calculations are made using the Reputation Calculation Engine (RCE). OpenPrivacy gathers the opinion of users with respect to a given reference (the work colleague) and then calculates a reputation value based on this collective opinion. Reputation is therefore the aggregated sum of opinion, one or more, for a single reference, in other words the work colleague. However, bias is an accumulation of opinions that represent the views of a single user. Bias may be divided into groups of opinion based on political, demographical, and so on. Bias can be used by an RCE in the calculation process so that a user can benefit from the bias of another user that they hold in high regard.

OpenPrivacy has created Sierra [Sierra], which is an implementation of a Reputation Management System (RMS). Sierra has a built in RCE and OpenPrivacy state that it frees developers from the concerns of issues relating to communications protocols and framework design.

Analysis

An analysis of OpenPrivacy reveals that there are many positive aspects to the way in which it manages trust. The focus of OpenPrivacy is based on providing privacy to its users. Such as mechanism, if successful, would prove very useful in allowing users to annotate work colleagues with trust data on an anonymous basis. OpenPrivacy allows a user to annotate other entities with their opinion, which are used in an aggregation based trust calculation to form a reputation for that entity.

However, OpenPrivacy has some weaknesses and open questions regarding its operation and completeness. The internal operation of the RCE is not detailed and it is not clear as to how text based opinions are aggregated. Annotation based on a clear scale may result in calculation mechanism that is clearly identifiable. The state of OpenPrivacy's development, implementation, and integration status is at times also

unclear. The most significant literature [Labalme et al, 2001] is a white paper that describes the OpenPrivacy platform but there are no subsequent publications available that detail experiments or provide critical analysis.

2.3.3.3 TRELLIS

The TRELLIS [Gil et al, 2002] project enables users to express their trust in a source, and in that sources statement(s), so that an individual's trust can be combined into an overall assessment of trust in that source. TRELLIS also enables users to annotate how they analyse and use information when making a decision. In this way an analyst can review information from sources and annotate it as contradictory or complimentary, provide their opinion on what they believe, and attribute trust to that source. TRELLIS is based heavily in military use case scenarios (for example military planning or intelligence analysis) but the ideas are applicable across a broad set of use case scenarios (for example genealogy).

TRELLIS provides a semantic annotation language based on RDF, XML schema, and OWL to annotate information analysis. The language uses seven basic components: (i) *statement*, (ii) *construct*, (iii) *source description*, (iv) *reliability qualifier*, (v) *credibility qualifier*, (vi) *likelihood qualifier*, and (vii) a *reason*. The language can be extended as required. In operation a *statement* would have associated with it a *source description* and *reliability*, *credibility*, and *likelihood qualifiers*. These components are used to make *units*. The basic structure of a units, as described in [Gil et al, 2002], is shown in Figure 2-2:

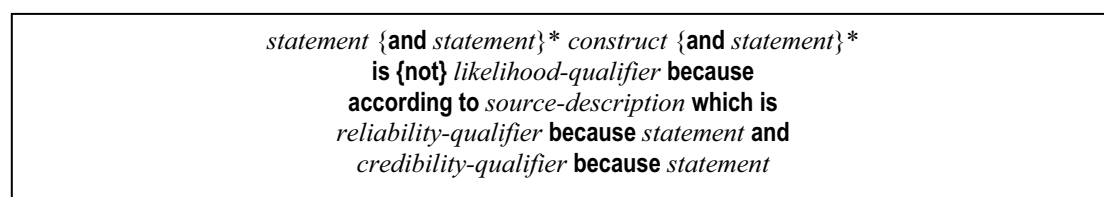


Figure 2-2 Structure of a TRELLIS unit

An analysis of a topic is composed of units, which can be linked as sub-units of each other to create a tree structure. An analysis is carried out by the analyst through the TRELLIS user interface. This interface allows the analyst to search for web resources, which are indexed by TRELLIS. Each resource can have meta-data that describes the resource (title, owner, published, and so on). The analyst then creates *statements* (that

may paraphrase the resource) adding *qualifiers* and building *units* from composed *statements*. Such an analysis is annotated by TRELIS in several mark-up languages. An analysis can be shared so that another analyst can view and import annotations as required.

An example of an analysis, based on a single unit, can be described using the example provided in [Gil et al, 2002], as shown in Figure 2-3:

water temperature unsustainable for SDV divers
is elaborated in
average March water temperature is 55-60 degrees
and
platoon requires minimum water temperature of 65 degrees
according to source
Cmdr Smith ***which is***
completely reliable (A)
because Cmdr Smith has 15 years experience with JSOC
and
probably true
because Cmdr Smith has been platoon cmdr for 3 years

Figure 2-3 Example of a TRELIS *unit*

In Figure 2-3 an analyst is trying to determine whether a military dive operation can take place. The conclusion of the analysis is that the water temperature is unsustainable for military divers. This conclusion was derived from two *statements*, which were provided by a *completely reliable* source and *probably true* due to the source's experience. It is this analysis process that is annotated and shared by TRELIS.

TRELIS provides the ability to derive an assessment for each source referenced in an analysis, specific to a topic, which is based on the values for *reliability* and *credibility* that were attributed to the source by one or more analysts. Overall ratings for source-*statement* pairs can also be provided based on the derived assessment of the source and the number of times that the statement was used, not used, or marked as tainted. An analyst can use TRELIS to search for sources of specific topics, which returns a ranking of sources based on their overall ratings. In addition, the analysis can view the details of a source's ratings.

Analysis

An analysis of TRELIS shows that it has many positive aspects. Most notable is that analysts attribute trust values based on a multi-faceted approach. The authors clearly state in [Gil et al, 2002] that *reliability* is not the same as *credibility*. The impact of using these two trust concepts to describe trust enables analysts to state a source's *reliability* independent of its *credibility*. A source may be very *reliable* but may not be *credible* across all domains. TRELIS also provides the ability to attribute trust specific to a domain, for example in Figure 2-3 Commander Smith is completely *reliable* in terms of military operations. This ability to annotate trust towards a specific domain means that trust calculations can be made specific to that domain. TRELIS also provides such a trust calculation mechanism that can be used to calculate an overall rating for a source, and a source-*statement* pairing. The rating can be used in the future assessment of information. In addition, calculations are domain specific, which could prove more useful than a general calculation.

2.3.3.4 Fidelis

Fidelis [Yao, 2003] provides a policy driven trust management framework for building secure, trust-oriented distributed applications. Fidelis allows beliefs, or assertions, between a trustor and a subject to freely pass between entities in the form of trust statements. These trust statements are represented as a public key credential, signed by the trustor. Fidelis might have been described in the credential trust management system section if it was not for the presence of an internal trust annotation mechanism, a trust calculation mechanism, and a use case scenario specific to online marketplace.

Fidelis is based on the use of credentials and policies, which are separate to each other. Fidelis considers credentials as static data structures that are simply assertions that have no processing semantics. Policies provide the semantics for such credentials; in essence policies interpret the assertions with locally defined semantics. Policies are described as local only to the person who specifies and manages them.

Fidelis defines trust as ‘a set of assertions that a principle held with regard to another principle’ [Yao, 2003]. A principal can be an individual, a group of individuals, or a group with a threshold value. A statement is “a template of an assertion, which may

be instantiated to create trust instances” [Yao, 2003]. A trust instance represents the “specific trust that a principal has with respect to another principal” [Yao, 2003], an example of such an instance is presented below in Figure 2-4.

C.credit_rating(“Company_A”, 4) A → B

Figure 2-4 Example Fidelis Trust Instance

In Figure 2-4, principal C has defined the credit_rating statement. The trust instance is issued by A to B and states that Company_A has a credit_rating of 4. Many of such trust instances can exist in a trust network, which is created when the instances are shared among principles.

The Fidelis Policy Language (PFL) is a central part of Yao’s research. There are two types of policy that can be defined; a Trust Policy and an Action Policy. A Trust Policy specifies conditions and rules for constructing new trust instances. Action Policies relate actions with trust instances. An Action Policy for the trust instance in Figure 2-4 might grant the right to purchase an item from Company_A if A rates Company_A’s credit_rating as 3 or greater. In this case the Action Policy would grant the action.

Fidelis research [Yao, 2002] has provided a case study for an online marketplace in which members can annotate each other, similar in operation to Figure 2-4. A customer in the online marketplace can buy an item and annotate the seller with their opinion (trust information) on a scale of 1 to 5. A trust calculation mechanism that uses a ratings aggregator can retrieve instances from the trust network, specific to a seller, and return the average rating for that seller to prospective customers. An Action Policy can be created that only allows a purchase to be made if the seller has an average feedback of 4 or more. In addition, a customer can delegate the purchases to the virtual marketplace. In doing so the identity of the buyer is hidden from the seller. This delegation mechanism respects the local Action Policies of the customer.

Analysis

Many positive aspects can be found in Fidelis approach to trust management. Fidelis provides mechanisms for the annotation of users with opinion based trust data and the subsequent calculation of an overall trust value based on multiple sources of trust data. Fidelis provides specialised trust annotation depending on its use case. In the online marketplace the specialisation was based on opinion, whereas in [Yao, 2002] a World Wide Web specialisation could be based on evidence such as an IP address or authentication result. In addition, Fidelis provides a mechanism that allows a user to delegate certain tasks to another user. Specialised trust annotation and calculation mechanisms are strongly desired in the research presented in this thesis.

2.3.3.5 FOAF Extended; TrustBot and TrustMail

Friend-Of-A-Friend (FOAF) [Dumbill et al, 2002] is a project that utilises a RDF vocabulary that users can use to describe information about herself and her friends, which includes statements that can be used to build a web of acquaintances. In [Golbeck et al, 2003], FOAF is extended to include information about trust. Each FOAF member annotates other members that they know with trust data. These relationships are used to build a larger network between users on the Semantic Web.

The trust data is a property that explicitly states the trust level associated with a friend on a scale of 1-9; absolute distrust to absolute trust. This trust can be made specific to a domain, such as medicine, or can be generalised. Figure 2-5 presents a FOAF extension where Bob has annotated Dan with a high level of trust, specific to medicine.

```
<Person rdf:ID="Bob">
  <mbox rdf:resource="mailto:joe@example.com"/>

  <trustsHighlyRe>
    <TrustsRegarding>
      <trustsPerson rdf:resource="#Dan"/>
      <trustsOnSubject
        rdf:resource="http://example.com/ont#Medicine"/>
    </TrustsRegarding>
  </trustsHighlyRe>
</Person>
```

Figure 2-5 Example of FOAF Extension

In [Golbeck et al, 2003] a trust calculation algorithm is presented with two applications to illustrate algorithms operation; TrustBot and TrustMail. The algorithm operates over a distributed network of trust data and is used to calculate an overall trust value for a user in the network. A weighted trust calculation regarding immediately unknown users can be inferred via known users, which can be seen as recommendations. TrustBot is an Internet Relay Chat (IRC) “bot” that can provide recommendations to users based on the network that it builds at runtime from a collection of distributed sources. Users can annotate other users so that the algorithm can calculate a weighted average as well as maximum and minimum path lengths and maximum and minimum capacity paths. Users can view the trust levels for other users on a general level or topic specific. TrustMail is an email client that operates in a similar way to TrustBot. TrustMail provides an inline trust reputation rating for an email message, which can also be general or topic specific.

These algorithms have been analysed in [Golbeck et al, 2004] and have been shown to be 60% accurate when the population of the network is made of 90% ‘good’ nodes, which is much better than most other reputation systems.

Analysis

An analysis of the trust based extension to FOAF shows many positive aspects. Using OWL to represent trust data is very useful for sharing trust data in heterogeneous networks. Allowing trust annotation to be made in general and specific to topics is more useful to users across a set of different applications. The trust calculation algorithm enables the inference of a recommendation where no direct relationship exists. These recommendations prove quite accurate in comparison to other recommender systems.

2.3.3.6 Other Notable Internal Approaches

Ntropi [Rahman, 2005] provides a decentralised algorithm for weighting information from someone else (secondary information) with respect to the requestor’s previous encounters with the secondary data provider. In this way the trust data from a provider who consistently over exaggerates will be offset to account for the over exaggeration.

Poblano [Chen et al, 2000] is a distributed trust management system based on the JXTA [JXTA] platform. Poblano allows its users to state their opinion of other users, which is shared and evaluated in order to calculate a trust value. Poblano's core focus is on the development of algorithms for propagating, updating, and retrieving trust data. However, the ability to annotate and calculate trust is also present. In Poblano, trust is specified on a scale of -1 to 4; distrust, ignore, minimal, average, good, complete, respectively. Trust relationships can be built between two users and with respect to a specific domain. In this way it is possible to state that user A trusts user B at the '4' (complete) level of trust with respect to cooking recipes. Searching for trust data is based on a peer-to-peer algorithm and can be made with respect to a specific domain. Threshold values can be set that specify the level of trust that is necessary before co-operation is possible between a set of users. Poblano is of interest as it provides an opinion based trust annotation mechanism that can be made specific to a particular application domain, which uses a set of scaled values to enable users to describe their trust in other users. Such an approach is desirable in this thesis.

2.3.3.7 Overall Analysis of Internal Trust Management

The internal trust management systems provide mechanisms for annotating an entity with trust data and for calculating a trust value based on that data. This results in a system that can automatically gather data and provide an overall trust value to help guide the user, which only requires minimal input from the user. The definitions of trust that have been used are both single-faceted and multi-faceted. There is however quite a diverse range of trust concepts used to describe trust, which may make interoperation between trust management systems challenging. For example, TRELLIS uses the trust concepts reliability and credibility, whereas OpenPrivacy uses reputation. The reviewed systems allow general and domain specific annotation and calculations that can take advantage of specific properties of a domain. For example, SULTAN, TRELLIS, and Fidelis provide domain specific trust annotation. In particular instances providing an overall trust value based on a particular domain could be perceived as being of more use than a trust value calculated using a non-domain specific approach. Golbeck's calculation algorithm can also infer a trust value where no direct relationship exists between the trust data requestor and provider. This can be quite useful in situations where large numbers of users exist and users have limited numbers of defined relationships. However, there may be more complexity

and greater time delays in retrieving trust data in this way. The survey of internal trust management systems has also indicated the benefits of using an ontology to represent a model of trust. TRELIS uses an ontology that can be reasoned over, easily shared, and extended, which facilitates the functions of the trust management systems. Finally, the ability to delegate trust decisions, as provided by SULTAN and Fidelis, is an important consideration. A user can delegate decisions where they feel that the decision can be made by a user or organisation that has more experience or competency. This reflects the real world where an employee may delegate the management of their pensions to a board of directors.

2.3.4 Online Community Trust Management Systems

The trust management systems described in the previous two sub-sections are systems that have been designed to provide trust services. In this sub-section several online communities are presented. These communities have taken advantage of trust management, and properties of the internal trust management systems, in the provision of their core operational services.

2.3.4.1 Advogato

The Advogato [Levien et al, 1998] project is a trust management system based on certificates. However, its primary research work is based on a trust algorithm that is highly attack resistant. The use case for such a trust algorithm is an online community of free software developers. Due to the focus on a highly attack resistant trust algorithm in an online community use case scenario the discussion of Advogato is situated in this section and not credential based trust management.

At the Advogato website, www.advogato.org, developers annotate each other with trust data that includes; *apprentice*, *journeyer*, and *master*. The Advogato trust algorithm is then used to provide an overall trust value for a community member with respect to other community members. Access to the features of the website is based on the trust level of the member. A *master* developer may be able to post and edit messages, whereas an *apprentice* developer may only be able to read messages.

Analysis

The algorithm for calculating trust is highly attack resistant. The system cuts out portions of the network that a seed node identifies as 'bad'. The algorithm removes such 'bad' nodes, as well as any nodes that certify such 'bad' nodes. In this way a calculation is primarily based on 'good' nodes, which results in the overall network remaining secure. This strength in the face of attack can also be found in the EigenTrust system [Kamvar et al, 2003], which also calculates a global trust value from trust data received from peers.

2.3.4.2 eBay and Amazon

eBay [eBay] is an online auction site that allows buyers and sellers annotate each other with trust data, in the form of feedback, after a transaction has been made. The trust annotation can be negative (-1), neutral (0), or positive (+1). In addition, a short free text comment can be provided as part of the feedback. Each eBay community member can see the feedback of every other eBay member.

The present eBay feedback system offers information to members in three ways. Firstly, an overall percentage of positive feedback is immediately presented alongside a member's identity. Currently, this is calculated by dividing the sum total of all feedback scores for a member by the sum total of all positive and neutral scores for that member. If, for example, a member has complete 100 transactions and has received 98 positive, 1 neutral, and 1 negative feedback score then that member will a positive feedback score of 99%. Secondly, the feedback scores for a user are also presented to the user over a timeline of twelve months, at one, six and twelve month intervals. In addition to this timeline the feedback scores are also presented by category; negative, neutral, and positive. Finally, the feedback comments that members can optionally add as feedback can be reviewed by another member.

Amazon.com [Amazon] is an online retail site through which buyers can purchase products. The Amazon brand has several other business entities offering a range of service offers; including Amazon Auctions [Amazon Auctions], Amazon zShops [Amazon zShops], and Amazon Marketplace [Amazon Marketplace]. Amazon offers a similar feedback system to eBay's for new and used items for Amazon Auctions, zShops, and MarketPlace in which feedback can be left after a transaction.

Feedback is provided on a scale of one to five stars and free text comments can be made. One and two star feedback is considered negative, three stars are neutral, and four or five stars are positive. The feedback information is presented in a similar fashion to eBay through the use of; (i) an initial positive rating indicator, (ii) feedback over a timeline, and (iii) free text comments. However, there are differences in how Amazon and eBay presents and calculates trust information. Firstly, Amazon calculates trust by only including ratings left by the buyer and not the buyer and seller. Secondly, Amazon's timeline presents the percentage of positive, neutral, and negative feedback scores categorised on a thirty day, ninety day, three hundred and sixty five day, and lifetime basis. Amazon and eBay both present free text feedback.

Analysis

An analysis of the trust management systems that eBay and Amazon provide indicates many strong points. Firstly, the trust annotation mechanism is simple and clear. Secondly, the trust calculations provide an understandable trust value to the users and members of both systems. However, the trust annotation and trust calculations are based on a single-faceted model of trust; reputation. This form of simplicity may be the reason behind the success of these trust management systems. However, it may also provide a model of trust that can not capture the wide and varied subjective views of trust as exhibited by an individual across a large and broad population. The research experiments and analysis presented in this thesis argues for the latter.

2.3.4.3 Slashdot and Epinions

Slashdot [Slashdot] is an online news service and forum for technology. Articles are submitted by users, appraised by editors, and comments can be posted for each article. When a user leaves a comment a set of moderators can increase or decrease a score attached to it. Moderators are chosen using an algorithm that randomly selects online, regular, long term Slashdot members who are willing to participate. A moderator can increase a comment score to a maximum of five (good) points and a minimum of minus one point (bad). The owner of a comment receives karma points if the score for that comment is positive. As a user gains karma points privileges become available. High levels of karma can lead to comment scores starting at a higher level. If a users karma points deteriorate then penalties can be applied, for example privileges could be revoked. This process is designed to promote quality and improve user experience.

Epinions [Epinions], which is owned by eBay, is a website that allows readers to author review of products, service, and so on. A user provides a text based review of a product and accompanies it with a rating on scale of one to five stars. One star is considered the worst and five is considered the best. The review includes separate sections for describing the pros and the cons of the item, and a 'bottom line' for conclusions. The reviews contain domain specific attributes that vary depending on the product reviewed. For example, a car review can include 'seat comfort' and 'roominess', which can be attributed with a rank on a scale of one to five. In this way Epinions users can write domain specific review about a product, service, and so on.

Epinions also allows users to rate a review of a five point scale; *off topic, not helpful, helpful, somewhat helpful, helpful, or very helpful*. A user can also decide to trust or block a reviewer. Epinions builds a web of trust based on the reviewers that a user has decided to trust. The Epinions system then presents future reviews to the user based on the users own web of trust and ratings. As a reviewer receives more positive ratings they can attain various statuses; *advisor, top reviewer, and category lead*. The reviews from an *advisor* have greater weight than an ordinary reviewer, and there is the ability to rate the review at another level; *most helpful*. The *top reviewer* reviews have additional weight and priority in certain circumstances. Finally, the *category lead* enjoys the benefits of both the *advisor* and *top reviewer* as well as even greater weighting. In addition, the *category lead* can assign *advisor* and *top reviewer* to certain categories. Furthermore, the *category lead* can add new products and services for review in Epinions.

Analysis

An analysis of the trust management systems provided by Slashdot and Epinions illustrates many strong points. Slashdot provides a basic policy, or filtering, service where user comments can be selected for viewing based on trust, which is a useful service for users. Epinions provides the ability to specialise reviews based on domain specific attributed. Domain specialisation within trust is a key factor in the research work presented in this thesis. However, like eBay and Amazon both Slashdot and Epinions may suffer by using a single-faceted model of trust.

2.3.4.4 Overall Analysis of Online Community Trust Management

The current online communities trust systems provide the incentive to build and maintain a solid online reputation. They handle large numbers of transactions and provide a useful trust management service to its users while doing so. The annotation mechanisms provided by the systems described in this section are clear and easy to understand and complete. The simplicity of their trust calculations provides a buyer with an understandable indicator that can help guide the decision to purchase a product or service, or believe the statements of a member. Such a clear trust annotation mechanism and simple trust calculation algorithm are desired in this research work presented in this thesis.

Online communities also provide a rich data set on which trust can be calculated. Yet unscrupulous community members can take advantage of online trust mechanisms for fraudulent and malicious purposes. However, trust algorithms presented in Advogato and EigenTrust have been shown to be highly attack resistant.

2.3.5 Online Social Network Trust Management Systems

Social networks capture the relationships that a member has with other members in an online community. Social networks such as Bebo [Bebo] and MySpace [MySpace] allows users to attach their friends to their online profile that enables the group of friends to communicate and inter-relate. A trust annotation mechanism in a social network could have defined paths that exist from the provider to the requestor of trust data, which is rare in large online communities such as eBay and Amazon where the odds of knowing a buyer directly, or through someone, are very low.

2.3.5.1 FilmTrust

In [Golbeck et al, 2006], Golbeck advances the research work presented in [Golbeck et al, 2003] and [Golbeck et al, 2004] to provide movie recommendations calculated through a Semantic Web based social network. Once again the FOAF extension, now referred to as the FOAF Trust Module, is used to build a network of movie raters. The trust calculation used is TidalTrust, which is based on earlier research presented in [Golbeck et al, 2003] and [Golbeck et al, 2004]. TidalTrust provides predictive ratings personalised to a user. Personalisation, in this instance, is based on the path that exists from the requestor of trust data to the provider of trust data, which is used

in the calculation algorithm. In FilmTrust a user can add friends and annotate them with trust data specific to movies. These annotations are kept private amongst users of FilmTrust. A user can then rate a film on a scale of one half stars to four stars, with half star increments. In addition, a text review can accompany the rating. Two ratings are calculated for each movie; a simple average of all ratings and a weighted average rating based on TidalTrust's inferred trust values. If a user seeks a film that her friends have not rated then the system will seek a path to other users who have rated the film. This is accomplished in a friend of a friend like fashion in that a friend's friend may have rated the film. The TidalTrust calculation algorithm can discount ratings based on the distance from the requesting user to the film rater.

The conclusion of accuracy experiments in [Golbeck et al, 2006] shows that when a user's rating of a movie is different to the average rating for that movie, it is likely that the recommended rating will more accurately reflect the user's tastes. However, these experiments also show that the accuracy of trust based predictive ratings is significantly better than the accuracy achieved through simple averaging calculations.

Analysis

An analysis of FilmTrust reveals many strong points. The accuracy of the TidalTrust algorithm in providing recommendations is admirable. The use of OWL to represent and share trust data has benefits, which have been previously stated. The clear trust annotation mechanism is key to a user providing useful trust data. Personalisation is based on a recognisable path from a trust data requestor to provider. Although this provides a clear map as to where trust data originated it does not provide a personalisation experience that leverages a personalised model of trust.

2.3.5.2 Analysis of Social Network Trust Management

Trust management systems that operate within a social network can have the benefits of an online community and its member's friendships and relationships. Social network trust management systems can provide a trust calculation mechanism whose trust data can be traced from its source to its destination, which may give more weighting to the trust values as perceived by the member. However, searching for trust data and discovering defined paths may be more complex and have a greater time delay than retrieving trust data in internal trust management systems.

2.3.6 Comparison Framework

It is important to note that the following discussion is related to only the reviewed trust management systems presented in Figure 2-6, where the focus is on the internal, online community, and social network based trust managements systems. As trust is determined outside of the system in most credential based trust management systems then core factors such as trust annotation and trust calculation are not applicable. Therefore, all credential based trust management systems, with the exception of REFEREE, are not included in the comparison framework as trust annotation and trust calculation are considered key requirements for a trust management system.

		REFEREE	SULTAN	OpenPrivacy	TRELLIS	Fidelis	FOAF extended	Advogato	eBay	Amazon	Slashdot	Epinions	FilmTrust
Model of Trust	Single-Faceted	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓
	Multi-Faceted				✓								
	Personalisable												
	Specialisable by developer	✓	✓		✓	✓	✓					✓	✓
	Specialisable by user												
Trust Annotation	Opinion	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Evidence	✓	✓		✓	✓						✓	
Trust Calculation	Simple	✓	✓	✓	✓	✓			✓	✓	✓	✓	
	Advanced						✓	✓					✓
Policy	Internal	✓	✓	✓		✓					✓		
	Separate												
	Not Used				✓		✓	✓	✓	✓		✓	✓
Architecture	Centralised	✓	✓		✓			✓	✓	✓	✓		
	Distributed			✓		✓	✓					✓	✓
Trust Representation	XML				✓								
	OWL				✓		✓						✓
	Other	✓	✓	✓		✓		✓	✓	✓	✓	✓	

Figure 2-6 Comparison Framework Chart

Figure 2-6 presents the comparison framework chart that will be used to compare the applicable trust management systems. The analysis of the comparison framework is conducted individually for each of the six factors in the following sub-sections. In addition, commonalities across trust management systems will be explored by combining some of the main factors together.

2.3.6.1 Model of Trust

Single-Faceted Approaches

As can be seen in Figure 2-6 most models of trust are based on a single-faceted approach. In single-faceted approaches it is reputation that is the most dominate synonym for trust.

Multi-Faceted Approaches

The literature review shows that there is a wide range of synonyms for trust that are used to define and describe trust. However, only TRELIS, which uses reliability and credibility to describe trust, implements a multi-faceted approach to trust management. Yet, TRELIS also uses these trust concepts independently, which is important to consider as it is possible for an information source to be very reliable in one area of expertise but not credible in other areas.

Personalisation

The literature review asserted, across all associated disciplines, that trust is a subjective entity. However, models of trust and trust management systems to date have not provided any personalisation mechanism that can capture this subjectivity, which is found within trust. Golbeck refers to personalisation within her FilmTrust project, but personalisation in FilmTrust is based on a defined path from a user seeking trust data to the provider of that trust data. Personalisation, in this thesis, is defined as the ability to capture both the wide and diverse range of views of trust that exist across a large and broad population as well as the subjectivity found in trust at the individual level. Therefore, in FilmTrust the model of trust is not personalised, as defined in this thesis. According to this definition personalisation within a model of trust is not found anywhere in the reviewed trust management systems.

Specialisation

In the selected trust management systems specialisation within a model of trust is limited to specialisation that is provided by the developer. This sort of specialisation is found predominantly in internal trust management systems and FilmTrust. No mechanism exists in any of the reviewed trust management system that allows an end user to specialise a model of trust towards a given domain.

2.3.6.2 Trust Annotation

It can be seen in Figure 2-6 that a trust annotation mechanism is provided by every trust management system. All of these trust management systems use only an opinion based approach. A smaller subset of these trust management systems, REFEREE, SULTAN, TRELIS, Fidelis, and Epinions, provides both an opinion based approach and an evidence based approach. An evidence based approach is observed most often in internal trust management systems. However, trust annotation only based on evidence has been used in at least one trust management system across all four categorisations of trust management systems; credential, internal, online community, and social network trust management systems.

2.3.6.3 Trust Calculation

The majority of trust calculation algorithms used by the trust management systems in Figure 2-6 are classified as simple. Only Advogato, FOAF extended, and FilmTrust provide advanced trust calculation algorithms. Most of the online community trust management systems do not provide advanced trust calculation algorithms.

2.3.6.4 Policy

Nearly all the online trust management systems do not use policy. It is Slashdot that provides limited internal policy specification techniques in the form of a filtering mechanism that enables a user to select comments based on the level of trust that they have been attributed.

Policy is heavily present, internally, in credential based trust management systems and also in the subsequent generation of trust management systems (in other words internal trust management systems). There are policy driven trust management systems, such as SULTAN and Fidelis, that make extended, internal, use of policy but by the generation of online trust management systems policy has become more or less redundant.

Not one of the reviewed trust management systems provides a form of policy that is separate to the trust management system. A separation of concerns allows multiple policy languages and approaches to be used by the trust management system.

In early credential based trust management systems it was policy that was at the core and trust that was external, but by the generation of online trust management systems it is trust that is at the core and policy that is external.

2.3.6.5 Trust Architecture

The architecture of the trust management systems was determined in order to gain some insight into the implementation details of these systems for use in designing the *myTrust* trust management service. Note that the trust management systems are designed to be either centralised or distributed. By a small margin the overall majority of trust management systems are centralised. However, the vast majority of online community architectures are centralised. In addition, a distributed architecture holds a slight majority in internal trust management systems.

2.3.6.6 Trust Representation

The vast majority of reviewed trust management systems do not use XML or OWL as a trust representation format. An ontological approach is found in only more recent trust management systems; TRELIS, FOAF extended, and FilmTrust. However, this is quite a reasonable observation as OWL only became a World Wide Web Consortium (W3C) [W3C] Recommendation in early 2004.

2.3.6.7 Factor Combinations

Trust Model and Trust Annotation

The most common set of attributes that the reviewed systems share is a single-faceted approach to modelling trust and an opinion based trust annotation mechanism. This combination is relatively easy to implement in that using a single-faceted model makes opinion based trust annotation relatively straight forward. In eBay, for example, it is *reputation* that is used in the single-faceted model, and subsequently it is easy to provide an opinion system that asked for positive or negative annotation.

In most systems where evidence based trust annotation is provided then specialisation (by developer) is also present. This combination tends to use a single-faceted approach. However, TRELIS uses a multi-faceted and specialisable (by developer) model of trust. In addition, its trust annotation is based on both opinion and evidence. This larger combination of a multi-faceted and specialised model of trust that can be

annotated based on opinion and evidence is very useful in that a wide range of application domains could be catered for.

Trust Annotation and Trust Calculation

All trust management systems that use an evidence based approach to trust annotation use a simple trust calculation algorithm. These trust management systems are REFEREE, SULTAN, TRELIS, Fidelis, and Epinions. In addition, all of these trust management systems also provide specialisation (by developer). The use of a simple trust calculation algorithm in these cases may seem counter-intuitive to some as a more complex data set could require more advanced trust calculations. However, the added complexity across multiple specialisation domains would require multiple trust calculation algorithms.

2.4 Summary

This chapter discussed several definitions, views, and ideas of trust as presented by researchers in psychology, sociology, and computer science. This discussion illustrated the general lack of consensus as to the meaning of trust. In addition, the current state of the art in trust management systems was reviewed and analysed. The current state of the art in trust management is totally bereft of personalisation within modelling trust, and only one trust management system uses a multi-faceted approach. A comparison framework was used to illustrate the differences and similarities between the trust management systems. Together the state of the art review of perspectives of trust, trust management systems, and the comparison framework impact the design of the model that this thesis presents, which is described in the next chapter.

3 DESIGN

3.1 Introduction

This chapter discusses the issues and challenges that impact the design of a multi-faceted model of trust that is personalisable and specialisable. The influences from the state of the art on the design of the model of trust and on the design of the trust management service are presented in Section 3.2. Influences from the state of the art focus on the various definitions, notions and views of trust found within the literature review and survey of trust management systems.

The overall framework and its general operation are presented in Section 3.3. It details the multi-faceted model of trust that is personalisable and specialisable and how the developed trust management service operates. The trust management service, *myTrust*, enables users to (i) annotate entities with trust information, (ii) share such trust information, and (iii) calculate recommendations based on trust. The multi-faceted model of trust that is personalisable and specialisable is at the core of *myTrust*, with annotation, sharing, and recommendations all based on this model of trust.

The design of the multi-faceted model of trust that is personalisable and specialisable is introduced in Section 3.4. This model of trust has been designed so that its overall structure is separated across four models. The upper ontology provides a set of trust concepts that are used in the generation of personalised models of trust and are also used to engineer specialised models of trust. The relationships that can exist between the extensible set of trust concepts is governed by the trust meta-model. The meta-model, upper ontology, personalised model, and specialised model are all presented and discussed.

Section 3.5 describes the design of the *myTrust* trust management service, including discussion of: trust data annotation, trust calculation algorithm, and trust policy.

3.2 Influences from the State of the Art

The state of the art discussed in chapter two impacts and influences the design of both the model of trust and *myTrust* trust management system. Section 3.2.1 first discusses the influence of the state of the art on the design and development of the multi-faceted, personalisable, specialisable model of trust. Then in Section 3.2.2 the influences on *myTrust* is discussed.

3.2.1 Influences on Design of Model of Trust

The literature review in the state of the art has influenced the choice of a set of desired properties upon a model of trust. These properties of the model of trust include the ability to:

- Capture the wide and varied views of trust that can exist across a large and broad population,
- Capture the subjectivity of trust at the individual level found within a large and broad population,
- Engineer multiple, specialised, application domain models,
- Build upon current models of trust and have the ability to be extended.

In order to capture the wide and diverse views of trust the model should be able to capture the wide range of synonyms that are used to describe trust. The state of the art has elucidated many definitions, ideas, notions, or views of trust that exist within the trust research community within computer science and areas beyond. Trust concepts used by computer science researchers to describe their view of trust include *belief*, *competency*, *honesty*, *confidence*, *faith*, *credibility*, *reliability*, and *reputation*. The computer science research community that has influenced the choice of trust concepts that are used in the model of trust proposed in this thesis.

The main influence on designing a multi-faceted model of trust is the myriad of trust concepts found in the state of the art. These trust concepts are used to form single-faceted models of trust. A single-faceted model may appeal to some, or many, individuals but it is hypothesised in this thesis that a single-faceted approach cannot capture the wide and varied views of trust that can exist across a large and broad

population. A multi-faceted model of trust, that incorporates many single-faceted models of trust, can be used to capture the wide and varied views of trust that can exist across a large and broad population. In the state of the art Grandison, Shadbolt, and Golbeck have used more than one trust concept when describing trust in their view. However, a multi-faceted approach, alone, is not enough to capture the subjectivity of trust at the individual level found within a large and broad population. Personalisation within the multi-faceted model of trust is required in order to do this.

Dieter Gollmann stated in his keynote speech at Policy 2005 that “trust is a fashionable but overloaded term with lots of intertwined meanings” [Gollmann, 2005]. Gollmann states that the trust terms, or trust concepts as they are referred to in this thesis, are intertwined. It is asserted in this thesis that trust concepts can be intertwined, or related to each other in some way. This thesis proposes a trust meta-model to capture these intertwined relationships that can exist between trust concepts in order to capture the individual subjectivity of trust. Thus, a multi-faceted model of trust and the relationships between trust concepts are used to provide a model of trust that can capture an individual’s subjective view of trust.

In the state of the art trust is described as being held with respect to some situation or context by at least [Marsh, 1994], [McKnight & Chervany, 1996], [Gambetta, 1998], [Grandison & Sloman, 2000], [Golbeck et al, 2003], and [Abdul-Rahman, 2005]. Such research has influenced the need for specialisation within the multi-faceted and personalisable model of trust. There are relatively limitless situations in which trust can be specialised. Since trust is used with respect to different situations then a model of trust must also be able to reflect such situations in order to provide a useful trust service. Therefore, it is an assertion of this thesis that the multi-faceted and personalisable model of trust must be able to be specialisable in order to capture a range of application domains and in order to reflect the application domain’s classes, properties, and relationships. This is defined as specialisation in this thesis.

Ontological approaches have been used in the state of the art by [Gil et al, 2002] and across Golbeck’s research to represent trust. The model of trust proposed in this thesis uses an ontological approach as ontology languages, such as OWL, can be used to provide an accurate reflection of the trust concepts and relationships in a sharable and

understandable format that can be reasoned about. In addition, the benefits that OWL provides such as extendibility, reusability, mapping, and semantics were key in the decision to choose OWL. Furthermore, the successful use of OWL by Gil and Golbeck to represent trust in their research added additional confidence.

3.2.2 Influences on Design of *myTrust*

It is important to note that privacy and attack resistant trust algorithms can be found in the reviewed trust management systems but these concerns are not in the scope of the research question posed in this thesis. However, it is also noted that privacy and attack resistance are highly regarded virtues and could be considered as future work.

The survey of trust management systems in the state of the art has influenced the design of *myTrust*. The comparison framework (see Figure 2-6) provided six factors that were used to analyse each trust management system. Every trust management system analysed made use of a model of trust, a trust annotation mechanism, and a trust calculation algorithm. Therefore, these could be considered as the core features of a trust management system, and this is true for *myTrust*. The influences on the model of trust and trust representation were discussed in the previous sub-section, so this section focuses on trust annotation, trust calculation, and policy. Architectural influences will not be discussed as the design of *myTrust* is distributed.

A trust annotation mechanism is found in every trust management systems selected in the comparison framework. It is an opinion based approach that every one of these systems utilises. Therefore, *myTrust* will also provide opinion based trust annotation. In addition to an opinion based trust annotation mechanism there are five systems, REFEREE, SULTAN, TRELIS, Fidelis, and Epinions that also provide an evidence based trust annotation mechanism. It is interesting to note that all five systems also have some level of specialisation within their model of trust. This correlation makes sense as a specialised model of trust could reflect complexities in an application domain that an opinion based annotation mechanism cannot capture. Thus, *myTrust* has been designed so that it provides trust annotation mechanisms that can cope with both opinion and evidence based trust data.

The majority of trust calculation algorithms used by the reviewed trust management systems are relatively simple. Online communities like eBay, Amazon, Slashdot, and Epinions tend to use simple trust calculation algorithms when aggregating opinion based trust data. Their simplicity and clarity may be of great benefit to the community members, and so *myTrust* will also support simple trust calculation algorithms. During the implementation of a prototype trust management system, the author of this thesis found that the complexity of a Web Services domain specific model of trust required an advanced trust calculation algorithm. Such an advanced algorithm was needed to handle the evidence associated with an instance of a Web Service in order to provide a trust based recommendations. Thus, *myTrust* will also support advanced trust calculation algorithms. *myTrust* is designed to support both simple and advanced trust calculation algorithms as it is not known in advance whether *myTrust* will be using opinion or evidence based trust data to calculate a trust value in order to provide a trust based recommendation.

Most of the reviewed credential based trust management systems use policy to work with trust annotation data and trust values to enable a user to specify rules that control access, selection, and so on. Some trust management system use a basic form of policy, whereas others, such as Vigil and IBM's Trust Establishment Framework adopt the more advanced approach of Role Based Access Control (RBAC). Policy is not intrinsically part of *myTrust*. Instead, it is shown in this thesis how *myTrust* provides trust services to the more sophisticated Community Based Policy Management (CBPM) in order to illustrate the usefulness of such a combination in providing dynamic and flexible management.

The key difference between all the reviewed trust management systems and *myTrust* is that *myTrust* carries out trust annotation and calculation functions using a multi-faceted model of trust that is personalisable and specialisable. In this way, these functions are based on a model of trust that can capture the wide and diverse views of trust found across a broad population as well as an individual's subjective view of trust. Furthermore, these functions are carried out with respect to specific application domains by use of a specialised trust model. This level of broad personalisation and domain specialisation within trust is not found anywhere in the current state of the art when modelling trust.

3.3 Overall Framework

The upper ontology, meta-model, personalised models of trust, and specialised models of trust are all used in an overall framework that enables the trust calculation to provide trust based recommendations. This section describes this overall framework and illustrates the roles of these models, where trust calculations are made, and where policy is specified and utilised.

3.3.1 General Operation

At the core of this thesis is the multi-faceted model of trust that is personalisable and specialisable. A framework has been built around this model (see Figure 3-1) to leverage its ability to provide personalised recommendations across multiple application domains. The upper ontology and meta-model are developed in Protégé [Musen et al, 1993] and both are used to support personalisation and specialisation. The upper ontology can be extended, but the meta-model is static and is therefore not subject to extension.

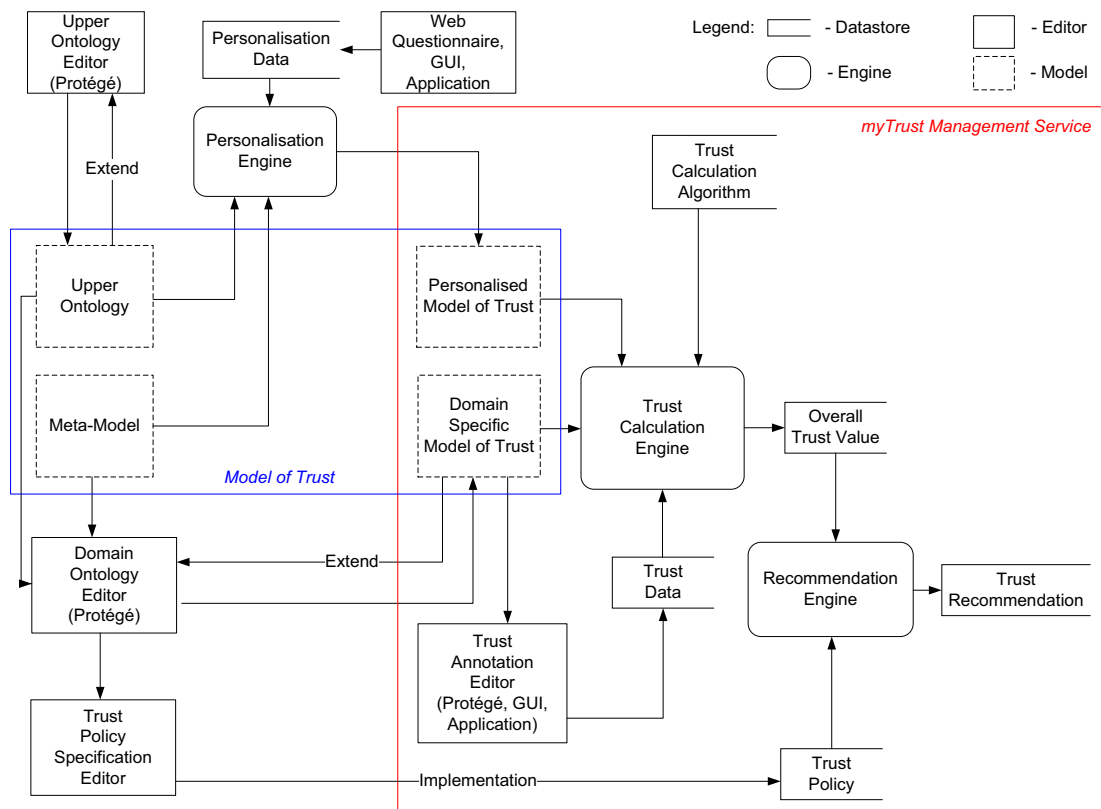


Figure 3-1 Overall Framework

The multi-faceted model of trust that is personalisable and specialisable is marked as *Model of Trust* and outlined in blue in Figure 3-1. The area outlined in red in Figure

3-1 is marked as *myTrust Management Service* and it utilises a personalised model of trust, a domain specific model of trust, associated trust data, and trust policy to provide trust based recommendations to applications that operate in Internet environments. Note that the *Model of Trust* is separate to, but used by, the *myTrust Management Service*.

The Personalisation Engine uses the upper ontology, meta-model, and personalisation data to generate a personalised model of trust using the personalisation algorithm (see Section 3.4.3). Personalisation data is the set of relationships that exists between trust concepts, as specified by a given user. Personalisation data can be gathered through a web based questionnaire, via a GUI, or through an application.

A domain expert can use an ontology editor to engineer a domain specific model from the upper ontology and meta-model. Domain specific models can be re-engineered as necessary. A domain specific model is used by the Trust Annotation Editor as the basis for capturing trust data. Trust annotation can be carried out via an ontology editor (for example Protégé), a GUI, or through an application.

The generic Trust Calculation Engine at the centre of *myTrust Management Service* uses one of many pluggable trust calculation algorithms to provide a personalised and domain specific trust value, which is referred to as an overall trust value. It is the task of a developer to create a trust calculation algorithm that reflects the domain specific model of trust. However, a *myTrust* user could create and share a trust calculation algorithm for substitution into the Trust Calculation Engine, which would be used to calculate an overall trust value.

The overall trust value is reconciled with a trust policy. Trust policies have a condition that states the minimum level of trust that is required to provide a positive recommendation for a given event. A trust policy can optionally include domain specific information, which is extracted from the domain specific model. The Recommendation Engine reconciles the overall trust value with a trust policy to provide a trust based recommendation, which is positive or negative.

3.4 The Multi-faceted Model of Trust that is Personalisable and Specialisable

The model of trust proposed in this thesis mirrors the myriad of human views of trust and its relationships, which is modelled through four distinct models that were developed using the Protégé ontology editor:

- (i) Upper ontology,
- (ii) Meta-model,
- (iii) Domain Specific model of trust,
- (iv) Personalised model of trust.

The upper ontology contains the extensible set of trust concepts that are used to build a multi-faceted model of trust. These trust concepts are classes with no properties. The meta-model governs the relationships that can exist between the trust concepts. The upper ontology and meta-model are separate so as to allow for the independent editing of trust concepts in the upper ontology. Combining the upper ontology with the meta-model enables the engineering of domain specific models of trust and also the generation of personalised models of trust.

A domain specific model is the instantiation of the upper-model and meta-model towards a given application domain. In domain specialisation the trust concepts in the upper ontology are sub-classed and domain specific properties are added. Domain models are kept separate to allow developers to capture and scope a range of domains, which can be used independently in applications.

Personalised models of trust are generated from the upper ontology and meta-model on a per user basis. A personalised model contains the set of relationships that may exist between trust concepts as provided by an individual. These relationships can be used to build a ranking of all the trust concepts (see Section 3.4.3). Therefore, the personalised model contains the trust concepts, the relationships between the concepts, and concept rankings and weightings. *myTrust* uses a personalised model in conjunction with a domain specific model and domain specific trust data in order to provide personalised trust based recommendations that are specific to an application

domain. Having a separate personalised model allows applications to selectively import an individual's personalised model of trust. Separation between personalised and domain models of trust allows a single application to provide personalised trust based recommendations to a range of individuals with respect to many different domains by selectively choosing personalised and domain specific models of trust.

3.4.1 Upper Ontology

The upper ontology, see Figure 3-2, provides a generic set of trust concepts that can be reused to generate personalised, and domain specific, models of trust. The eight trust concepts used in this ontology appear in the state of the art as trust synonyms that have been used for the term trust. It is the upper ontology that enables the model of trust to be multi-faceted. With respect to the model of trust proposed in this thesis the upper ontology is a standard, which has been developed and evaluated over the course of this PhD thesis. Such a standard could potentially be adopted by a group such as the World Wide Web Consortium (W3C), or by the trust management community as a *de facto* standard. The upper ontology is extensible in that more trust concepts can be added (or removed if required) by either a developer or end user.

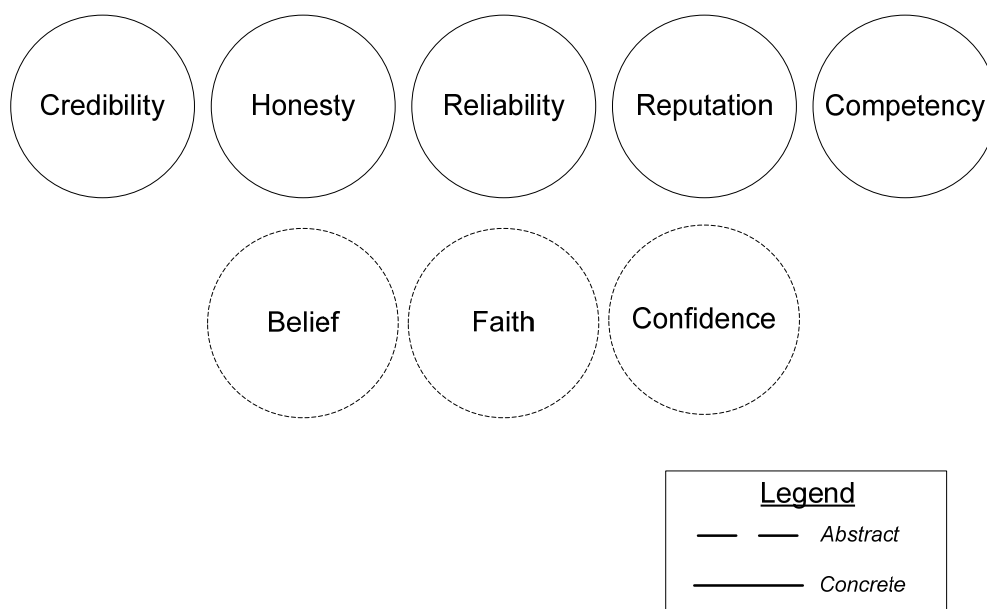


Figure 3-2 Upper Ontology

The OWL based upper ontology is illustrated in Figure 3-2 and within the upper ontology there are two types of trust concepts; *concrete* concepts and *abstract* concepts. The *concrete* concepts are *competency*, *credibility*, *honesty*, *reputation*, and *reliability*. The *abstract* concepts are *belief*, *confidence*, and *faith*. The *concrete* concepts are considered to be more defined and tightly scoped than *abstract* concepts, which are more open to interpretation and are loosely scoped. This categorisation of trust concepts was based on the initial perception of trust concepts as held by the author of this thesis. However, the evaluation chapter provides experimental evidence that supports this categorisation of trust concepts as *abstract* or *concrete*. The significance of using these two types to categorise trust concepts becomes apparent when considering the personalisation, specialisation, and calculation mechanisms in the forthcoming sub-sections.

The strength of the upper ontology is that it empowers a multi-faceted approach to modelling trust. The author acknowledges that the eight trust concepts are limited to trust concepts that have been used in the state of the art and that many more may exist, for example loyalty. In addition, allowing a user to extend the upper ontology beyond the standard would create a disparity between the different instances of upper ontology. This could lead to situations where *myTrust* might have to calculate an overall trust value with incomplete information, for example no loyalty trust data.

Before developing the upper ontology several design issues were addressed. Initially, the biggest issue was finding and choosing the trust concepts that would make up the upper ontology. However, this has been addressed in the state of the art chapter. Once the trust concepts had been chosen, it was decided by the author that *belief*, *confidence*, and *faith* were in some way not as defined as the rest of the trust concepts. This posed a challenge in that the implementation of the model of trust had to be able to distinguish between the two types of trust concepts. The OWL language was very well suited to handling this task. By using OWL it was possible to develop the upper ontology by creating classes to represent the eight trust concepts before each was annotated as *abstract* or *concrete*.

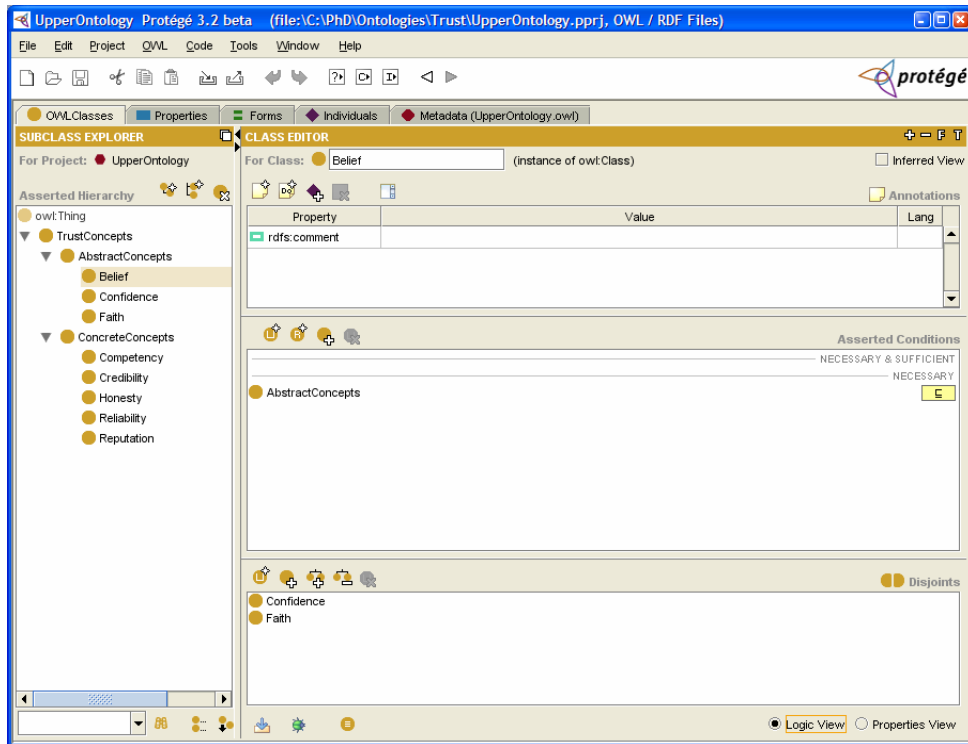


Figure 3-3 Upper Ontology Protégé View

Figure 3-3 shows the overall view of the upper ontology as seen through the Protégé ontology editor. This example describes the upper ontology as a set of trust concepts that are either *abstract* or *concrete*, and enumerates the *abstract* concepts that are disjoint with the highlighted *abstract* concept *belief*.

```

<owl:Class rdf:about="#Reputation">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#ConcreteConcepts"/>
  </rdfs:subClassOf>
  <owl:disjointWith>
    <owl:Class rdf:about="#Credibility"/>
  </owl:disjointWith> <owl:disjointWith>
    <owl:Class rdf:about="#Competency"/>
  </owl:disjointWith> <owl:disjointWith>
    <owl:Class rdf:about="#Honesty"/>
  </owl:disjointWith>
  <owl:disjointWith rdf:resource="#Reliability"/>
</owl:Class>

```

Figure 3-4 Upper Ontology OWL snippet (Reputation Only)

Figure 3-4 presents a snippet of the upper ontology in OWL format, which shows the OWL representation of *reputation*, which is a sub-class of *ConcreteConcepts* and is disjoint with the other *concrete* concepts. The upper ontology OWL document can be found in APPENDIX I – Trust Ontology Documents, Upper Ontology.

3.4.2 Trust Meta-model

The model of trust proposed in this thesis uses the trust meta-model to capture the intertwined nature of trust, as subjectively viewed by individuals across a large and broad population. Relationships can exist, or can form, between trust concepts that are found in the upper ontology. These relationships can vary from person to person, or from domain to domain, and are governed through the meta-model. Therefore, the trust meta-model provides three relationships that are used to capture all possible relationships between *abstract* and *concrete*, *abstract* and *abstract*, *concrete* and *concrete* trust concepts. Capturing the relationships between trust concepts enables the proposed model to accurately reflect an individual's subjective view of trust through a personalised model of trust. In addition, domain experts have the option to use these relationships when engineering domain specific models of trust.

Much like the upper ontology, the trust meta-model is a standard, which could also be adopted by a group such as the W3C, or on a *de facto* basis. Unlike the upper ontology the trust meta-model cannot be extended. The relationships were designed and developed by the author, yet the evaluation chapter provides experimental evidence that supports the use of such relationships.

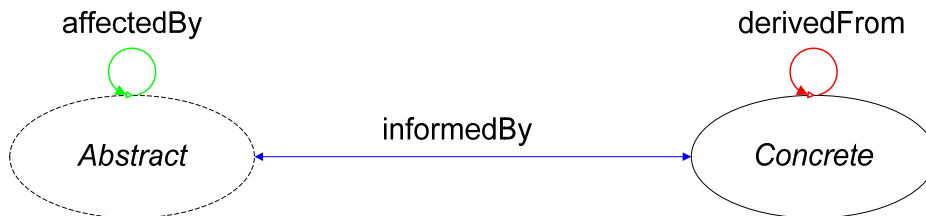


Figure 3-5 Meta-model

The upper ontology classifies trust concepts as either *concrete* or *abstract*. In a personalised model of trust a user can assert that one trust concept influences another trust concept. Therefore, it was decided that three relationships were required to capture all possible relationship combinations. As per Figure 3-5 the relationships are *derivedFrom*, *informedBy*, and *affectedBy*. These three relationships were inspired by three real world analogies, which can be used to loosely describe what each relationship in the trust meta-model means.

The strongest relationship is *derivedFrom* and implies a measured bond between *concrete* and *concrete* concepts only. A family based real world analogy inspired this relationship. In general, the bonds between family members are very strong, very influential, and personality characteristics can be seen across family members. It is the powerful influential strength that *derivedFrom* mirrors.

The second strongest relationship is *informedBy* and it can be formed between any *abstract* and *concrete* concept, and vice versa. In terms of a real world analogy the *informedBy* relationship mirrors a friendship. In general, friendships are quite close relationships and friends can influence each other. However, in general friendships do not influence people to the extent of that a family member can influence over a lifetime.

The weakest relationship is *affectedBy* and it captures relationships between *abstract* and *abstract* concepts. In terms of a real world analogy the *affectedBy* relationship mirrors a work colleague. Work colleagues can have some specific influence but generally it is not as strong as a friend or family member.

When the upper ontology and meta-model are used together it is possible to generate personalised models of trust that capture an individual's subjective view of trust. For example, one person may trust the British Broadcasting Corporation (BBC) news service because they feel the BBC has a strong *reputation* and therefore their reports have *credibility*. The personalised model of trust for this person can capture this relationship (*credibility derivedFrom reputation*) and all the other relationships that may exist across all of the trust concepts. Another person may feel that the BBC has a lot of *competency* and therefore that person could have a lot of *confidence* in their reports. The personalised model of trust for this person would capture this relationship (*confidence informedBy competency*). In this way the multi-faceted, personalised model of trust can capture a wide and diverse range of views that exist across a population.

The main strength of the trust meta-model is that it allows intertwined relationships between trust concepts to be captured. The trust meta-model is therefore the link between the upper ontology and personalised, and specialised, models of trust. The trust meta-model may have a limitation or weakness when considering an extended upper ontology. In order for the trust meta-model to work with a new trust concept added to the upper ontology, that concept would have to be identified as either *abstract* or *concrete*. The author was able to hypothesise, and later confirm through experiment, as to whether the eight trust concepts were *abstract* or *concrete*. A similar process would be required to enable the trust meta-model to operate correctly.

The trust meta-model had certain design issues that had to be addressed. The main issue was deciding whether the trust meta-model should be a separate and independent model. Separating the model had the advantage of enabling many personalised, and specialised, models to be generated from the upper ontology without any intrinsic overlap between all four models. In this way each model could be created, updated, used, stored, and deleted independently. However, such separation would require additional programming efforts and architectural design. The benefits provided through a separation of concerns led to the independent construction of the trust meta-model, the upper ontology, personalised and specialised models of trust.

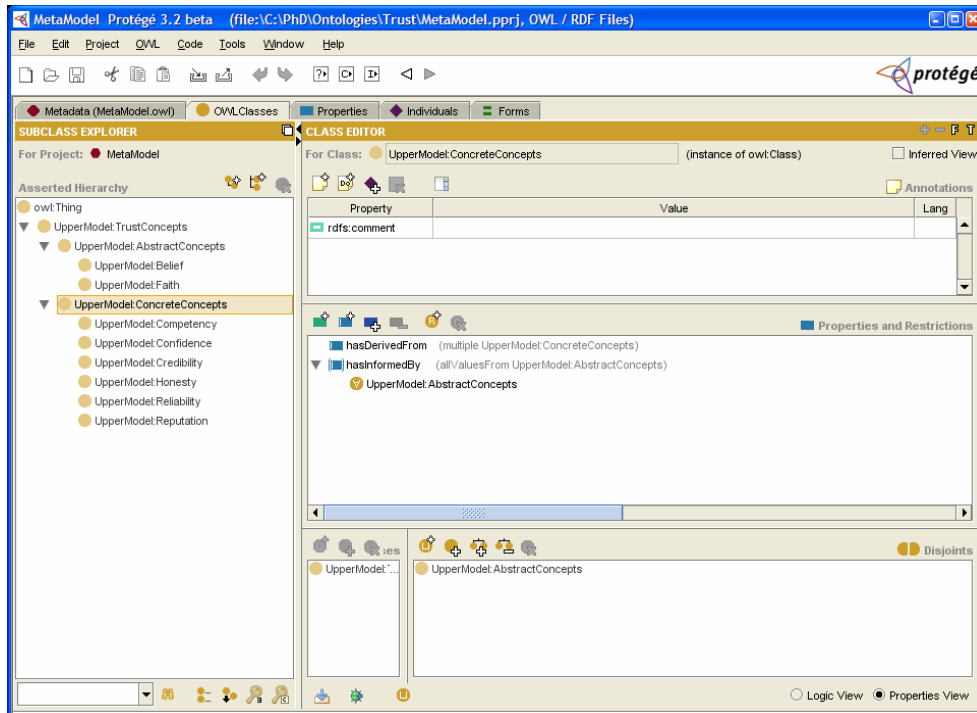


Figure 3-6 Meta-model Protégé View

Figure 3-6 presents the Protégé view of the meta-model, which imports the upper ontology. The *derivedFrom* and *informedBy* relationships are created in Protégé and the developer assigns these two relationships to the highlighted *concrete* concepts sub-class. The *affectedBy* and *informedBy* relationships were created and assigned to the *abstract* concepts sub-class (not illustrated in Figure 3-6) by the developer.

```

<owl:Ontology rdf:about="">
  <owl:imports rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl"/>
</owl:Ontology>

<owl:ObjectProperty rdf:ID="hasDerivedFrom">
  <rdfs:domain rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#ConcreteConcepts"/>
  <rdfs:range rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#ConcreteConcepts"/>
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="hasAffectedBy">
  <rdfs:range rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#AbstractConcepts"/>
  <rdfs:domain rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#AbstractConcepts"/>
</owl:ObjectProperty>

```

Figure 3-7 Meta-model OWL snippet

Figure 3-7 presents a snippet of OWL that reflects Figure 3-6. The *derivedFrom* relationship is assigned to *concrete* concepts and the *affectedBy* relationship is assigned to *abstract* concepts. The complete OWL representation of the trust meta-model can be found in APPENDIX I – Trust Ontology Documents, Meta-Model.

3.4.3 Personalised Model Design

It was hypothesised that the subjective nature of trust required personalisation when modelling trust. A personalised model of trust can capture this subjectivity. Personalised models are generated and updated by the user but not necessarily via an ontology editor. The mechanism used to generate a personalised model of trust is presented and described later in this sub-section.

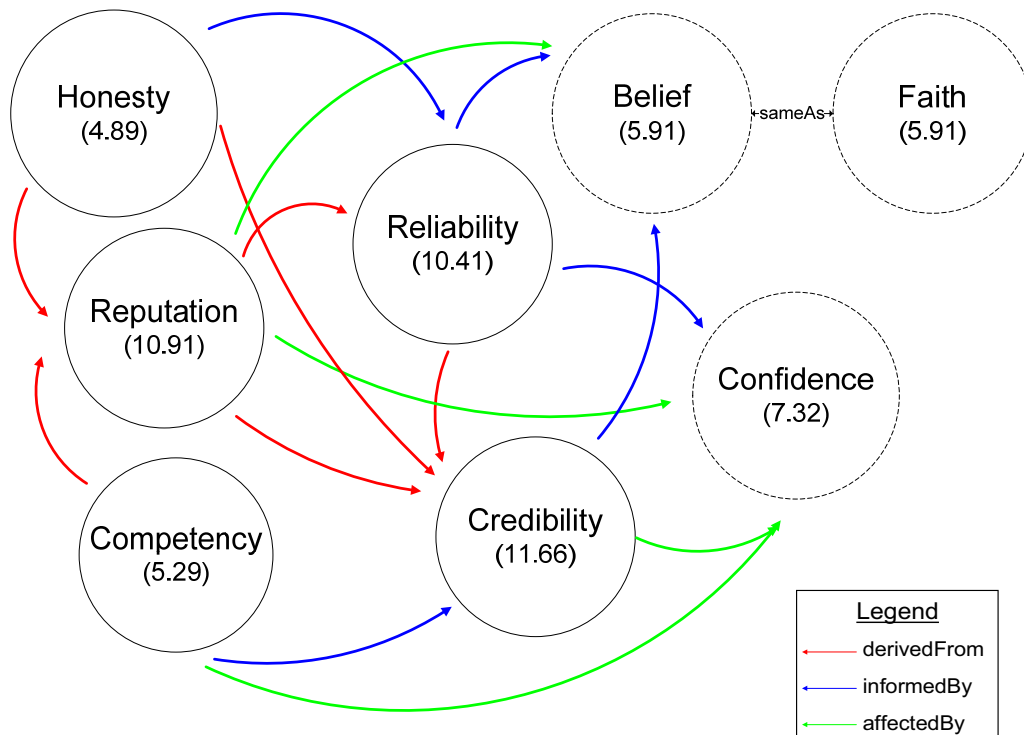


Figure 3-8 Personalised Model of Trust

Figure 3-8 illustrates an example personalised model of trust. In essence, the personalisation mechanism enables the generation of such personalised models of trust, based on trust concepts from the upper ontology and the three relationships found in the meta-model.

In the example shown in Figure 3-8, personalisation allow this user to assert that the concept *honesty* is influenced by *reputation* (*honesty derivedFrom reputation*), that *reliability* is influenced by *belief* (*reliability informedBy belief*), and so on. This can be repeated in order to build up a model of trust that suits the user’s requirements and that reflects their individual subjective view of trust.

The strength of the personalised approach is that a model of trust becomes user centric; in other words tailored to each user. This leads to the provision of trust recommendations that are personalised, which in turn may be of greater value to the user than a recommendation based on a generalised model of trust. The personalised approach is also adaptable in that it is possible for a user to alter their model as they see fit. For example, a young undergraduate student might be less interested in *reliability* and *credibility* in comparison to an older professional person. Yet, over time as the student becomes a professional it is possible for the model to be adapted to remain relevant. There is however extra overhead involved in providing a personalised model of trust, which may be construed as limitations. The overhead is in terms of additional user input and computational processing power. However, it is argued that the additional benefits of personalisation within trust justify the additional overheads. The benefits include a more bespoke user experience, calculated trust values that are specific to each user, and subsequently the provision of management functionality that is subjective to each users individual trust requirements.

The main design issues with the personalised model of trust were (i) how to enable a user to provide personalisation data, (ii) how to automatically generate a personalised model using that data, and (iii) how to store the model for later use.

In addressing the first design issue a mechanism was required that enabled the capture of the personalisation data describing relationships that may exist between trust concepts for an individual. The mechanism chosen was that of a web based questionnaire (see APPENDIX II – Research Experiments, Experiment Three, Trust Model Generation). It presents each trust concept to an individual, for example *credibility*. Then that individual is asked to select any of the other concepts that they feel *credibility* influences, for example *reputation* as per Figure 3-8. This process is repeated for all eight trust concepts and results in a set of data that can be graphed as per Figure 3-8. This mechanism is carried out via a web page, but it could also have been integrated into an application or as part of an interview process. The benefits of using a web based questionnaire include (i) simple and wider questionnaire distribution, (ii) easy input of result data into databases for analysis, and (iii) the test subjects took the questionnaire independent of any interference from the author.

In addressing the automated generation of personalised models of trust a mechanism was required to enable models to be built from personalisation data. In Figure 3-8, a number is associated with each of the eight trust concepts. The numbers are referred to as concept weights, which can be used to rank the trust concepts. The rank and weight of each trust concept is automatically calculated using a personalisation algorithm, which uses the set of personalised trust relationships that exist between the trust concepts. It is the Personalisation Engine, in Figure 3-1, that uses this algorithm. Since the relationship allocations are applied on a per user basis the rankings are also per user based. Therefore, personalised models of trust provide a set of personalised rankings for the trust concepts, which are then used to calculate a trust based recommendation. The personalisation algorithm is based on Kleinberg’s ‘Hypertext Induced Topic Selection’ (HITS) [Kleinberg, 1998] (see APPENDIX III – Implementation Code, Trial Data, and Sundry, Sundry, HITS Algorithm).

	userSource	modelName	concept	reputation	reliability	competency	credibility	honesty	belief	fath	confidence	personalisedRank
1	18041978	default	reputation	1	0	1	0	1	0	0	0	10.911
2	18041978	default	reliability	1	1	0	0	1	0	0	0	10.4114
3	18041978	default	honesty	0	0	0	0	1	0	0	0	4.89452
4	18041978	default	fath	1	1	0	1	0	0	1	0	5.91035
5	18041978	default	credibility	1	1	1	1	1	0	0	0	11.6609
6	18041978	default	confidence	1	1	1	1	0	0	0	1	7.31751
7	18041978	default	competency	0	0	1	0	0	0	0	0	5.29372
8	18041978	default	belief	1	1	0	1	0	1	0	0	5.91035

Figure 3-9 Personalisation Data and HITS Results

The HITS algorithm was chosen as the notions of hub and authority that it presents are applicable to the trust concepts and the model of trust proposed in this thesis. The HITS algorithm enables the weighting and ranking of web pages based on the URL links between those web pages. The HITS algorithm is suitable for use in generating personalised models of trust as the trust concepts are comparable to web pages, and the trust relationships are comparable to URL links. The HITS algorithm is applied to an individual’s personalisation data to weight and rank the individual concepts in respect to each other. A trust concept can be viewed as an authority if it influences many other concepts (see *credibility*, Figure 3-8), and viewed as a hub if it is influenced by other concepts (see *honesty*, Figure 3-8). The personalised model of trust presented in Figure 3-8 is captured in Figure 3-9 as a set of database entries. In Figure 3-9 each relationship that is presented between concepts is mapped as a ‘1’ in order to allow the HITS algorithm to calculate a weighting for each trust concept. For example, *reputation* influences *honesty* as per the selected cross-reference in Figure 3-9.

In Figure 3-8 and Figure 3-9, *credibility* is influential to many other concepts, whereas *honesty* is only influenced by many other concepts. It is for these reasons that the HITS algorithm has calculated and issued *credibility* with a relatively high score (11.66) and *honesty* with a relatively low score (4.89). From this it can be inferred that it is *credibility* that is this individual's highest ranked (number one) trust concept, and it is *honesty* that is ranked last (eight place).

The HITS algorithm is very applicable and suitable for automatically generating personalised models of trust. The HITS algorithm itself is a very reputable mechanism for weighting and ranking web pages using the links between pages. The evaluation chapter illustrates that the HITS algorithm is an accurate mechanism for generating personalised models of trust.

In order to store the personalised models of trust a tool was developed in Java that took the results of the HITS algorithm, and the relationship datasets from an intermediate MySQL [MySQL] database server, and converted them into an OWL based personalised model of trust. This personalised model can then be shared, read, parsed, and used to calculate trust based recommendations on a per user basis. Storing a personalised model of trust in a MySQL database allowed the evaluation of experiment results belonging to hundreds of test subjects without the need for a more powerful computer system that could handle hundreds of ontology documents.

Figure 3-10 provides a view of the personalised model of trust as seen in Protégé. A partial instance of the personalised model is shown in Figure 3-11. Finally, Figure 3-12 provides a snippet of OWL code from that personalised model.

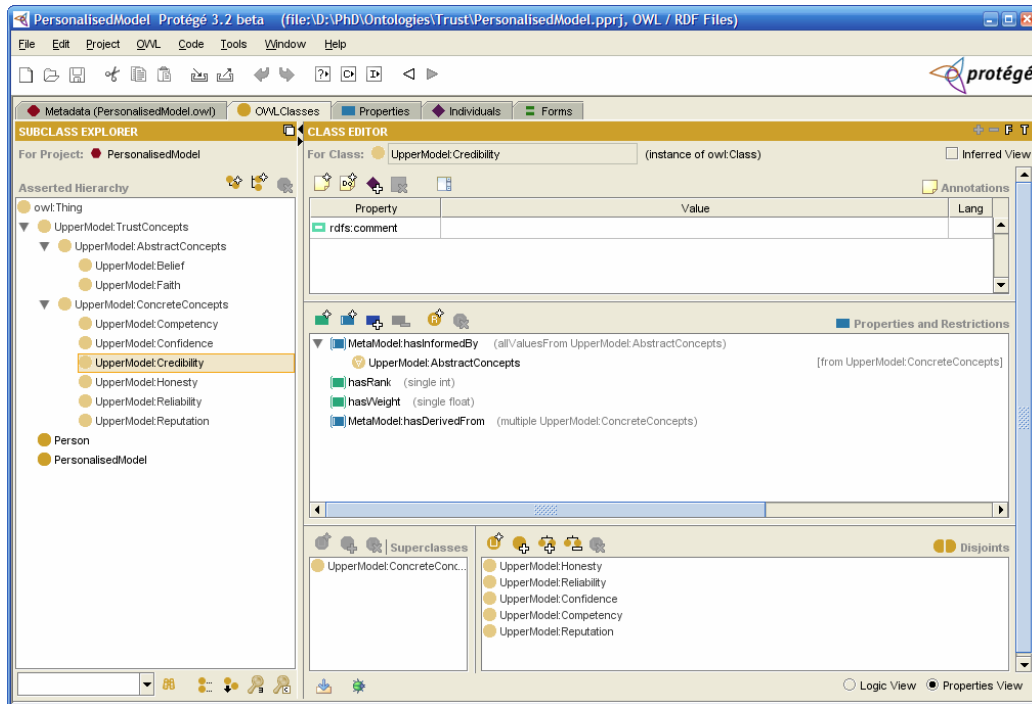


Figure 3-10 Personalised Model Protégé View

Figure 3-10 is focused on the highlighted trust concepts *credibility*, which in addition to the imported properties has two OWL datatype properties; *hasRank* and *hasWeight*.

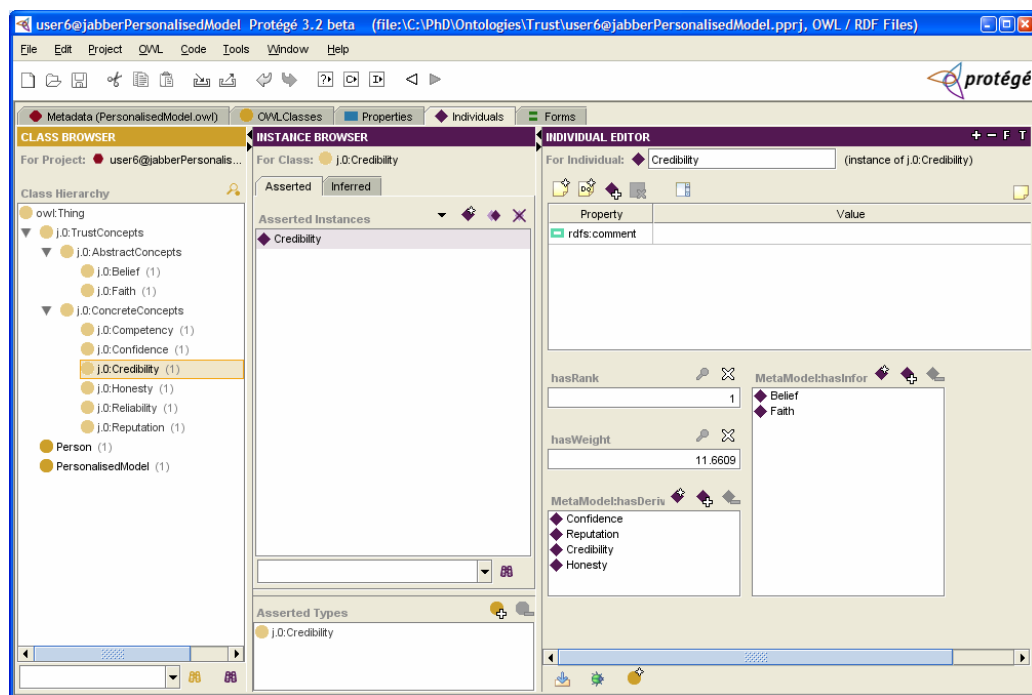


Figure 3-11 Instance of a Personalised Model of Trust in Protégé

In Figure 3-11 the partial instance illustrates *credibility*, which is ranked number one with a HITS algorithm calculated numeric weighting of ‘11.66’. The remaining seven trust concepts also have a weight and ranking. The benefits of such rankings is that trust based recommendations can be calculated with respect to each individual’s personalised model of trust. However, a domain specific model is also required to provide such recommendations.

```

<j.0:Confidence rdf:ID="Credibility">
  <MetaModel:hasDerivedFrom rdf:resource="#Credibility"/>
  <hasRank rdf:datatype="http://www.w3.org/2001/XMLSchema#int">
    1
  </hasRank>
  <hasWeight rdf:datatype="http://www.w3.org/2001/XMLSchema#float">
    11.66
  </hasWeight>
  <MetaModel:hasInformedBy rdf:resource="#Confidence"/>
</j.0:Confidence>

```

Figure 3-12 Partial Instance of a Personalised Model of Trust OWL snippet

Figure 3-12 provides a snippet of OWL code that presents an instance of *hasRank* and *hasWeight* for *credibility*. A complete, example, personalised model of trust can be found in APPENDIX I – Trust Ontology Documents, Personalised Model.

3.4.4 Domain Specific Model Design

Systems such as REFEREE, SULTAN, TRELIS, and Fidelis provide a model trust that can be specialised. A user can assign a trust level to another user with respect to a certain domain, for example the user can state that they have a high level of trust in another user with respect to a specific domain such as ‘auto-mechanics’ or ‘medical procedures’. In these cases, trust values can be a general integer value between 1 and 10 [Golbeck, 2004], a float value between 0 and 1 [Labalme & Burton, 2001], or a positive, neutral, or negative assignment as used in eBay [eBay]. However, in many cases the underlying trust value does not reflect domain specific properties.

The Protégé ontology editor was used in the development and engineering of two domain specific models using the upper ontology and trust meta-model. The Web Services domain specific model (see Section 3.4.4.2) is a more complex model with various relationships and a wide range of properties, which provides a set of evidence on which recommendations are calculated. The Instant Messaging domain specific

model (see Section 3.4.4.3) is a relatively simplistic model that has a single property, an integer, which reflects the opinion of users about another user with respect to trust. For the purposes of this thesis Protégé outputs the models in OWL format, which enables the capture of both simplistic and complex domain specific models.

The strength of providing domain specialisation is that it enables the capture of domain specific attributes in order to provide trust based recommendations that are specific to an application domain. Therefore, recommendations are not only personalised on a per user basis but they are specific to a particular application domain. Recommendations specific to a domain may be more useful to users than recommendations made with respect to no particular domain. Recommendations can then be made across a range of application domains therefore increasing the value of the model of trust. The combination of personalisable and specialisable model of trust is not found in current state of the art approaches. However, a possible, minor, limitation to this approach is that the engineering of specialised models of trust requires one or more domain experts, which is an issue not found in generalised domain approaches. In addition, mechanisms will be required to gather instance data for use in calculations that must also be able to operate on a per domain basis.

The main design issues with domain specialisation were (i) how to design, develop, and maintain a domain model and (ii) how is trust annotation provided in conjunction with a domain model.

In addressing the first design issue it is a domain expert that designs and develops domain models, such as Web Services. This model could be released as a standard (W3C, *de facto*, and so on) but it could be edited and updated either by the original developer, community members of the particular domain, or even an individual.

Addressing how trust annotation is provided depends on the domain itself. If two or more domain models exist that are very simple and opinion based then tools could be re-used across these two models. However, if the two domain models are complex and very specific, and very different, then trust annotation tools would have to be developed separately.

3.4.4.1 General Engineering Process

A domain specific model is an instance of the upper ontology and meta-model, which is engineered towards a specific application domain. The process of engineering a domain specific model is undertaken by a domain expert, who can be a developer or end user. The engineering process is a three step process that is repeated for all eight trust concepts;

- (1) Any sub-classes of a trust concept are identified and created,
- (2) For each sub-class, any properties that reflect that particular domain are identified and created,
- (3) The domain expert can add relationships between the chosen trust concept and other trust concepts.

In step one the domain expert examines a trust concept, particular to a given domain. For example the domain may be a postal service and the concept in question could be *reliability*. In this step the domain expert asks ‘what makes a postal service reliable?’ The domain expert might decide that *reliability* could be sub-classed into ‘performance’ and ‘message delivery’. These two classes are created as sub-classes of *reliability* and step two begins.

In step two the domain expert takes each sub-class and creates properties that reflect that sub-class. For example, the domain expert might decide that the ‘time taken’ to deliver an item of post is part of ‘performance’ and ‘guaranteed’ is part of ‘message delivery’. The ‘time taken’ property might be created as an integer that represents the number of hours that it takes to deliver an item of post and ‘guaranteed’ could be a Boolean. In this way it can be said that the *reliability* of a postal service can be based on its ‘performance’ and ‘message deliver’ What is considered a trusted postal service is based on instance values and user preference. For example, a user may state in a policy that ‘guaranteed’ ‘24 hour’ services are considered trustworthy and acceptable. This approach is similar to the approach taken by [Bargh et al, 2002] in state of the art where ‘dependability’ consists of ‘security’, ‘reliability’, ‘availability’, and so on.

In regard step three, the domain expert can optionally add trust relationships between the trust concepts in the domain specific model, which serves to reflect any relationships that may be present within that particular domain in the real world. For example, the domain engineer can assert that the *reliability* of a web service is *derivedFrom* the *reputation* of that web service. This is the same as using the meta-model to assign relationships between trust concepts when generating personalised models of trust. By doing this the domain engineer is providing their expert opinion to provide a ranking for the trust concepts specific to a domain. These rankings could be used in conjunction with, or in place of, or ignored by trust calculation algorithms to provide trust based recommendations.

3.4.4.2 Evidence Based - Web Service Specialisation

The Web Services domain specific model describes each trust concept in terms of Web Service specific sub-classes and properties. The objective of developing this domain specific model was to develop a platform that allows trustworthy Web Service selection. The Web Services domain was chosen as it has a rich set of properties, which are clearly defined and reflective of the Web Services domain. This led to the establishment of the term ‘evidence based’ to describe complex application domains. The strength of evidence based specialisation is that it can capture the complex properties and relationships found in complex application domains.

A partial illustration of the model of trust specialised towards Web Service is shown in Figure 3-13. Note that only *reliability* is illustrated. In this Web Services domain model *reliability* is selected from the upper ontology and the developer has decided that *reliability* has the following sub-classes; *assurance*, *availability*, *performance*, and *msgDelivery*. The developer has decided that this sub-class contains a set of properties. For example, *availability* has a set of properties that includes *downtime*, *meanTimeBetweenFailure*, and *resilience*. In addition, the domain expert has decided to use the *derivedFrom* relationship found in the trust meta-model. The developer has asserted that *reliability* is *derivedFrom* *reputation* in the Web Services domain specific model. Each of the remaining seven trust concepts are engineered in this way. The full Web Service domain specific model can be found in OWL format in the appendices, see APPENDIX I – Trust Ontology Documents, Specialised Models, Web Services Domain Specific Model.

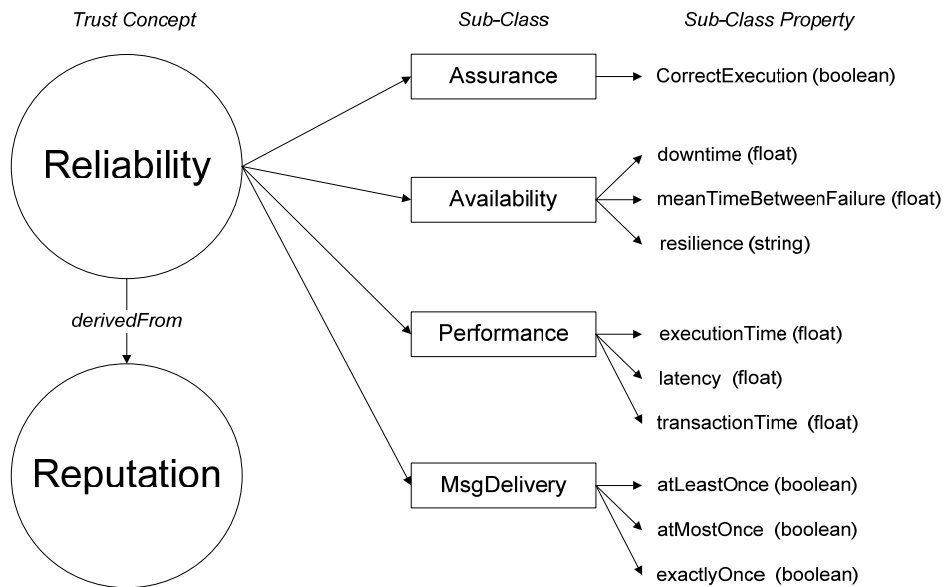


Figure 3-13 Web Service Specific Model (Reliability Only)

It is important to note that the selection criteria for the sub-classes and properties in Figure 3-13 were based on a rapid prototyping of a domain specific model of trust specialised towards Web Services and are by no means empirical or complete. They do however reflect a set of classes, properties, and relationships whose instances could be used to describe what make a Web Service trustworthy or untrustworthy. In total, the Web Services domain has 38 classes, 8 object type properties, 46 datatype properties, and 28 restrictions.

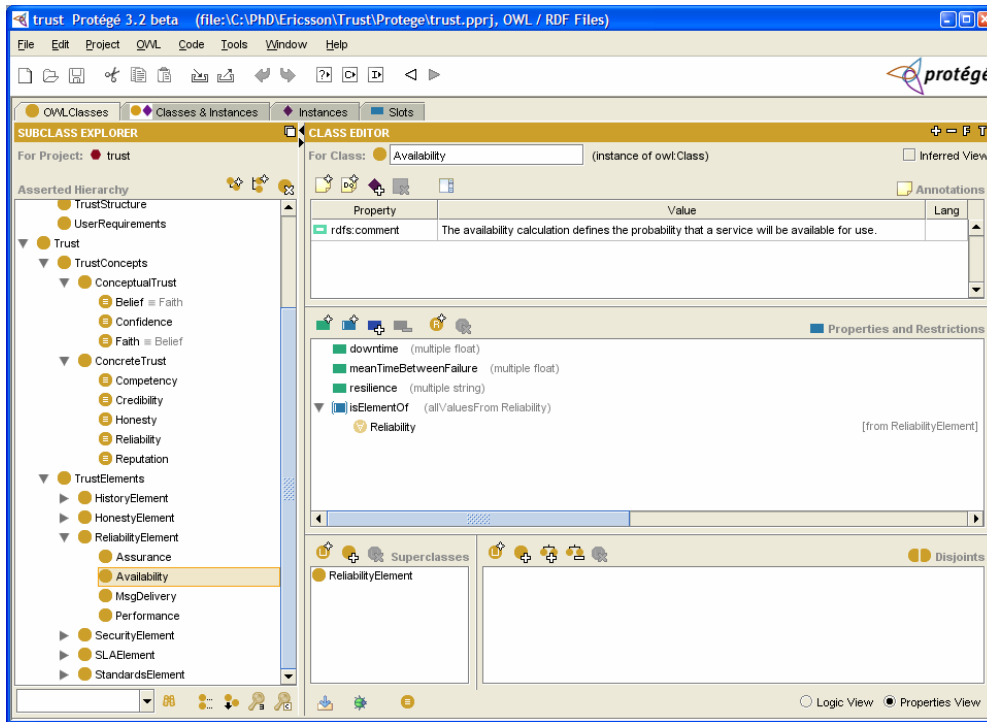


Figure 3-14 Web Services Domain Model Protégé View

Figure 3-14 illustrates the Protégé view of the Web Services domain specific model for the highlighted sub-class *Availability*. As per Figure 3-13 the datatype properties of *Availability* are *downtime*, *meanTimeBetweenFailure*, and *resilience*.

3.4.4.3 Opinion Based - Instant Messaging Specialisation

The Instant Messaging domain specific model describes each trust concept in terms of Instant Messaging specific sub-classes, properties, and relationships. The objective in creating the Instant Messaging domain model was to develop a platform that allows Instant Messaging users to regulate access to certain information based on trust. The Instant Messaging model is quite simplistic and is based on the opinion of Instant Messaging users. The strength of a simple model can lead to easier user interaction, simple trust calculations, and maybe more easily understood recommendations. In addition, domain experts would not be required to engineer a simple, opinion based model of trust. However, a simple model can not capture every application domain.

The Instant Messaging domain model has a single property, an integer value, associated with each of the eight trust concepts as per Figure 3-15. This reflects the opinion of one Instant Messaging user about another Instant Messaging user.

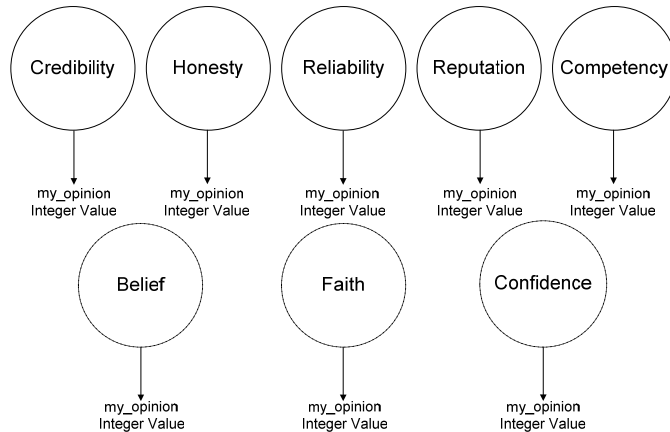


Figure 3-15 Instant Messaging Specific Model

The integer values can be ‘1’, ‘2’, ‘3’, or ‘4’, which translates to *very low*, *low*, *high*, or *very high*, respectively. In this way the single property associated with each trust concept reflects the opinion of a user for that trust concept. For example, Alice can annotate Bob with a *very high reputation* value. The Instant Messaging domain has 12 classes, 4 object properties, 5 datatype properties, and no restrictions.

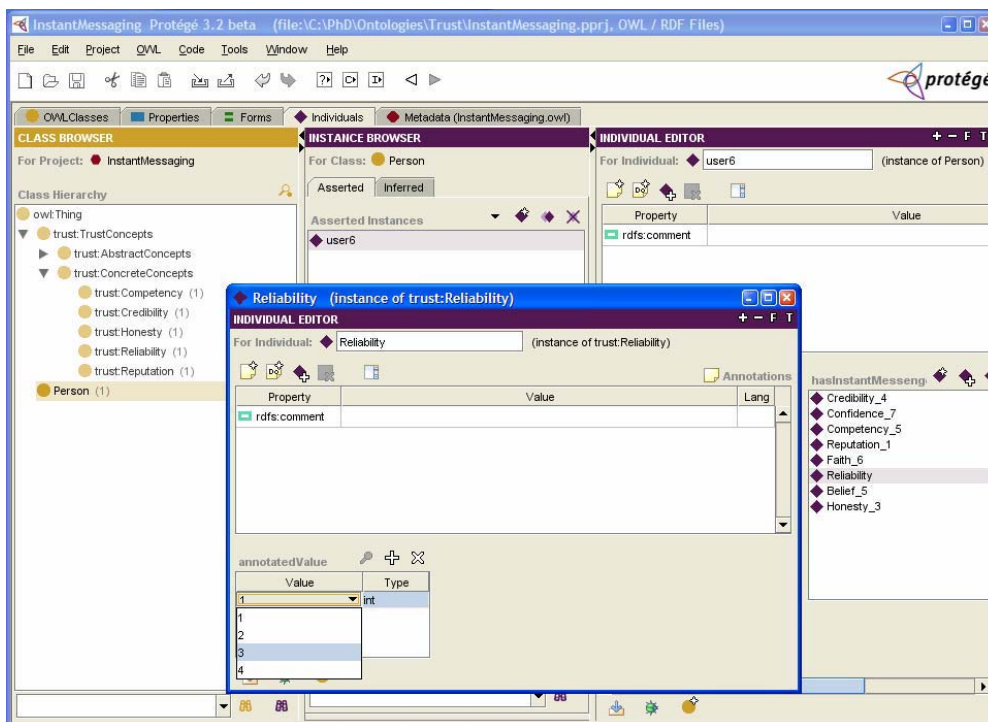


Figure 3-16 Instant Messaging Domain Model Protégé View

Figure 3-16 illustrates one annotation mechanism where annotation can be made using Protégé. In this instance a user has annotated ‘user6’ with a *reliability* value of ‘3’, or *high*. The full ontology Instant Messaging domain specific model can be found in OWL format in the appendices, see APPENDIX I – Trust Ontology Documents, Specialised Models, Instant Messaging Domain Specific Model.

3.5 Supporting Framework

The overall framework presented in Figure 3-1 makes use of trust data, trust calculations, and also trust policy to provide personalised trust based recommendations specific to different application domains.

3.5.1 Trust Data Annotation

Trust data is gathered via a mechanism of trust annotation that reflects a domain specific model of trust. Whether this mechanism is simple or complex depends on the particular domain specific model of trust.

In the Instant Messaging domain, users are allowed to annotate other users with their opinion with respect to each of the eight trust concepts. This is a simple mechanism that provides the user with a GUI (see Section 6.6.2.1) so that annotation can be carried out. However, the annotation of Web Services is based on the rich set of properties found in the Web Services domain. Therefore, this necessitates a different annotation approach. In this thesis a user annotates a Web Service using Protégé (see Section 6.4.2.1). However, the mechanism for annotating a Web Service could be automated. This would require an additional monitoring mechanism that would automatically annotate a Web Service on a user's behalf.

3.5.2 Trust Calculation Algorithm

Performing trust calculations must reflect both the individual user's personalised view of trust and application domain in which it is calculated. As per Figure 3-1 the Calculation Engine uses a personalised model of trust in conjunction with a domain specific model and associated trust data to calculate an overall trust value. The personalised models of trust vary from person to person but the rank and weight data can be used in the same way across personalised models. Therefore, in this thesis a trust calculation algorithm does not need to handle differences within the ontological structure of personalised models of trust. However, a trust calculation algorithm must cope with differences across application domains in order to use domain specific trust data effectively. Two different trust calculation algorithms have been implemented that reflect the variances across the Web Service (see Section 6.4) and Instant Messaging (see Section 6.5) domain models. A trust calculation algorithm is a separate entity, reflecting an application domain, which the Calculation Engine uses.

3.5.2.1 Stages of Trust Calculation Algorithm

A trust calculation algorithm has three stages. The first stage focuses on the personalised model of trust, the second on the domain specific model of trust, and the third on domain specific trust annotation data.

To date, all trust calculation algorithms used a personalised model of trust in the same way. The ranking for all trust concepts is extracted from a personalised model of trust and the top three, or the three most highly ranked, trust concepts are used by the trust calculation algorithm. It is possible to reduce or increase the number of trust concepts that are used within the calculation algorithm, which can be accomplished by simply altering an input parameter. The HITS algorithm assigns a weight, as well as a rank, to each of the trust concepts and this weight could be used to indicate how many trust concepts should be used.

Once the trust calculation algorithm has extracted the three highest ranking concepts from the personalised model it seeks an associated domain specific model in order to extract information relevant to each of these three trust concepts. In the case of the Instant Messaging domain model this information is an integer from 1 to 4 and reflects an associated level of trust, for example *reliability* might be annotated as *very high*. Whereas, in the Web Services domain model the information that reflects the trust concept *reliability* includes *downtime*, *meanTimeBetweenFailure*, *resilience*, and *latency*.

Finally, instances of trust annotation data are sought for each of the top three trust concepts, relevant to the domain specific model. It is *myTrust* that returns instances of trust data for use in the trust calculation algorithm, which provides an overall trust value based on this data.

3.5.3 Trust Policy

Any management mechanism that uses the multi-faceted model of trust must be able to leverage this model of trust, work with the calculated overall trust values, and be flexible in its ability to work in Internet Environments. These requirements can be satisfied through the use of policy. A policy can refer to and use aspects of the model

of trust, incorporate the outcome of calculations as conditions, and model a wide range of events that occur in Internet environments. These policies can be managed internally in *myTrust*, or separate to *myTrust*.

A trust policy is used in conjunction with an overall trust value to provide a trust recommendation. A policy is specified by the individual, specific to a domain specific model of trust such as Web Services or Instant Messaging. For example, with respect to Web Services a policy might state that in order to automatically select a particular vendor service the service must have a minimum trust level of *very high*, and have a *downtime* of less than 10 seconds.

3.5.3.1 *myTrust* and Policy

Policy is not internal to the *myTrust* trust management service. However, a tool was designed and developed to enable an Instant Messaging user to create policies with respect to granting access to certain information (see Section 6.6.2.2). This tool is a GUI that enables policy specification, which is stored in a MySQL database and converted to OWL format. Therefore, in this instance policies are specified using the GUI and policies are represented in OWL. However, a more advanced policy solution also exists in the form of a Community Based Policy Management (CBPM) system. In the CBPM system a user can create a policy that states the minimum trust value that is accepted in order to allow access to certain information (see Section 6.5.2.3). The overall trust values are used by the CBPM in conjunction with these policies to provide a trust based recommendation, which states whether access should be granted or denied.

3.6 Summary

This chapter discussed the issues, influences, and challenges that impacted the design of the personalisable and specialisable model of trust that is multi-faceted. The design of this model of trust was presented in terms of the upper ontology, meta-model, and personalised and specialised models of trust. In addition, the framework in which this model of trust resides was described and its operation detailed.

4 IMPLEMENTATION

4.1 Introduction

This chapter discusses the implementation of the *myTrust* trust management service that provides trust based recommendations using the multi-faceted model of trust that is personalisable and specialisable.

Section 4.2 presents the implemented architecture of *myTrust* and details the various layers that are found in the architecture. This section includes UML sequence diagrams detailing the architecture interactions.

In Section 4.3 the supporting technologies are detailed, which includes technologies that support the model of trust in terms of representation, storage, reasoning, querying, and so on.

4.2 *myTrust* Architecture

The implemented architecture of *myTrust* is presented in Figure 4-1, which realises the architecture described in Figure 3-1 in the design chapter.

The *myTrust* architecture implements a Service Oriented Architecture (SOA), which enables trust recommendations to be issued without requiring an application to have knowledge of the underlining implementation platform. SOA provides a level of interoperability between platforms and between the services themselves, which are loosely coupled. In addition, SOA promotes reuse and it also enables *myTrust* to provide trust management services to independent platforms.

The communication protocol in *myTrust* is Simple Object Access Protocol (SOAP) and services are defined in Web Service Definition Language (WSDL). This provides a well known and standard way for providing communications between *myTrust* and the external systems that seek trust recommendations.

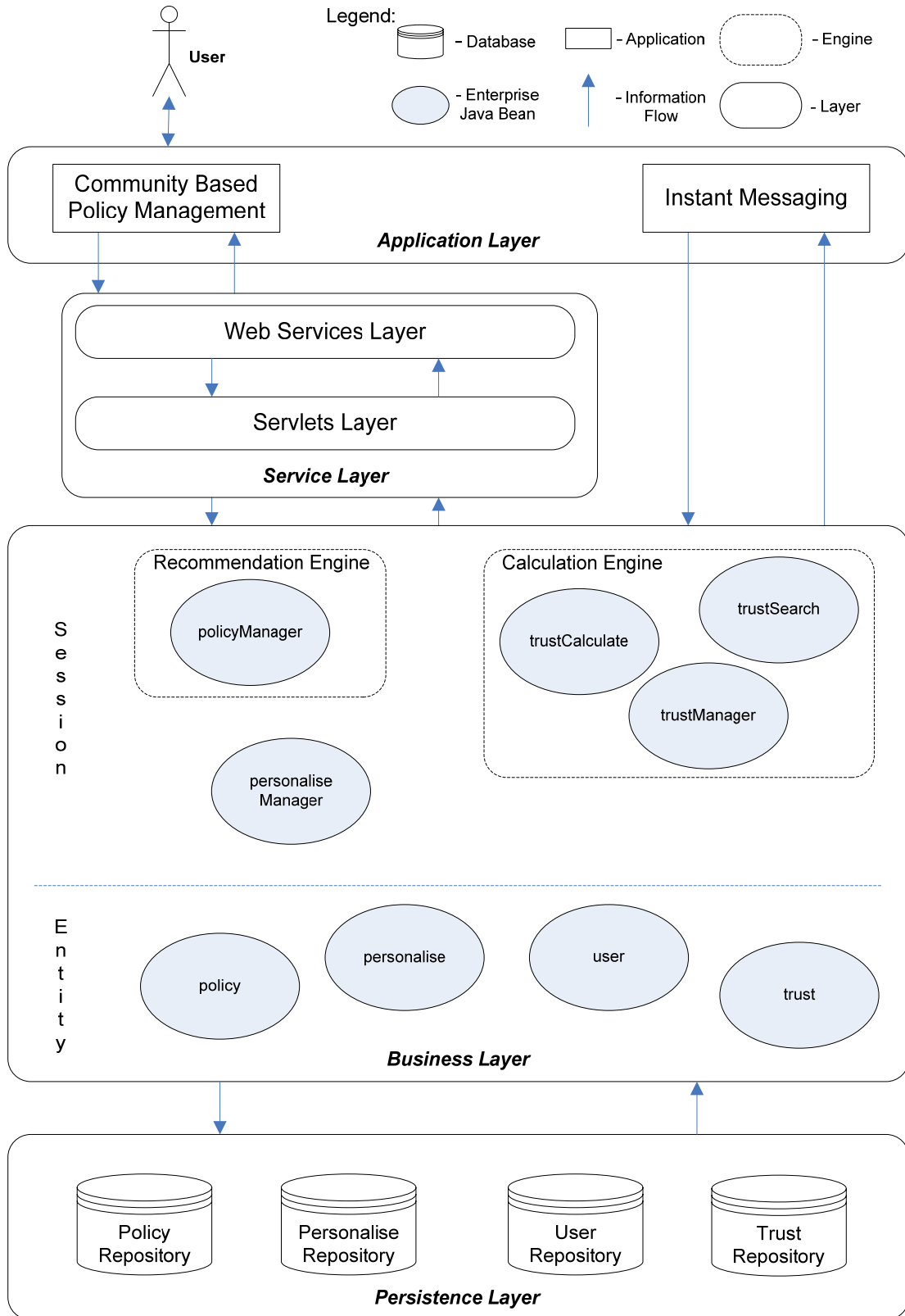


Figure 4-1 Implemented *myTrust* Architecture

As per Figure 4-1, the implemented architecture of *myTrust* is comprised of various layers including the Application Layer, Service Layer, Business Layer, and Persistence Layer.

The Application Layer contains applications that seek trust management services from *myTrust*. The Community Based Policy Management (CBPM) system and enhanced Instant Messaging (IM) application both seek a trust management service in this thesis.

The Service Layer is composed of a Web Service Layer and a Servlets Layer. The CBPM interacts with *myTrust* using the Service Layer, which in turn communicates with the Business Layer, whereas the enhanced IM application bypasses this layer.

The Business Layer is implemented using Enterprise Java Beans (EJB), which are deployed on a JBOSS [JBoss] Application Server. The EJB's are configured in the entity bean and session bean design pattern to control access to the Persistence Layer. In this configuration the entity beans access the MySQL database using a JDBC connection. A façade of session beans then communicate with these entity beans. EJB's are highly adaptable and available across disparately managed networks. This enabled *myTrust* to provide trust management services to the CBPM system and enhanced IM application, which reside across disparately managed networks in the overall Computer Science computer network in Trinity College Dublin.

The Persistence Layer is made up of several repositories (MySQL databases), which store all data used to provide trust based recommendations. This includes personalised model data (Personalise Repository) and trust data (Trust Repository), as well as policy (Policy Repository) and other user data (User Repository).

4.2.1 Architecture Overview

Figure 3-1 in the design chapter illustrates the overall framework in which *myTrust* operates. Figure 4-1 illustrates the implemented *myTrust* architecture. Figure 4-2 combines aspects of the *myTrust* framework featured in Figure 3-1 with aspects of the implemented *myTrust* architecture featured in Figure 4-1 in order to illustrate, and clarify, how Figure 3-1 and Figure 4-1 relate.

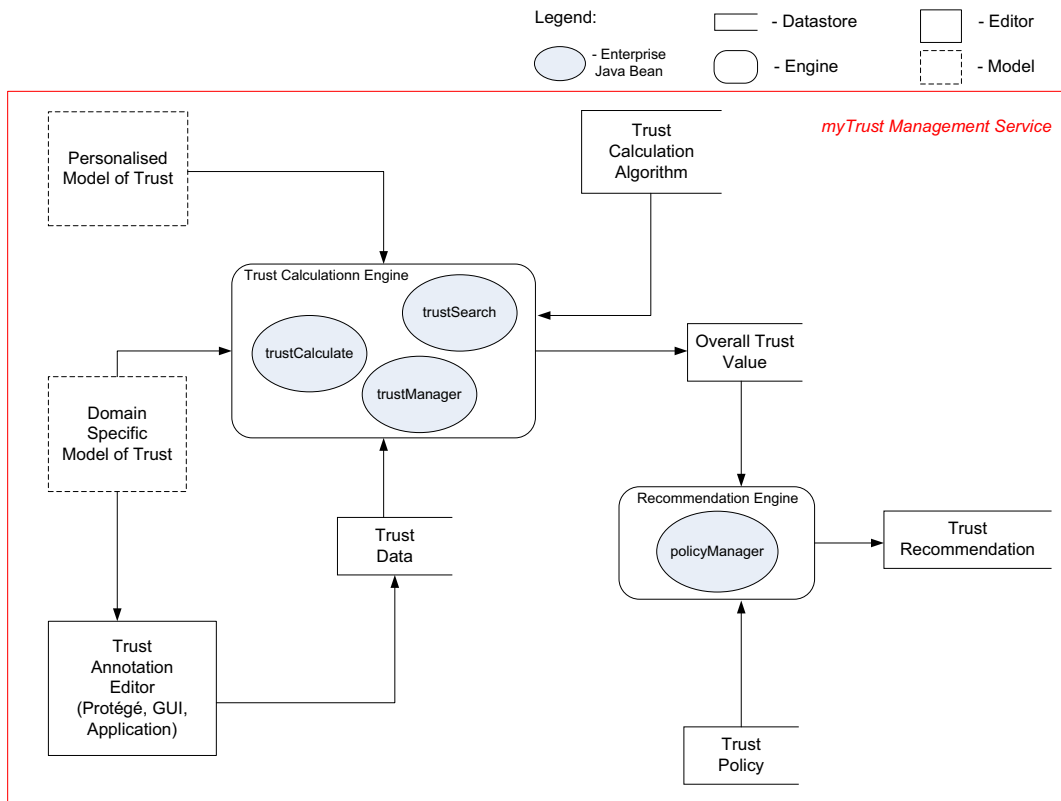


Figure 4-2 *myTrust* Framework and Implementation Relationship

In Figure 4-2 the Trust Calculation Engine is realised through the set of trustCalculate, trustSearch, and trustManager session beans. It is these beans that retrieve a personalised model of trust, domain specific model of trust, trust data, and a trust calculation algorithm in order to calculate an overall trust value.

The Recommendation Engine in Figure 4-2 is realised through the policyManager and cbpmManager. These beans use the overall trust value and a retrieved trust policy to make a trust recommendation.

The implementation code for *myTrust* can be found in APPENDIX III – Implementation Code, Trial Data, and Sundry, under Implementation Code The following sub-section presents the implementation undertaken for each layer.

4.2.2 *myTrust* Application Layer

myTrust provides a trust management service to the CBPM system and the enhanced IM application. This section describes how the CBPM and enhanced IM interface with *myTrust*.

4.2.2.1 Interface Options

In operation an application has two options when interfacing with *myTrust* in order to avail of the trust management service. As per Figure 4-1 the interface options are (i) via the Service Layer or (ii) via the Business Layer.

Applications can interface with the Service Layer, which in turn uses the functions of the EJB's within the Business Layer. The Business Layer communicates with the Persistence Layer in order to retrieve, create, edit, and delete information. In this way an application connects to *myTrust* as a Web Service. Offering an interface through a Service Layer enables external Web Services, applications, and systems to easily communicate with *myTrust*. The CBPM interfaces with *myTrust* using the Services Layer. It was possible to offer *myTrust* services to CBPM using two lines of code and a reference file. This occurred in a very short timeframe as the CBPM needed to know very little knowledge about the implementation of *myTrust*. All the CBPM developer needed to know was the URI to the Web Service.

Applications can also interface directly with the Business Layer, which bypasses the Service Layer. This approach is taken by the enhanced IM application. Such an approach requires the developer of an external application to have significantly more knowledge about the implementation of *myTrust*. The developer of an external application would have to liaise with the developer of *myTrust* in order to ensure correct understanding and subsequent communication between that external application and *myTrust*. This increases the amount of time that it takes to develop a link between that external application and *myTrust*. It is important to note that the enhanced IM application was allowed to interface directly with the business layer for testing and debugging reasons, and future applications will use the Service Layer.

Therefore, the quickest way to get connected to *myTrust* and avail of its trust management service is to use the Service Layer and connect through a Web Service. In this way the external application and *myTrust* are considered loosely coupled. However, for research and development purposes it is also possible to provide a more tightly coupled option. The Service Layer approach is the default method for allowing external applications to access *myTrust*.

4.2.3 *myTrust* Service Layer

This section details the operation of the *myTrust* Service Layer in terms of the Web Service Layer and the Servlets Layer.

4.2.3.1 Web Services Layer

Trust calculations can be sought from *myTrust* using the WSDL definition file `CalculateTrustSourceDestination.wsdl`, see Figure 4-3. The service takes two string parameters, a source user and a destination user, and returns an integer value representing the overall trust value. In effect, the service answers the question “How much does the source user trust the destination user” with an integer value for trust ranging from 1 to 4; *very low*, *low*, *high*, and *very high*.

```
<wsdl:message name="calculateTrustResponse1">
  <wsdl:part name="calculateTrustReturn" type="soapenc:int"/>
</wsdl:message>

<wsdl:message name="calculateTrustRequest1">
  <wsdl:part name="in0" type="soapenc:string"/>
  <wsdl:part name="in1" type="soapenc:string"/>
</wsdl:message>
```

Figure 4-3 Partial `calculateTrustSourceDestination` WSDL File

4.2.3.2 Servlets Layer

A Servlets layer was developed to provide a communication medium between the Business Layer and future development of Java Applets. Providing Java Applets with a link to *myTrust* is difficult because Java Applets can not communicate with a machine outside of the local machine on which it is running for security purposes. The introduction of Servlets enabled an Applet on one machine to communicate with *myTrust* on another machine that is in the Computer Science network at Trinity College Dublin.

The Web Services took advantage of this existing architecture and communicated with the Business Layer through these Servlets, which take the same parameters as the web service; source user, destination user, and returns a calculated overall trust value.

4.2.4 *myTrust* Business Layer

The core logic of *myTrust* lies in the Business Layer, as per Figure 4-1. This section describes the functions of the main session and entity beans under the sub-headings Trust Calculation Engine and Recommendation Engine.

4.2.4.1 Trust Calculation Engine

The Trust Calculation Engine is comprised of three main session beans (`trustManager`, `trustSearch`, and `trustCalculate`). These three session beans use two entity beans (`trust` and `user`) to communicate with the Persistence Layer. The entity beans are discussed in Section 4.2.5. The Trust Calculation Engine communicates trust data using two Java objects referred to as `trustData` and `userData` objects.

The `trustData` object has ten variables. These ten variables are instances of the eight trust concepts, a source user, and a destination user. The `trustData` object reflects the trust data that the source user has in the destination user. An example instance of `trustData` can be seen in seen in Figure 4-20.

The `userData` object has two variables. These two variables are instances of a `userID` string value and a `userRef` integer value. The `userData` object is used by *myTrust* to provide an anonymous user reference for a user identifier. An example instance of `userData` can be seen in Figure 4-22. Please note that this is an internal data representation mechanism for *myTrust*.

This current section presents UML diagrams that illustrate the calculation of an overall trust value. In addition, this section also describes the creation, storage, and retrieval of trust data that may exist between a source user and a destination user.

trustManager

The session bean trustManager provides functions to Web Services, Servlets, and applications, as well as to the session beans trustSearch and trustCalculate. The trustManager session bean provides six main functions:

(1)createUser(string src)

- Given a user source, creates a new user as illustrated in Figure 4-4.

Web Service
Servlet
Application

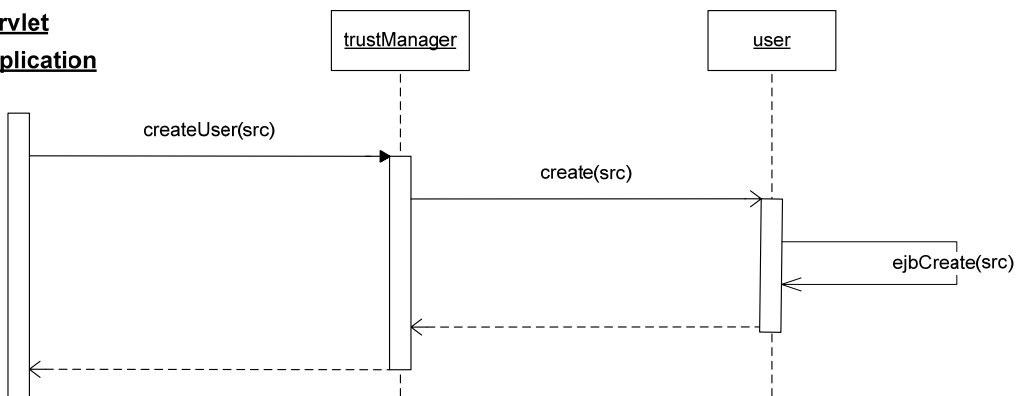


Figure 4-4 UML Sequence Diagram for trustManager.createUser

(2)trustData createUserDestinationTrust (string src, int belief, int competency, int confidence, int credibility, int faith, int honesty, int reliability, int reputation, string dest)

- Given a user source, a user destination, and eight integers that correspond to the eight trust concepts, creates a new set of trust data for the destination user as seen by the source user. See Figure 4-5.

Web Service
Servlet
Application

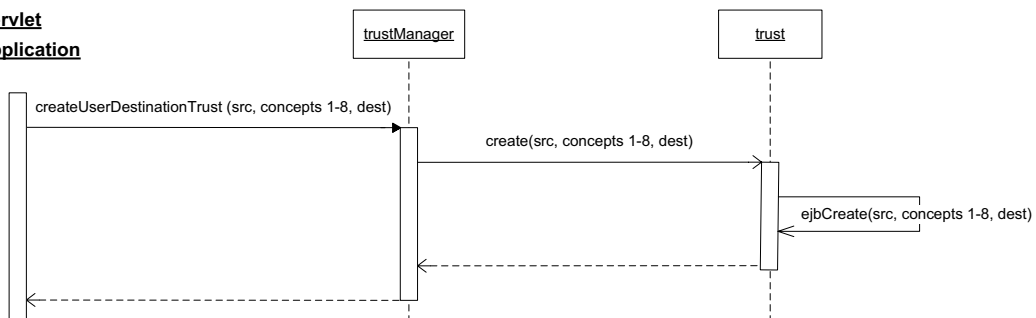


Figure 4-5 UML Sequence Diagram for trustManager.createUserDestinationTrust

(3)trustData getTrustByUserSource (string src)

- Given a source user, returns trust data for all the users that the source user knows. See Figure 4-6.

Web Service

Servlet

Application

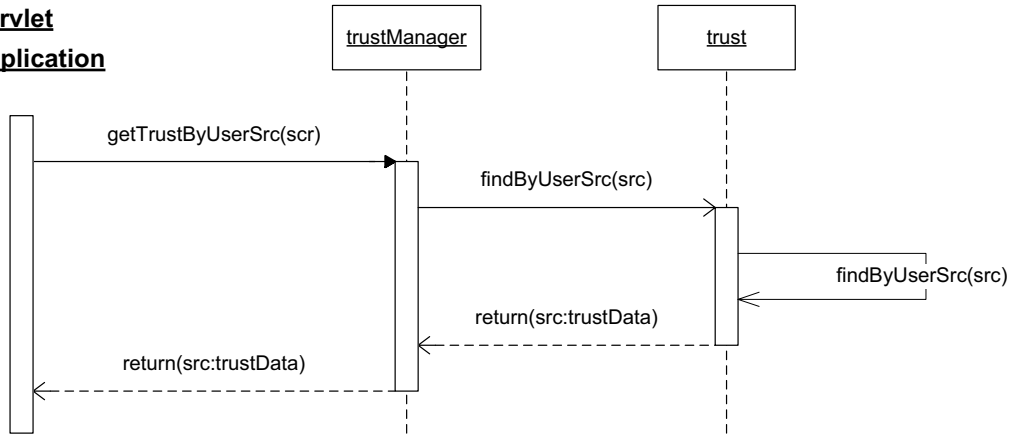


Figure 4-6 UML Sequence Diagram for trustManager.getTrustByUserSource

(4)trustData getTrustByUserDestination (string dest)

- Given a destination user, returns trust data for all the users that know the destination user. See Figure 4-7.

Web Service

Servlet

Application

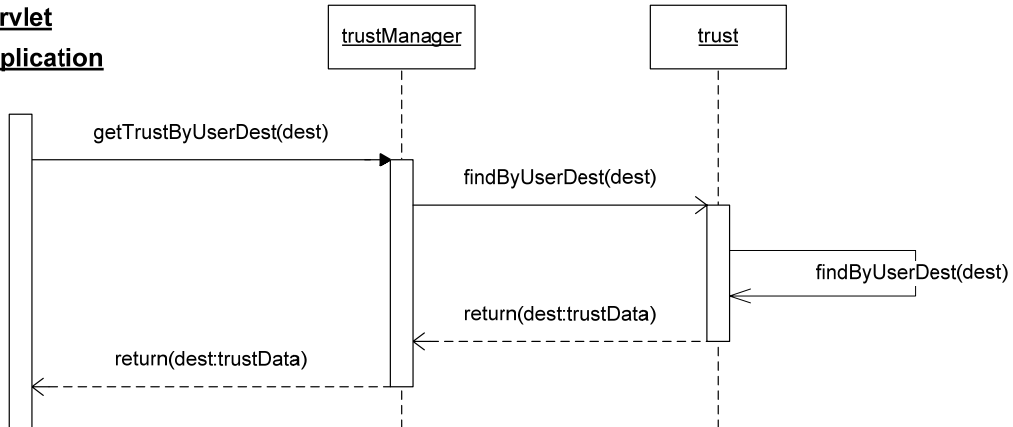


Figure 4-7 UML Sequence Diagram for trustManager.getTrustByUserDestination

(5)trustData getTrustByUserSourceDestination (string src, string dest)

- Given a source user and a destination user, returns trust data that the source user has in the destination user. See Figure 4-8.

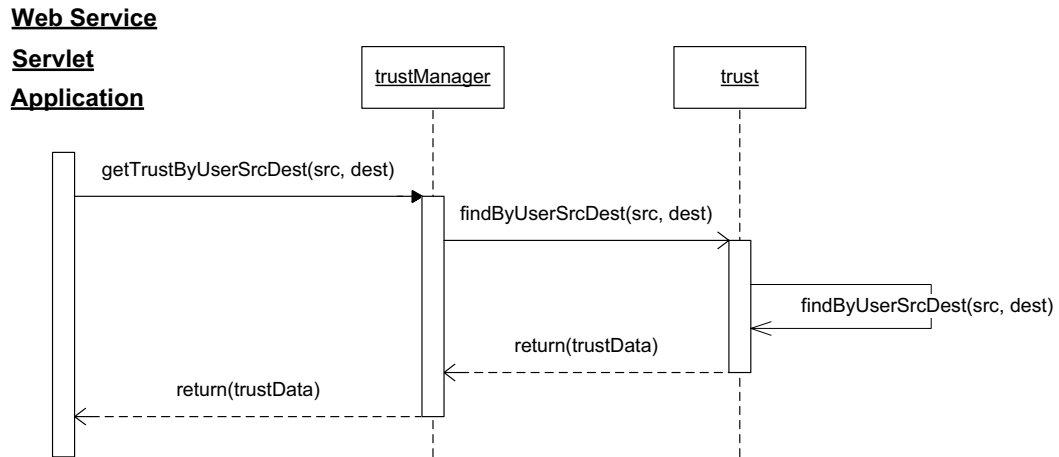


Figure 4-8 UML Sequence Diagram for trustManager.getTrustByUserSourceDestination

(6)editTrust (string src, int belief, int competency, int confidence, int credibility, int faith, int honesty, int reliability, int reputation, string dest)

- Given a user source, a user destination, and eight integers that correspond to the eight trust concepts, edits existing trust data for a destination user as seen by a source user. If trust data does not exist it is created as per Figure 4-5, otherwise the existing data is edited to reflect the parameters of this function. See Figure 4-9.

Web Service
Servlet
Application

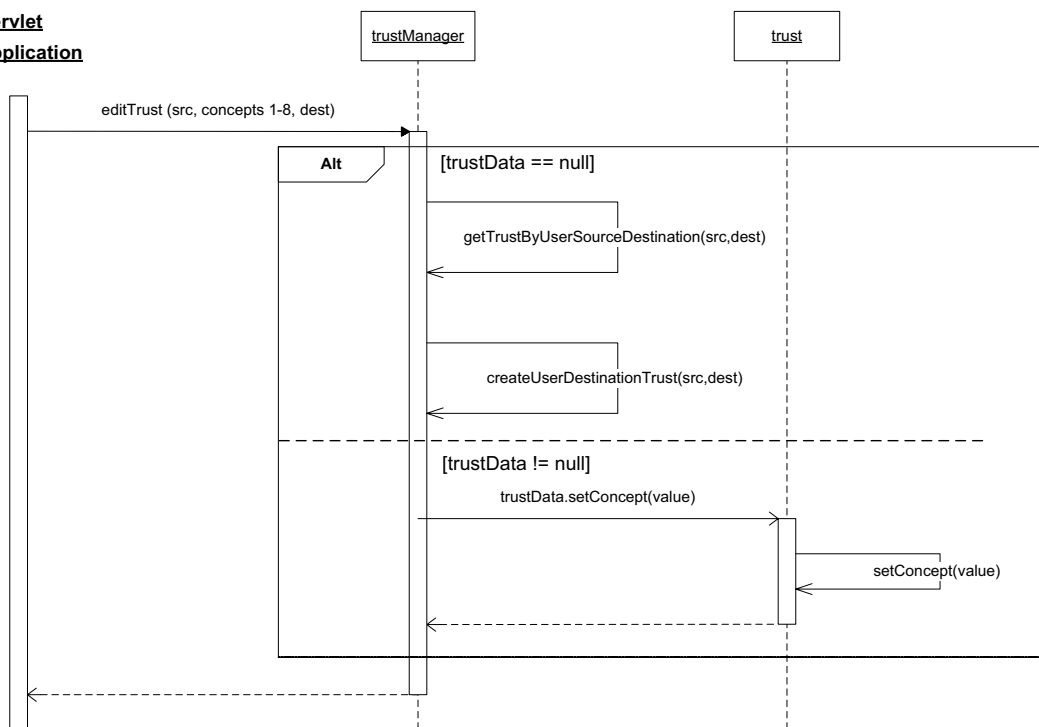


Figure 4-9 UML Sequence Diagram for trustManager.editTrust

Trust Search

The trustSearch session bean provides four main functions;

(1)trustData getUserSource(string src)

- Given a source user, returns trust data for all the users that the source user knows. See Figure 4-10.

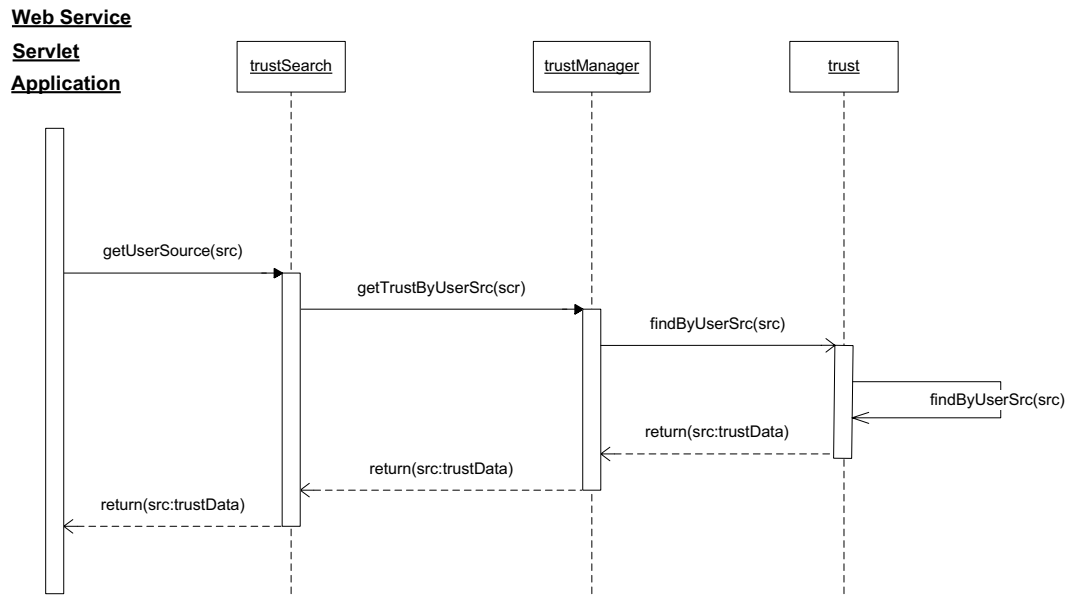


Figure 4-10 UML Sequence Diagram for trustSearch.getUserSource

(2) trustData getUserDestination(string dest)

- Given a destination user, returns trust data for all users that know the destination user. See Figure 4-11.

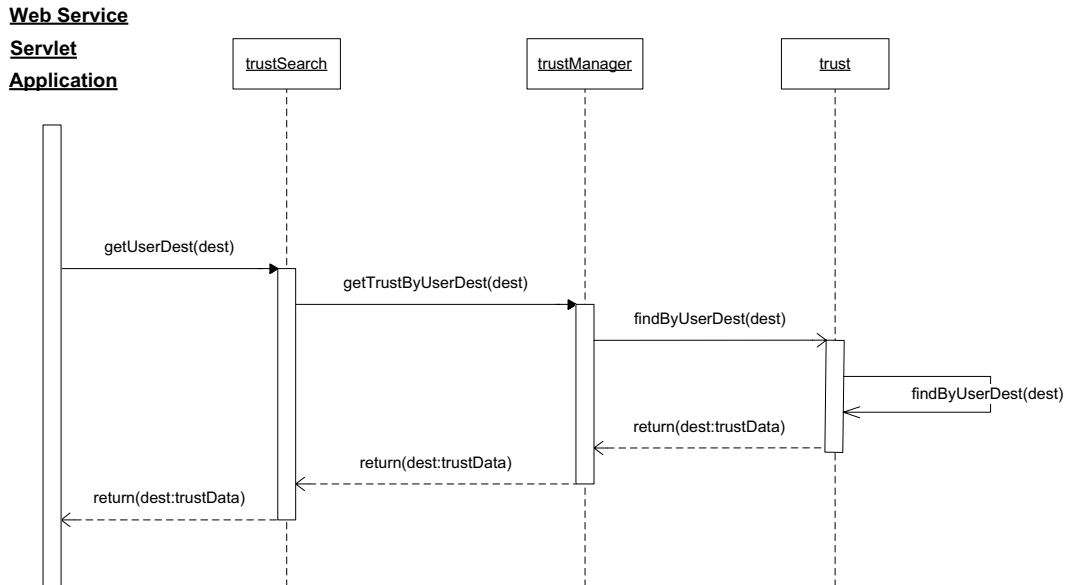


Figure 4-11 UML Sequence Diagram for trustSearch.getUserDestination

(3) trustData get1degrees(string src, string dest)

- Given a source user and a destination user, returns trust data that the source user has in the destination user. See Figure 4-12.

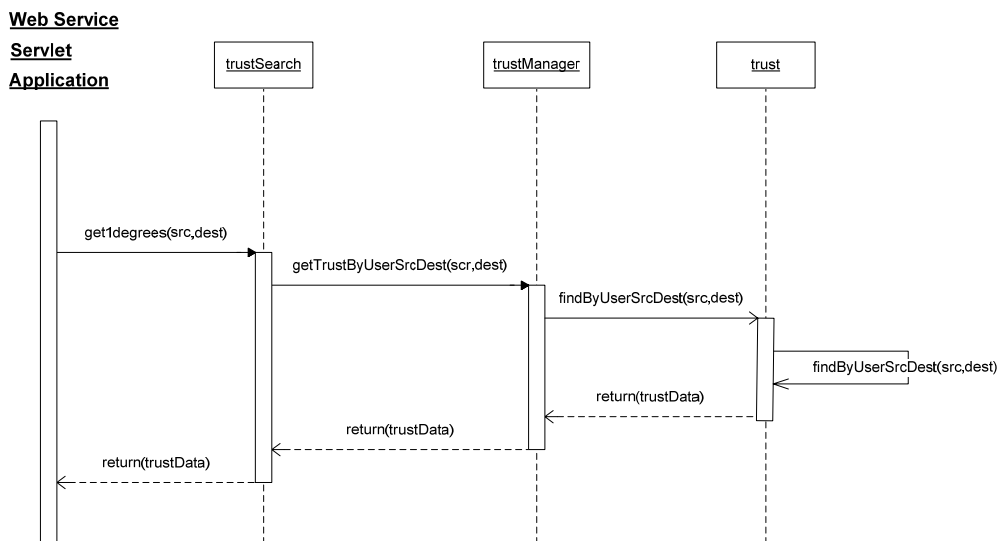


Figure 4-12 UML Sequence Diagram for trustSearch.get1degrees

(4)trustData get2degrees(string src, string dest)

- Given a source user and a destination user, returns trust data for users who know both the source and destination user, like friend of a friend. This function can be theoretically increased to 6 degrees of separation. See Figure 4-13.

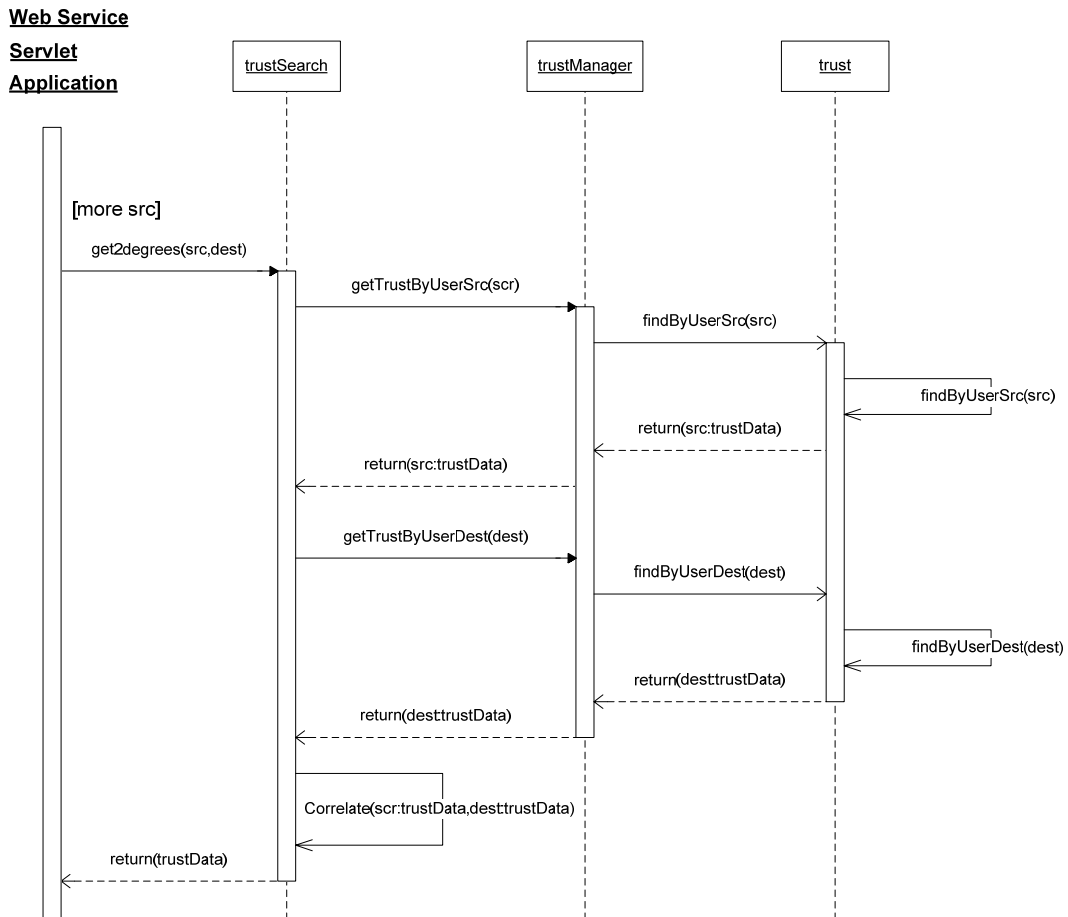


Figure 4-13 UML Sequence Diagram for `trustSearch.get2degrees`

trustCalculate

The trustCalculate session bean provides the two main functions:

(1)int calculateTrust (string src, string dest)

- Given a source user and a destination user, returns an overall trust value that reflects the source users trust in the destination user. See Figure 4-14.

(2)int calculateTrust (array src, string dest)

- Given an array of source users and a single destination user, returns an aggregated overall trust value that reflect the array of source users trust in the single destination user. See Figure 4-14.

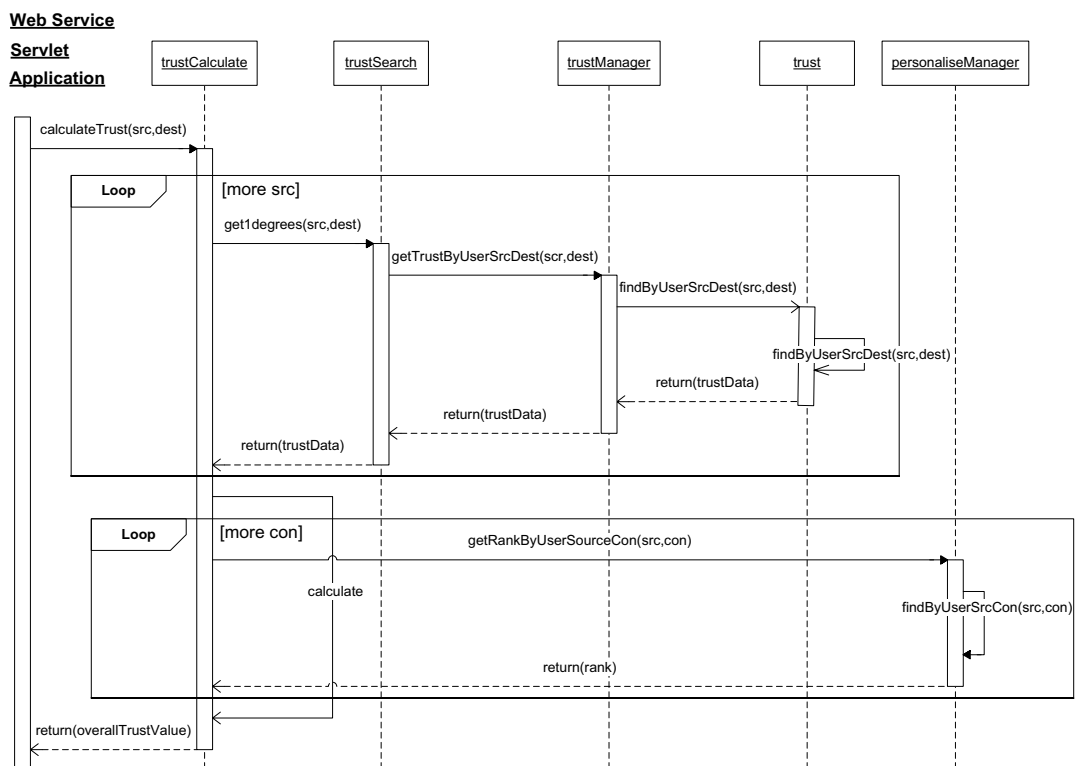


Figure 4-14 UML Sequence Diagram for trustCalculate.calculateTrust

Figure 4-14 illustrates a UML sequence diagram for the two functions calculateTrust. An application, Servlet, or Web Service can invoke these functions. When invoked the session bean trustCalculate retrieves trust data (trustData in Figure 4-14) for the destination user it has been provided with. It does this for one or many source users and the trust data is used to calculate an overall trust value. The session beans trustSearch and trustManager are used to retrieved this trust data. The calculations use the personalised rankings for each user source, which are retrieved from personaliseManager. Finally, the personalised overall trust integer value is returned.

personaliseManager

The `personaliseManager` retrieves personalised model of trust data using Java objects referred to as `personaliseData` objects. Within these objects are five variables, which are instances of a personalised model of trust for one source user with respect to one trust concept. This `personaliseData` object includes the primary key, source user, trust concept, rank, and weight. An example `personaliseData` instance can be seen in Figure 4-24.

The `personaliseManager` session bean provides the two main functions:

(1) `int getRankByUserSourceConcept (string src, string concept)`

- Given a source user and a trust concept, returns an integer value that reflects the given concepts rank in the source users personalised model of trust. See Figure 4-15.

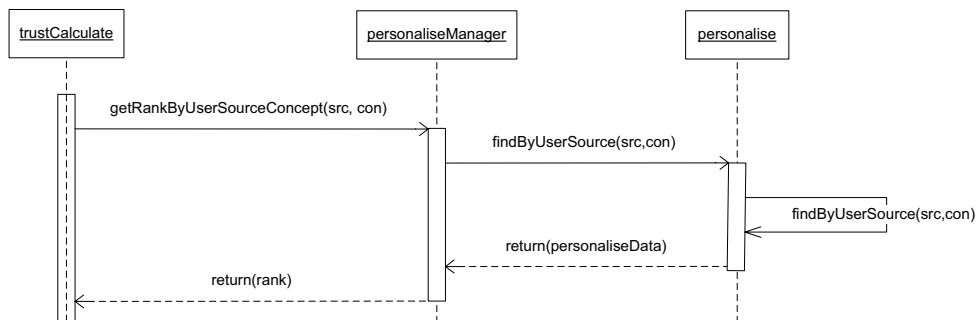


Figure 4-15 UML Sequence Diagram for `personaliseManager.getRankByUserSourceConcept`

(2)float getWeightByUserSourceConcept (string src, string concept)

- Given a source user and a trust concept, returns a float value that reflects the given concepts weight in the source users personalised model of trust.

See Figure 4-16.

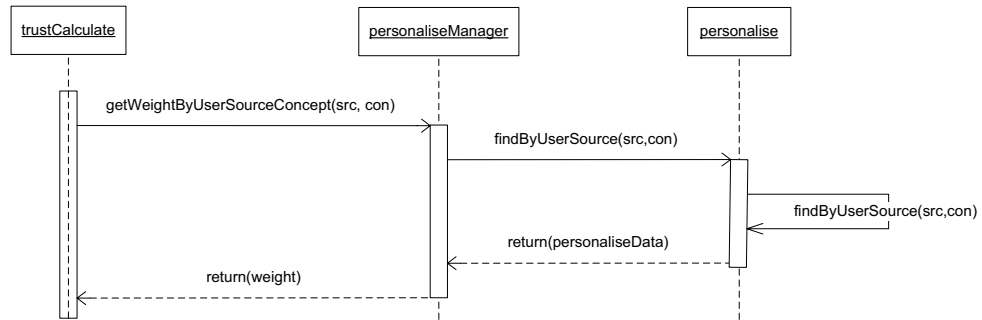


Figure 4-16 UML Sequence Diagram for personaliseManager.getWeightByUserSourceConcept

4.2.4.2 Recommendation Engine

The Recommendation Engine uses one session bean (policyManager), which communicates with the Persistence Layer through one entity bean (policy). This entity bean is discussed in Section 4.2.5. The Recommendation Engine can retrieve policy data that exists for a source user with respect to a given event in order to provide a trust based recommendation.

The Recommendation Engine communicates policy data in objects referred to as policyData objects. Within these objects are twelve variables. The most significant four variables are the source user, an event, a condition, and an action. The remaining eight of these twelve variables are optional; they are the eight trust concepts that may be used as additional constraints. An example instance of policyData instance can be seen in seen in Figure 4-26.

policyManager

The policyManager session bean provides the two main functions:

(1) createPolicy(string src, string event, int condition, int action)

- Given a source user, an event, a condition, and an action, a new policy can be created. Additionally, additional condition information can be provided that corresponds to the eight trust concepts. For example, a minimum overall trust value and a minimum value for *reputation* can be parameters. See Figure 4-17.

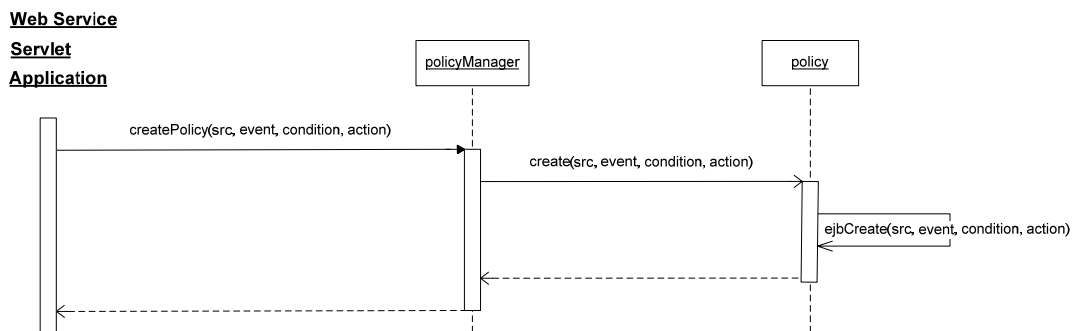


Figure 4-17 UML Sequence Diagram for policyManager.create

(2) policyData getPolicyByUserSourceEvent(string src, string event)

- Given a source user and an event, returns policy data for that user with respect to the specified event. See Figure 4-18.

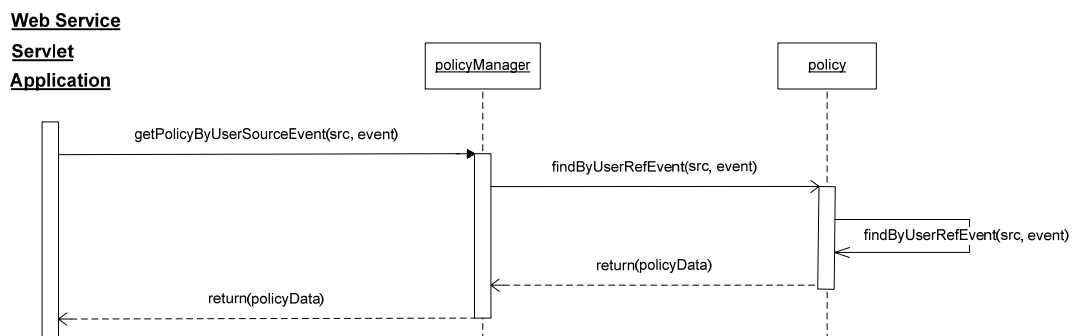


Figure 4-18 UML Sequence Diagram for policyManager.getPolicyByUserSourceEvent

4.2.5 myTrust Persistence Layer

The Persistence Layer in Figure 4-1 is comprised of four MySQL database repositories. For each of these four repositories this sub-section will present a database schema, the full set of MySQL database queries, and an example database instance.

4.2.5.1 Trust Repository

Field	Type	Null	Key	Default	Extra
userSource	varchar(100)	YES			
belief	int(11)				3
faith	int(11)				3
confidence	int(11)				3
credibility	int(11)				3
reputation	int(11)				3
reliability	int(11)				3
honesty	int(11)				3
competency	int(11)				3
userDestination	varchar(100)	YES			
trustRef	int(11)		PRI		auto_increment

Figure 4-19 trustDB Database Schema

Figure 4-19 presents the trustDB database schema. The primary key is an auto-incrementing integer value called trustRef. There are userSource and userDestination string values and a single integer value for each of the eight trust concepts. Note that the default value for these trust concepts is '3', or *high*.

The trust repository is access through the trust entity bean, which resides in the Business Layer. The following three database queries are supplied by the trust entity bean and the returned trustRef value is used to select an instance of trustData.

(1) findByUserSourceDestination(src, dest)

```
"SELECT trustRef FROM trustDB WHERE userSource = 'src' AND userDestination = 'dest'"
```

(2) findByUserSource(src)

```
"SELECT trustRef FROM trustDB WHERE userSource = 'src'"
```

(3) findByUserDestination(dest)

```
"SELECT trustRef FROM trustDB WHERE userDestination = 'dest'"
```

The trustDB database is used to store trust annotation data. In this instance it is used to store opinion based trust annotation data. An instance that corresponds to this database can be seen in Figure 4-20. This example shows trust data that Dave has annotated Austin with. In Figure 4-20 Dave states that he holds Austin in *very high* regard for *confidence*, *reliability*, and *reputation*.

	Instance
userSource	Dave
belief	high
competency	high
confidence	very high
credibility	high
faith	high
honesty	high
reliability	very high
reputation	very high
userDestination	Austin
trustRef	1001

Figure 4-20 Example trustData Instance

4.2.5.2 User Repository


Field	Type	Null	Key	Default	Extra
 userRef	int(11) unsigned		PRI		auto_increment
userID	varchar(100)	YES			

Figure 4-21 userDB Database Schema

Figure 4-21 presents the userDB database schema, which is used by the user entity bean. The primary key is an auto-incrementing integer value called userRef. The only other database value is the userRef string value.

The user repository is access through the user entity bean. The following database query is supplied by the user entity bean and the returned userRef value is an internal value that identifies a *myTrust* user.

(1) findByUserID(userID)

"SELECT userRef FROM userDB WHERE userID = 'userID'"

The userDB database is used to store a *myTrust* userID (a user name) as an anonymous userRef value. An instance that corresponds to this database can be seen in Figure 4-22. In this example the userID value, Dave, has an internal userRef value 1234.

	Instance
userRef	1234
userID	Dave

Figure 4-22 Example userData Instance

4.2.5.3 Personalise Repository


Field	Type	Null	Key	Default	Extra
userSource	varchar(100)				
concept	varchar(100)				
value	float			0	
rank	int(11)			0	
 primkey	int(11)		PRI		auto_increment

Figure 4-23 personaliseDB Database Schema

Figure 4-23 presents the personaliseDB database schema. The primary key is an auto-incrementing integer value called primkey. There are userSource and concept string values, a single integer value denoting a concepts rank, and a float value denoting a concepts weight. Note that the default value for the rank and weight is '0'.

The rank and weight data in the personaliseDB database is calculated for each trust concept in a personalised model of trust by the Personalisation Engine, previously seen in Figure 3-1. The source code for the Personalisation Engine is in APPENDIX III – Implementation Code, Trial Data, and Sundry, under Implementation Code.

The personalise repository is accessed through the personalise entity bean, which resides in the Business Layer. The following database query is supplied by the personalise entity bean and the returned PrimaryKey value is used to select an instance of personaliseData.

(1) findByUserSourceConcept(src, con)

```
"SELECT PrimaryKey FROM personaliseDB WHERE userSource = 'src' AND
concept = 'con'"
```

The personaliseDB database is used to store a personalised model of trust data. An example instance is shown in Figure 4-24, where the trust concept *credibility* is a ranked number 1 with a value, or weighting, of 11.66 for source user Dave.

	Instance
primkey	02102006
userSource	Dave
concept	Credibility
rank	1
value	11.66

Figure 4-24 Example personaliseData Instance

In addition, a personalised model of trust can be converted from MySQL to OWL by a tool that was developed to accomplish this task in order to promote sharing and enable ontological reasoning for personalised models of trust. The source code for this tool is in APPENDIX III – Implementation Code, Trial Data, and Sundry, under Implementation Code.

4.2.5.4 Policy Repository

Field	Type	Null	Key	Default	Extra
policyRef	int(11)		PRI		auto_increment
userRef	int(11)				0
event	varchar(100)				
condition	int(11)				3
action	int(11)				1
reputation	int(11)				0
reliability	int(11)				0
competency	int(11)				0
credibility	int(11)				0
honesty	int(11)				0
belief	int(11)				0
faith	int(11)				0
confidence	int(11)				0

Figure 4-25 policyDB Database Schema

Figure 4-25 presents the policyDB database schema. This repository is only needed to store trust policies for the enhanced IM application only. The primary key is an auto-incrementing integer value called policyRef. There is one single string value that denotes the policy event. For example, this could be ‘Location Information’. A single integer is used to represent the condition of the policy. This integer reflects the level of trust that is required in order to grant (or deny) and action that is associated with an event. The default value for the condition value is ‘3’, or *high*. A single integer is used to represent the action of the policy. This integer is used to state what action should be taken if a condition is satisfied. Its default value is ‘1’, or grant. A value of ‘0’ is used

to deny an action. Each trust concept has an associated integer value that can be used to state additional policy constraints'. However, the default value for these constraints is '0', or not required.

The policy repository is accessed through the policy entity bean, which resides in the Business Layer. Two database queries are provided by the policy entity bean and the returned policyRef value is used to select an instance of policyData.

(1) findByUserRefEvent(userRef, event)

```
"SELECT policyRef FROM policyDB WHERE userRef = 'userRef' AND event = 'event'"
```

(2) findByUserRef (userRef)

```
"SELECT policyRef FROM policyDB WHERE userRef = 'userRef'"
```

The policyDB database is used to store policy data. In Figure 4-26 an example policyData instance is provided. This is Dave's policy (Dave has userRef '1234') for access to his Location Information. It states that the minimum overall trust value required to be granted access to Dave's Location Information is *very high*. In addition, Dave has stipulated that the *reputation* and *belief* trust values of the person seeking access are *very high*. A user that meets these requirements will be granted access but a user with less than *very high* overall trust will be denied.

	Instance
userRef	1234
policyRef	20052006
event	Location Information
condition	<i>very high</i>
action	grant
belief	<i>very high</i>
competency	not used
confidence	not used
credibility	not used
faith	not used
honesty	not used
reliability	not used
reputation	<i>very high</i>

Figure 4-26 Example policyData Instance

4.3 Supporting Technologies

The multi-faceted model of trust that is personalisable and specialisable needs to be represented in a format that is extensible, sharable, computer readable, human understandable, and has the ability to describe relationships and has a rich typing of properties; OWL meets such requirements. However, many more technologies are required to reason about and query such a language. Furthermore, a range of technologies realises the framework in which this model of trust resides.

4.3.1 Model Representation using OWL

OWL, or more specifically OWL Description-Logic (OWL-DL) [OWL-DL], is used extensively in the implementation of the upper ontology, meta-model, and personalised and domain specific models of trust. OWL-DL provides the maximum expressiveness of OWL while also retaining computational completeness (all conclusions are guaranteed to be computed) and decidability (all computations will finish in finite time). OWL can capture the trust concepts in the upper ontology, categorising them as *abstract* or *concrete*, and then define the associated relationships in the meta-model.

The ability to import documents into other OWL documents is used when generating personalised models of trust and also when engineering domain specific models of trust. Domain specialisation can require the modelling of complex application domains and OWL has a vocabulary for describing properties, property characteristics, classes, enumerated classes, relationships, and cardinality that greatly aids this modelling task.

4.3.2 Model Storage using OWL and MySQL

The upper ontology, meta-model, and personalised and domain specific models of trust can all be stored, shared, parsed, and reasoned about as OWL documents. The only one of these models that is not developed in an ontology editor is the personalised model of trust. However, as described in Section 4.2.5.3 a tool has been implemented to automatically transform personalisation data in a MySQL database into an OWL document. The personalisation data, trust data, and trust policies that are used to provide trust based recommendations are stored in a MySQL database. These three sources of data were captured for a combined total of approximately 400 test

subjects (experiment two and three) using a Web based questionnaire. Linking the Web based questionnaire to the MySQL database enabled a simple and efficient way to capture and analyse the results of nearly 400 test subjects. For example, the MySQL database enables efficient calculations to be performed and this was useful from an evaluation viewpoint. Performing the evaluation calculations for approximately 400 test subjects using OWL documents would increase the complexity and computational requirements. Thus, the evaluation of experiments two and three were carried out with the aid of MySQL.

4.3.3 Model Reasoning using Jena

Hewlett Packard's Jena [Jena, HP] reasoner can parse and convert OWL documents into internal models on which it is possible to infer knowledge and reason. Jena was chosen as it is a very popular and is often cited as a software kit for handling OWL documents. The Personalisation Engine in Figure 3-1, that automatically transforms a user's personalisation data into their personalised model of trust, uses Jena extensively.

The Trust Calculation Engine in Figure 4-2 uses Jena to parse domain specific models of trust. These domain models are transformed into internal models and used to identify trust concepts, relationships, sub-classes, and properties. This ability is key to providing personalised and domain specific trust based recommendations.

4.3.4 Model Querying using RDQL

The RDF Data Query Language (RDQL) [RDQL, HP], also developed by Hewlett Packard, is used to query instances of an OWL document. RDQL could be used to query instances of personalised models of trust. For example, an RDQL query could be used to ascertain the rank and weight of a particular trust concept for a given user. However, it is possible to do the very same query using Jena.

There are circumstances in which RDQL can be used to answer queries that OWL can not reason about. For example, if Daniel's father Thomas has a brother Declan it can be said that Daniel's uncle is Declan. OWL can not combine the 'parent and brother' property with the 'uncle' property. However, an RDQL query will provide a solution.

Such a challenge arose in the domain of Web Services when the author of this thesis was developing the Web Service domain model with Ericsson Research Group, Ireland, in 2004. An RDQL solution was implemented to resolve the issue. The solution was to create two queries and combine them as one. Using the example above the query first sought parents of Daniel. Then the query sought brothers of Thomas. In this way the query returned Daniel's uncle, Declan.

4.3.5 Model Access using JBoss, EJB's, Servlets, and Web Services

The JBoss Application Server was used to host the implemented *myTrust* Enterprise Java Bean's, Servlets, and Web Services as described earlier in Section 4.2. The JBoss Application Server resides on a Dell Inspiron 8600 laptop, which has a 1.5 GHz Pentium M CPU, 768 Megabytes of RAM, 60 Gigabyte HDD, and a LAN Ethernet connection. The CBPM system and the enhanced IM application can use the EJB's that are hosted on the JBoss server to logically access the MySQL database. In addition, it is this system that ran the Eclipse [Eclipse] open source Integrated Development Environment (IDE) in which all source code development took place.

4.4 Summary

This chapter discussed the implementation of the trust management service *myTrust*, which uses the multi-faceted model of trust described in chapter 3. The implemented architecture comprises of an Application Layer, Service Layer, Business Layer, and Persistence Layer. The interface mechanism adopted by the CBPM and enhanced IM application were described and analysed. The Web Service Layer and Servlets Layer were addressed. The interactions between the session and entity beans were mapped as UML sequence diagrams. The database repositories were also described.

The next chapter looks at the evaluation of the multi-faceted model of trust and the *myTrust* trust management service.

5 EVALUATION

5.1 Introduction

The research question posed in this thesis is whether a multi-faceted model of trust that is personalisable and specialisable is both necessary and accurate to the user in providing a dynamic and flexible trust based decision support mechanism within Internet environments. It was decided to evaluate the necessity for a multi-faceted model of trust, the necessity for personalisation, and accuracy of the model to the user through a series of web based surveys. The advantage of a web based survey approach was that it enabled the automated collection of user feedback from a broad population. Four experiments were conducted. In sections 5.2 to 5.5 these are described under the headings of goals, hypothesis, overview, results, and findings.

Illustrating the need for specialisation within the model of trust was made through the implementation and analysis of multiple domain specific models, which were presented in Section 3.4.4. Illustrating that the multi-faceted, personalisable, and specialisable model of trust can provide a dynamic and flexible trust based decision support mechanism is made through several trials, which are detailed in Chapter 6.

5.1.1 Evaluating the Multi-faceted, Personalisable Model of Trust

A multi-faceted approach to modelling trust is distinctly lacking across the current state of the art, as is personalisation within models of trust. Evaluating the necessity for a personalised, multi-faceted approach is paramount to answering the research question posed in this thesis and critical to the arguments supporting such a model. In experiment one the necessity for a multi-faceted model of trust was evaluated through a survey that allowed each participating subject to rank a variety of trust concepts that are commonly found in single-faceted models of trust. The rankings provided by each subject were used to evaluate the necessity for personalisation within the multi-faceted model of trust at the individual level and across a broad population. In addition to evaluating the necessity for a multi-faceted and personalised approach experiment one also addressed the hypothesis reflected in the model that some trust concepts are *abstract* and others are *concrete*. Experiment one is described in Section 5.2 along with the results and findings.

5.1.2 Evaluating the Accuracy of Recommendations

Evaluating the accuracy of trust based recommendations made using the multi-faceted model of trust was carried out in experiment two and three. These evaluations were conducted through web based surveys that enabled participating subjects to build a personalised model of trust, which was subsequently used to provide personalised recommendations. Please note that the web based surveys gathered personalisation data that was used by the Personalisation Engine (see Figure 3-1) to generate a personalised model of trust for each subject using the HITS algorithm.

These recommendations were cross-referenced with an answer set provided by the same subject in the same experiment. In this way the accuracy of calculated trust recommendations could be measured against a set of answers supplied by subjects. Experiment two is described in Section 5.3 along with the results and findings. Experiment three is described in Section 5.4 along with the results and findings.

5.1.3 Evaluating Trust Concept Clarity

Experiment four evaluated whether there is a link between the ranking of trust concepts, over a broad population, and the clarity of these concepts. Examining the trust concepts in terms of their clarity, as perceived by the subjects who took part in survey three, also addresses the hypothesis that trust concepts can be categorised as *abstract* or *concrete*. Experiment four is described in Section 5.5 along with the results and findings.

5.2 Experiment One- Necessity for a Multi-faceted, Personalisable Model of Trust

In the introduction chapter of this thesis the third goal that was derived to evaluate the research question stated that it was necessary to “Evaluate the necessity for a multi-faceted model of trust that is personalisable and specialisable”. This experiment has been designed to address the personalisation aspect of this goal. The state of the art conducted prior to the design and development of the model of trust as an ontology suggested eight trust concepts that should be in the model; *belief*, *competency*, *confidence*, *credibility*, *faith*, *honesty*, *reliability*, and *reputation*. This experiment evaluated how useful these concepts are to the subjects across three scenarios. In designing the model of trust it was hypothesised that some trust concepts were well-defined and based more on evidence, and therefore categorised as *concrete*, while others were open to interpretation and based more on opinion, and therefore categorised as *abstract*. This experiment evaluated the correctness of this *concrete* and *abstract* hypothesis and the subsequent identification of trust concepts as either *abstract* or *concrete*. At design time it was also hypothesised that the trust concepts influence each other to some degree as viewed by an individual and that the degree of influence would most likely be different from person to person. Evaluation into whether there is any influence exerted between concepts in relation to the top three trust concepts chosen by each subject was also conducted. Furthermore, evaluation into whether personalisation within a model of trust is seen in the overall group of subjects taking part in the experiment was also carried out. Each of the three scenarios presented to the subject represents a change in risk, which may have an effect on an individual’s personalised model. The final evaluation tried to ascertain the level of change in an individual’s model across all three scenarios.

5.2.1 Experiment Goals

The goals for this experiment were to (i) evaluate the upper ontology, (ii) evaluate the meta-model, (iii) evaluate the need for personalisation, and (iv) examine how individual models of trust alter as risk changes.

5.2.2 Hypotheses

Several hypotheses were derived that directly relate to each of the evaluation goals. The first hypothesis was derived from the first goal and it is that the trust concepts found in the upper ontology are (a) useful to some extent to subjects and (b) that subsets of these concepts could be classified as either *abstract* or *concrete*. From goal two the hypothesis to be tested is that the model of trust had different strength relationships that linked, and interlinked, *abstract* and *concrete* concepts. From goal three and four the hypothesis to be tested is that personalisation is required when modelling trust, and that an individual's model of trust alters as risk changes.

5.2.3 Experiment Overview

The questionnaire was designed in association with Dr. Deirdre Bonini, Psychology Department, Trinity College Dublin. The experiment was specifically designed and developed to meet the evaluation goals listed above. The questionnaire was comprised of three simple scenarios in which the subject was asked to rate the set of eight trust concepts. Each concept was rated on the basis of how useful the subject thought the concept is when determining a level of trust specific to each scenario. This enabled the evaluation to address the hypothesis that the trust concepts are useful to subjects. Scenario one presented a low risk scenario in which the subject was buying an item online for \$10, scenario two was medium risk at \$100, and scenario three was high risk at \$1000. The three scenarios, with increasing risk to the subject, enabled the evaluation to address the hypothesis that a personalised model of trust alters as risk increases. This is done by measuring the level of change, if any, found in a subject's personalised model of trust over the three scenarios. The only difference between each scenario was the level of risk involved. The subjects were informed that no credit card fraud was involved.

For each scenario the subject was asked to complete three stages. Firstly, the subject was asked to scale each of the trust concepts in terms of usefulness from one to five on a Likert scale; one representing *very low*, two *low*, three *no opinion*, four *high*, and five *very high*. Secondly, the full set of trust concepts was presented and the subject was asked to rank the three concepts they considered most important in relation to determining how much the subject trusted the seller. Finally, the subject was asked to choose a trust concept that most influences each of their chosen top three concepts. Only the top three concepts were chosen as this provided enough data to conduct analysis without requiring the test subject to repetitively answer the same question for trust concepts that had less and less importance to them. The Likert scales, rank, and influence results were used to analyse the experiment hypotheses posed in Section 5.2.2.

The subjects were offered the opportunity to take part in a competition to win two tickets to a U2 Vertigo Tour 2005 concert in Dublin, Ireland to entice their participation. Advertisement for the participants was conducted via email, to a wide range of faculties within Trinity College Dublin, including Computer Science, Psychology, Dentistry, Zoology, and Arts at undergraduate, postgraduate, and staff levels. In addition Ericsson Research Group, Ireland received emails, and notes were posted on forums such as trustcomp.org and U2.com.

In total 279 fully completed questionnaires were received, which were analysed with the aid of the Statistical Package for Social Sciences (SPSS). It can be ascertained from email addresses that approximately 45% (126) of 277 of these subjects were from Trinity College Dublin. In addition, approximately 14% (40) of the 277 of these subjects were from the Computer Science department, Trinity College Dublin. The remaining 41% (113) of test subject did not provide a Trinity College Dublin email address, or a Computer Science (TCD) email address.

The subjects were also asked to complete a three question personal survey that anonymously recorded their age (see Figure 5-1), sex (see Figure 5-2), and asked whether they had any history of online purchases (see Figure 5-3). Please note that RNS* in all three figures represents subjects who answered ‘Rather Not Say’ to each question.

Age Group	Number	Percentage
<20	17	6.1
20-29	176	63.1
30-39	48	17.2
40-49	20	7.2
>50	12	4.3
RNS*	4	1.4

Figure 5-1 Participant Age Demographics

Sex	Number	Percentage
Male	158	56.6
Female	117	41.9
RNS*	2	0.7

Figure 5-2 Participant Sex Demographics

Answer	Number	Percentage
Yes	255	91.4
No	22	7.9
RNS*	-	-

Figure 5-3 Online Purchase History

The experiment questionnaire can be viewed in APPENDIX II – Research Experiments, Experiment One; Necessity of Personalisable, Multi-Faceted Approach. Please note that in this appendix it is only the \$1000 scenario that is presented as the \$10 and \$100 are similar in every way except monetary value.

5.2.4 Results

This section presents the results that were attained from the experiment, the next section provides analysis of these results. Furthermore, the entire set of anonymised data records are provided in the accompanying DVD media under Experimental Datasets - Experiment 1.

Scenario One (\$10)

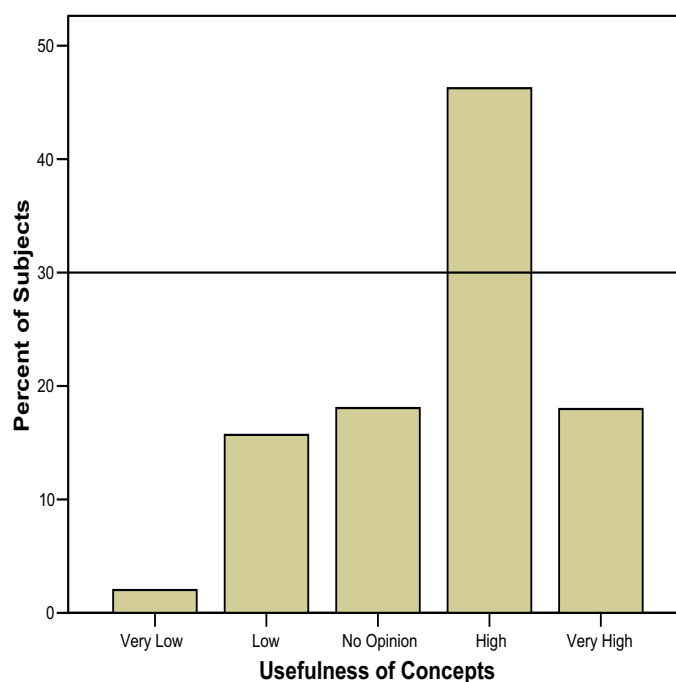


Figure 5-4 Likert Scales for all Concepts for Scenario One (\$10)

Each subject was asked to rate how useful each trust concept is to them when determining trust in the \$10 scenario. The data from the experiments found that, as per Figure 5-4, 64.3% of the subjects view the set of trust concepts as *high*, or *very high* in terms of usefulness in determining trust in the \$10, or least risk, scenario.

Scenario Two (\$100)

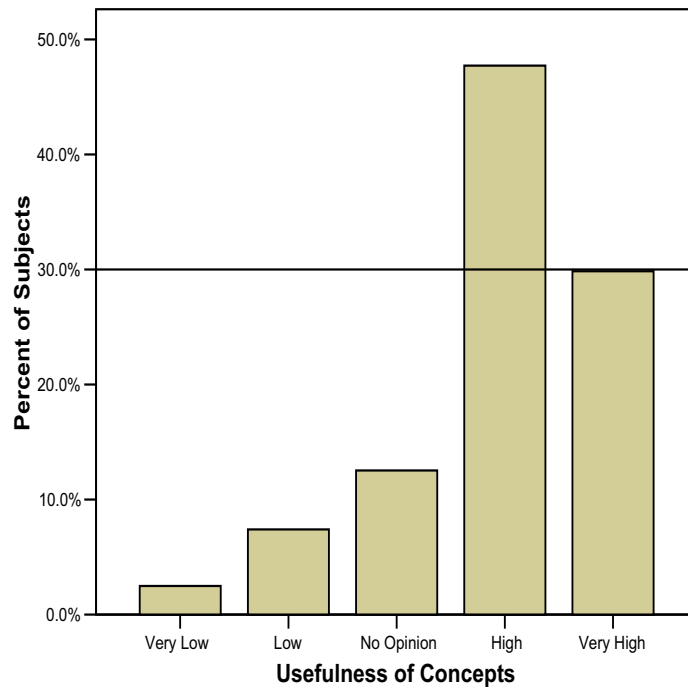


Figure 5-5 Likert Scales for all Concepts for Scenario Two (\$100)

Each subject was asked to rate how useful each trust concept was to them when determining trust in the \$100 scenario. In Figure 5-5 it is 76.9% of the subjects that view the set of trust concepts as *high*, or *very high* in terms of usefulness in determining trust in the \$100, or medium risk, scenario.

Scenario Three (£1000)

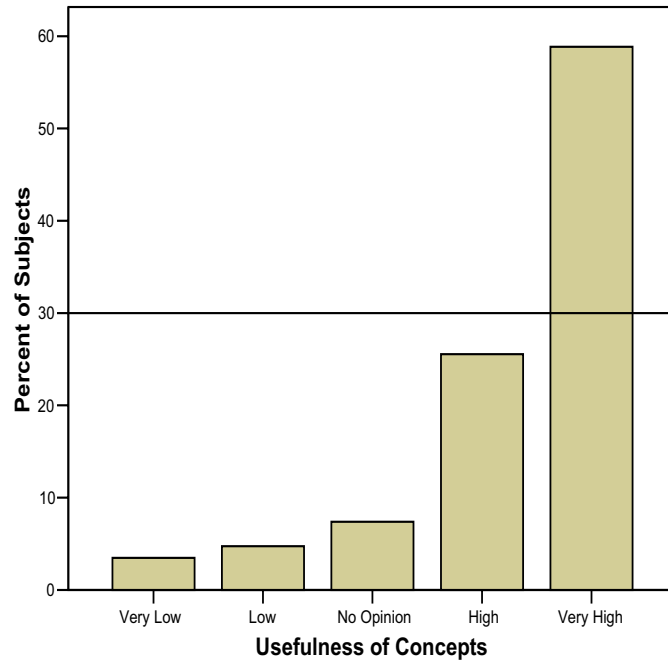


Figure 5-6 Likert Scales for all Concepts for Scenario Three (\$1000)

Each subject was asked to rate how useful each trust concept was to them when determining trust in the \$1000 scenario. In Figure 5-6, it is 83.9% of subjects that view the set of trust concepts as *high* or *very high* in terms of usefulness in determining trust in the \$1000, or high risk, scenario.

Figure 5-7 illustrates the total of number one rankings that each concept received in the highest risk scenario; \$1000 scenario where the trust concepts have the highest percentage of *high* and *very high* usefulness scores.

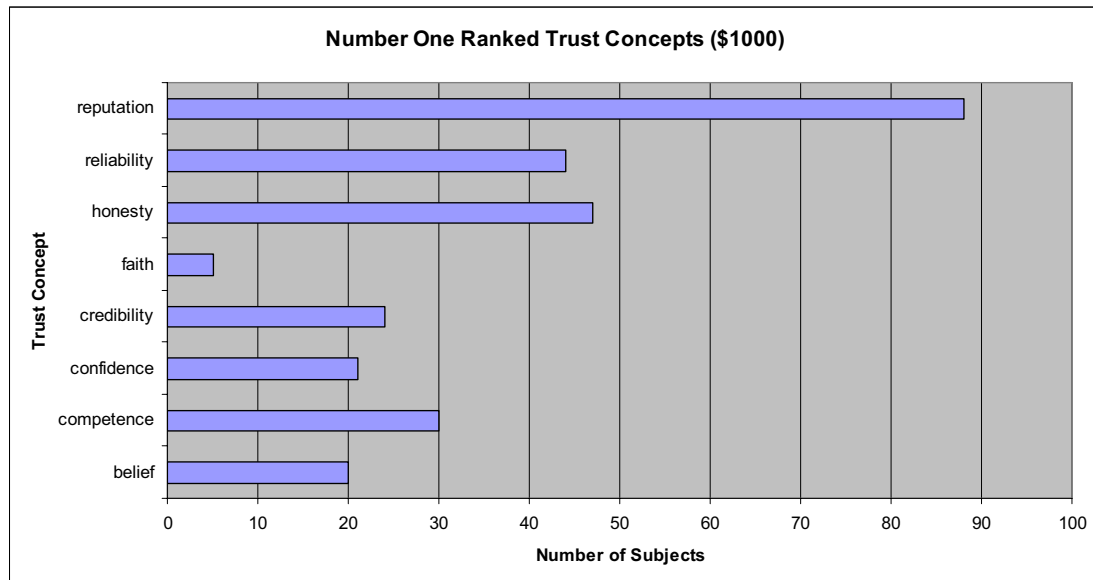


Figure 5-7 Number One Ranked Trust Concepts by Subjects in \$1000 Scenario

In Figure 5-7, the concept *reputation* has the most number one rankings. It is interesting to note that the state of the art review of trust management systems showed that *reputation* is the most dominate synonym for trust in single-faceted approaches. These systems include; eBay, Amazon, OpenPrivacy, Trellis, and FilmTrust. The concepts *honesty* and *reliability* have the second and third highest amount of number one votes, respectively. In addition, the concept *faith* received the least amount of number one rankings, closely followed by *belief* with the second least number one rankings.

It is important to note that the results show that across all three scenarios only one concept, *reputation*, retains the most number one rankings, and concepts that have the second most (*honesty*) and third most (*reliability*) number one rankings also do not change. In addition, the concept *faith* retained the least amount of number one rankings across all three scenarios. However, the concept with the second least number one rankings, across all three scenarios, is *confidence* (*belief* has the third least number one votes).

The frequencies, Likert scores, and influence data in Figure 5-8 were tabulated from results attained across all three scenarios; \$10, \$100, and \$1000 results combined.

- (i) The concepts with the lowest frequencies across all three ranking scores are *faith* (3.7%), *confidence* (7%), and *belief* (8.8%),
- (ii) The concepts with the highest amount of *very low* and *low* Likert scores are *belief* (19%) and *faith* (16.5%), and
- (iii) The least influential concepts across are *faith* (3.8%), *belief* (7.2%), and *confidence* (7.9%).

Figure 5-8 Abstract Concept Table

Figure 5-9 shows the percentage of influencing concepts that were based on results attained across all three scenarios. It can be read as follows; *credibility* is influenced by *reputation* 115 times over all three scenarios as illustrated in the shaded region.

	Belief	competency	confidence	credibility	faith	honesty	reliability	reputation
belief	X	15	16	25	41	47	12	25
competence	42	X	29	34	6	13	148	50
confidence	13	34	X	40	9	26	33	43
credibility	48	52	36	X	10	85	63	127
faith	18	6	8	8	X	34	12	10
honesty	50	17	25	52	16	X	40	57
reliability	25	120	26	55	4	76	X	159
reputation	26	80	35	115	10	100	193	X

Figure 5-9 Concept A Influenced By Concept B

The data in Figure 5-10 has been used to calculate overall percentages in Figure 5-9.

	Concrete	Abstract
Concrete	81.6%	78.7%
Abstract	18.4%	21.3%

Figure 5-10 Percentage of Influencing Concepts

The results, illustrated in Figure 5-10, found that 81.6% of *concrete* concepts are influenced by *concrete* concepts, with the remaining 18.4% of influencing concepts being *abstract*. There were similar results for what concepts influence *abstract* concepts. Figure 5-10 shows that 78.7% of *abstract* concepts are influenced by *concrete* concepts with the remaining 21.3% of influential concepts being *abstract*.

Across all three scenarios a very strong two-way interdependence between *reliability* and *reputation* (193/159) was found; 193 subjects stated that *reliability* is influenced by *reputation* and 159 subjects stated that *reputation* is influenced by *reliability*. Taking that the two highest influencing concepts in the \$1000 scenario are *reputation* and *reliability* the data illustrates that the bonds between *reliability* and *reputation* actually increase in strength as risk increases over the three scenarios; 57/49, 64/50, 72/60 respectively. Across the same three scenarios strong two-way interdependences between *reliability* and *competence* (148/120), *reputation* and *credibility* (127/115), and *honesty* and *reputation* (100/57) were found. In contrast, other two-way interdependent relationships that exist outside of the three most popular influential concepts vary from 9/8 (*faith* and *confidence*) to 52/34 (*competence* and *credibility*).

	Scenario 1 (\$10)	Scenario 2 (\$100)	Scenario 3 (\$1000)
Rank 1	5.02%	7.9%	12.19%
Rank 2	5.37%	8.24%	9.68%
Rank 3	3.58%	4.3%	5.37%

Figure 5-11 Maximum Percentage of Subjects with Matching Rank, Influence, and Likert Scales

The data collected from the experiment was used to investigate what percentage of subjects has similar models of trust. In order to do this the choices that each subject made for concepts ranked one to three, over all three scenarios, were cross-referenced with the choices that all the subjects made. As per Figure 5-11 the results show that only 5% of subjects applied the same Likert scale and influence to the number one ranked concept in the \$10 scenario. This figure rose to 7.9% and 12.19% in the \$100 and \$1000 scenarios, respectively. Although more than twice as many subjects had matching choices in the \$1000 scenario, with respect to the \$10 scenario, the evaluation shows that almost nine out of ten people had made different choices. There were similar figures for the concepts ranked two and three.

5.2.5 Key Findings

The key findings are presented in four sections that reflect the experiment hypothesis outlined in Section 5.2.2. The results presented in Section 5.2.4 are the basis for the analysis found in this section. Please also note that the analysis is based on the findings from a group of subjects that, over 80% of the time, are aged between 20 and 39. In addition, 90% of all subjects have an online purchase history. However, the subject set is relatively large and broad with 279 subjects, multiple disciplines, and a fairly even gender balance.

5.2.5.1 Trust Concepts: Usefulness and *Concrete* or *Abstract* Categorisation

The SPSS analysis found a positive correlation, 0.318 at the 0.01 level of significance, between increasing risk and the increasing Likert scales the subjects provided for the trust concepts. Therefore, it can be said with confidence that this SPSS analysis suggests that as risk increases so too does the subjects regard for usefulness of the trust concepts rise. This can be seen in Figure 5-12.

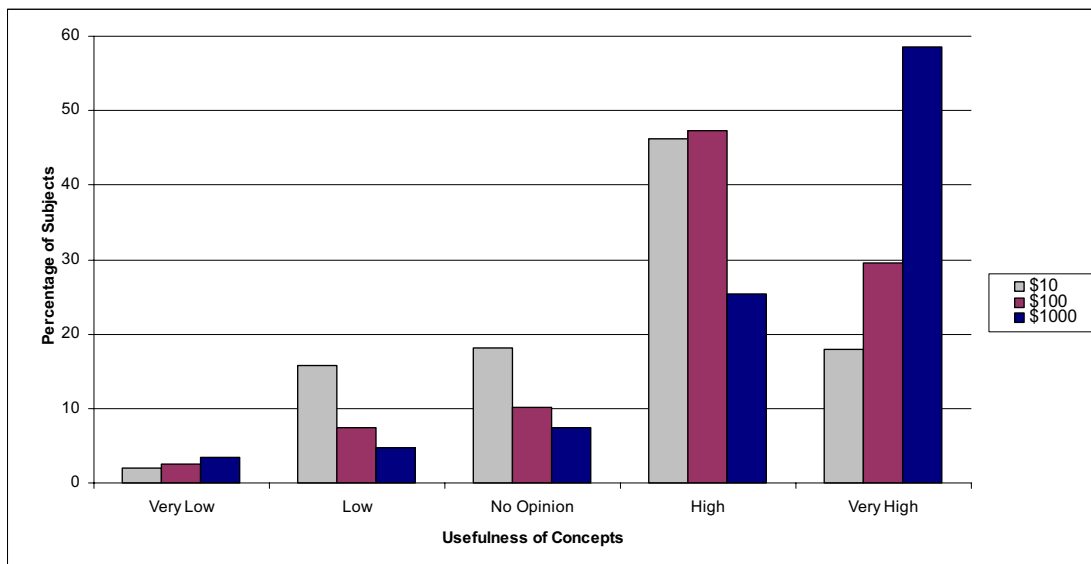


Figure 5-12 Likert Scales for all Concepts for All Scenarios

Figure 5-12 is generated from the data provided in Figure 5-4 (\$10 scenario), Figure 5-5 (\$100 scenario), and Figure 5-6 (\$1000 scenario). The progression observed over the three scenarios is shown in Figure 5-12. There is a reduction in the percentage of subjects who found the trust concepts to be of *low* or *very low* usefulness as seen in the \$10 scenario where there is a significant number of *low* and *very low* usefulness

scores (17.7%), which reduces continuously through the \$100 scenario (9.9%) to the \$1000 scenario (8.2%). However, this downward decline in the combination of *very low* and *low* usefulness scores is noticed only when individually considering *low* usefulness scores. The *very low* usefulness scores marginally increase as risk increases from \$10 (2%) through \$100 (2.5%) to \$1000 (3.5%). The percentage of subjects who select *no opinion* diminishes as risk increases from \$10 (18.1%) through \$100 (10.2%) to \$1000 (7.4%). Over the three scenarios the percentage of subjects who select *high* or *very high* increases from 64.3%, through 76.9%, to 83.9%, yet it is only the *very high* scenario that increases continuously. The Likert scale with the highest percentage of subjects shifts from *high* to *very high* as risk increases.

This experiment data and its analysis suggests that people are more willing to examine all trust concepts as risk increases, while at the same time the concepts become more important as risk increases. Therefore, this addresses the hypothesis that the trust concepts currently found in the upper ontology are useful to subjects. However, it is important to note that later analysis reveals that certain trust concepts are more useful to subjects than others, namely *concrete* and *abstract* concepts.

The expected difference between the *abstract* and *concrete* concepts was re-enforced in the experiment data found in Figure 5-8 regarding rank, Likert scales, and influence. In addition, clear differences exist in the amount of number one rankings that each trust concept received, as illustrated in Figure 5-7. The differences found in Figure 5-7 and Figure 5-8 can be attributed to *concrete* and *abstract* typing of trust concepts. It is argued that the subjects do not see *abstract* concepts, such as *belief* and *faith*, as well-defined and measured as they attribute *low* or *very low* scale scores, *low* or *very low* ranking, and *low* or *very low* influence to them. The influence of *abstract* concepts (3.8%, 7.2%, and 7.9%) falls far below the random influence average of 12.5% (100% divided by 8, for each trust concept), whereas concrete concepts meet, or surpass, this average. In contrast, *concrete* concepts, such as *reputation*, *reliability*, and *credibility* score highest in rank, influence, and on the Likert scale. The *high* and *very high* Likert scales received for the usefulness of *concrete* concepts as well as the highest scores in rank can lead to the conclusion that *concrete* concepts have a greater impact on overall trust than *abstract* concepts. These differences support the hypothesis that certain trust concepts are *abstract* while others are *concrete*.

5.2.5.2 Meta-Model: Different Strength Relationships

At design time it was decided that a meta-model would be required to govern the interactions between *abstract* and *concrete* trust concepts. The meta-model established in Figure 5-13 illustrates the three relationships that can exist between two concept types. The relationships exist between concepts of the same type as well as between different concept types.

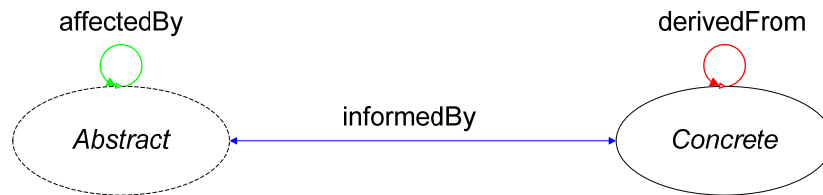


Figure 5-13 Trust Meta-Model

The *derivedFrom* relationship can only exist between *concrete* concepts and the *affectedBy* relationship can only exist between *abstract* concepts. The two different concepts types can be linked together via the bidirectional *informedBy* relationship.

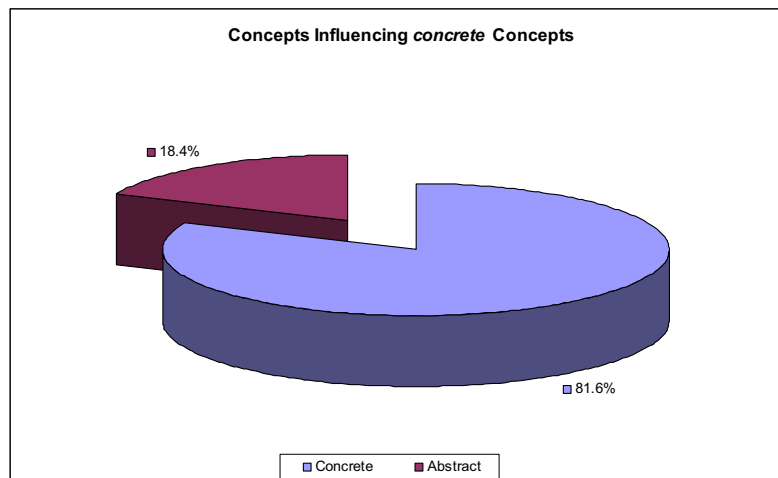


Figure 5-14 Concepts Influencing *concrete* Concepts

The results illustrated in Figure 5-10 suggest that there is a very strong and consistent influence between *concrete* and *concrete* trust concepts. As per Figure 5-14, which is based on the data in Figure 5-10 it can be seen that 81.6% of subjects state that *concrete* concepts are influenced by other *concrete* concepts. This provides sufficient influence to suggest a *derivedFrom* type of relationship.

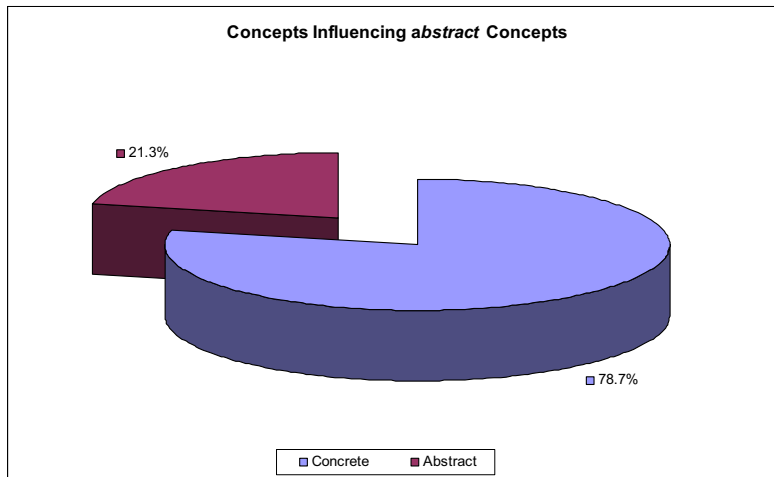


Figure 5-15 Concepts Influencing *abstract* Concepts

In Figure 5-15 (based on the data in Figure 5-10) it can be seen that one in five subjects state that *abstract* concepts are influenced by other *abstract* concept. Therefore, sufficient influence exists between *abstract* and *abstract* concepts, 21.3%, to suggest an *affectedBy* type of relationship.

In Figure 5-14 it is shown that 18.4% of subjects state that *concrete* concepts are influenced by *abstract* concepts, and in Figure 5-15 it is shown that 78.7% of subjects state that *abstract* concepts are influenced by *concrete* concepts. There is sufficient influence to suggest an *informedBy* type of relationship.

The difference in influence between *concrete* and *concrete* concepts, between *abstract* and *abstract* concepts, and also between the two different types of concept suggests that the three possible relationships are required in order to reflect the differences in influence, as per the hypothesis.

5.2.5.3 Necessity for Personalisation

The statistics from the results show that the subjects have general similarities when applying Likert scales to the trust concepts (75% *high* or *very high* usefulness over all three scenarios), and also in viewing *concrete* concepts as being favoured more in rankings and in influence. However, experiment results indicate that individual differences in the subjects scale, rank, and influence of concepts makes a personalised approach necessary. The most highly ranked concept across all three scenarios is *reputation* with an average of 82 number one votes, yet this tally of votes reflects 29% of the overall vote; 71% think differently. There are also similar and reduced

percentages for the number two and number three top ranked concepts respectively. The results presented in Figure 5-11 state that only one person in twenty applies the same Likert scale and influence to the concept ranked first in the \$10 scenario. However, that figure increases continuously to one in twenty for concept ranked first in the \$1000 scenario. There are similar trends for Likert scale and influence pairs in concepts ranked number two and three. In addition, highly ranked trust concepts within Likert scales and influences (taken separately) do not necessarily reflect the entire broad spectrum of subjects. The combination of the similarities and differences found within rankings, Likert scores, and influences strongly suggests that personalisation is required in modelling trust, thus supporting the original hypothesis.

5.2.5.4 Trust Model Alteration as Risk Increases

Statistically, 55% of all subjects altered their top ranked concept at least once over the three scenarios, with 40% changing from \$10 to \$100, and 40% changing from \$100 to \$1000 scenarios. Furthermore, 71% also make at least one alteration in concepts ranked number two over all three scenarios, and in terms of concepts ranked number three 73% make some alteration over all three scenarios. Therefore, the probability exists that a user's personalised model of trust alters in some way as risk increases over all three scenarios. This supports the hypothesis that an individual's model of trust alters as risk changes.

5.3 Experiment Two- Accuracy of Model of Trust

In the introduction chapter of this thesis the fourth goal that was derived to evaluate the research question stated that it was necessary to “Evaluate the accuracy of trust based recommendations calculated using the developed trust management service”. This experiment has been designed to address this goal.

5.3.1 Experiment Goals

The goals of this evaluation were to evaluate (i) the accuracy of recommendations based on the multi-faceted model of trust, and (ii) where this accuracy is satisfactorily high and where the accuracy is less than desired.

5.3.2 Hypotheses

The hypothesis of this experiment was that the accuracy of recommendations calculated through the multi-faceted model of trust would alter as the risk associated with these scenarios varied. Second, the hypothesis stated that lowest risk recommendations would have greater accuracy than recommendations with higher risk associated with them. It is important to note that the second part of the hypothesis was based on the author’s intuition, which believed that people find it easy to make decisions in lower levels of risk than at higher levels of risk where decisions are not as obvious. Therefore, the hypothesis stated that as risk increases the accuracy of the recommendations would decrease.

5.3.3 Experiment Overview

Experiment two was specifically designed and developed to meet the two evaluation goals listed above. The web based survey for experiment two was comprised of four sections. In addition, two brief follow up surveys that were based on section four of experiment two were issued seven and fourteen days after experiment two began. The seven day survey was carried out in order to get a community view of what actions would be granted to people with different levels of trust. This information could be used as a candidate solution that might increase the overall accuracy of recommendations by filtering recommendations where the accuracy is less than desired. The fourteen day survey was carried out to see if, and how, the original set of decisions changed over fourteen days.

In the first section of experiment two's survey each of the eight trust concepts was presented individually and sequentially to the subject. For each presented trust concept the subject was asked to choose which, if any, of the remaining seven trust concepts are influenced by the presented trust concept. This process was repeated for all eight trust concepts. This gathered personalisation data that was used to generate a personalised model of trust for each subject that took part in experiment two.

In the second section of the survey the subject was presented with four different actors and asked to annotate each of these four actors with trust information, which represented the eight trust concepts found in the upper ontology. The four actors that were presented are *family member*, *work colleague*, *friend of a friend*, and *complete stranger*. The subject was asked to think of a real person that they know and who fits the actor presented, for example their actual mother, father, or sibling for *family member*. The subject was also told that these people will be used later in the questionnaire. As each actor was presented to the subject that subject was asked to select and apply a rating for each of the eight trust concepts with respect to the presented actor. The subject could rate each of the eight trust concepts on one of four scales; *very low*, *low*, *high*, and *very high*, for example a subject may state that *family member* has *very high* trust rating for *reputation*. This provided a set of trust data for each subject that was used to calculate an overall trust value for each actor with respect to that subject.

The third section asked the subject to create a single rule for each of the four targets that they were presented with. These four targets were *pencil*, *bank pin*, *laptop*, and *mobile phone*. For each of the four rules the subject was asked to assign a minimum level of trust that they would require in order to allow the presented target to be borrowed or known (for *bank pin*). For each of these actions they assigned one of four scales; *very low*, *low*, *high*, and *very high* to create a rule for that given action (for example minimum level of trust required to borrow *mobile phone* is *high*). In this way each subject created an action policy, which stated the minimum amount of trust that an actor needed to be granted that action.

In section four the subject was asked whether or not they would allow each of the four actions to take place with respect to each of the four actors that the subject had in mind from section two. Each of the four actions was presented to the subject along with the four actors. The subject had to decide for all 16 actor and action combinations whether they would allow, or not allow, the action to be granted. The answers provided for these 16 actor and action combinations was compared with (i) the calculated overall trust value made in section two, and (ii) the policy for a specific action that was created in section three. These comparisons between answers and recommendations were used to derive the accuracy of trust based recommendations.

A competition to win two tickets to a Robbie Williams concert in Dublin, Ireland was used to entice participation in experiment two. Advertisement for participation was conducted via email, to a wide range of faculties within Trinity College Dublin, mainly Computer Science at undergraduate, postgraduate, and staff levels. In addition the Ericsson Research Group, Ireland received emails and notes were posted on forums such as trustcomp.org and Robbie Williams fan websites. In total, this subject cachement domain yielded 282 fully completed questionnaires were received over a two week period. It can be ascertained from email addresses that approximately 66% (187) of these 282 subjects were from Trinity College Dublin. In total, approximately 17% (48) of the 282 subjects were from the Computer Science department, Trinity College Dublin. Therefore, the remaining 17% of test subject did not provide either a Trinity College Dublin or a Computer Science email address. The experiment questionnaire can be viewed in APPENDIX II – Research Experiments, Experiment Two; Accuracy Survey.

Seven days after the release of the experiment two another email was sent to the same subject cachement domain. They were asked to provide a set of answers to a follow up survey that was similar to section four of the original survey seven days prior. This follow up survey was issued in order to ascertain what actions the test subject community at large would grant to people with different levels of trust. For example, would the community grant access to a *mobile phone* to a person with *very low* trust? The seven day follow up can be viewed in the last section of APPENDIX II – Research Experiments, Experiment Two; Accuracy Survey.

Fourteen days after the release of experiment two a smaller set of ten original experiment two test subjects, who were known to the author and based in Trinity College Dublin, were chosen to answer an exact replica of section four of the original survey. From this follow up survey it was possible to ascertain how many decisions a test subject might change for all 16 actor and action combinations after fourteen days.

5.3.4 Results

Figure 5-16 presents the data received from all 282 subjects who took part in the original experiment two survey, and who provided a fully completed questionnaire.

Required Trust	Calculated Trust Value	Number of Recommendations	Correct (Allowed)	Correct (Not Allowed)	Incorrect (Allowed)	Incorrect (Not Allowed)	Percentage Correct
Overall Very High	Very High	323	216	0	0	107	66.9%
	High	819	0	444	375	0	54.2%
	Low	517	0	449	68	0	86.9%
	Very Low	201	0	189	12	0	94%
Overall High	Very High	177	160	0	0	17	90.4%
	High	535	416	0	0	119	77.7%
	Low	313	0	188	125	0	60%
	Very Low	99	0	84	15	0	84.8%
Overall Low	Very High	86	84	0	0	2	97.7%
	High	269	238	0	0	31	88.5%
	Low	174	128	0	0	46	73.6%
	Very Low	55	0	31	24	0	56.4%
Overall Very Low	Very High	150	149	0	0	1	99.3%
	High	425	411	0	0	14	96.7%
	Low	272	253	0	0	19	93%
	Very Low	97	89	0	0	8	91.8%
Totals	Very High	736	609	0	0	127	
	High	2048	1065	444	375	164	
	Low	1276	381	637	193	65	
	Very Low	452	89	304	51	8	
		4512	2144	1385	619	364	
Percentage			47.52%	30.70%	13.72%	8.07%	
Total Percentage			78.21%		21.79%		

Figure 5-16 Total Recommendation Accuracy in Experiment one

The data in Figure 5-16 has been sorted according to the four possible levels of trust that a subject might require ('Required Trust' column) for any given action. The levels of trust that can be required are *very low*, *low*, *high*, and *very high*. At each of these required trust levels the data is broken down further into the four possible calculated trust levels ('Calculated Trust Value' column). In this way the required trust for an action is correlated to the calculated trust for the associated actor for all 16 combinations. The number of recommendations made using the personalised model of trust for each of the 16 actor and action combinations are also presented ('Number of Recommendations' column). The numbers of correct and incorrect recommendations made with respect to answers provided by the subject are presented over four columns for all 16 combinations. Finally, the percentage of correct recommendation is provided for all 16 combinations (Percentage Correct column).

An analysis of Figure 5-16 shows that the majority of incorrect recommendations, 79.9%, in conditions where the subject did not allow the action to take place occur where calculated trust is either *high* or *very high*. In these cases the subject had provided an answer that did not allow the associated action to take place yet the calculated recommendation did allow the action to take place. Note that in both cases the calculated trust was equal to, or greater than, the required trust. The total percentage of incorrect answers, in conditions where the subject did not allow the action to take place, accounts for 8.07% of overall recommendations.

From Figure 5-16, the majority of incorrect recommendations, 91.7%, in conditions where the subject did allow the action to take place occur where calculated trust is *low* or *high*. Furthermore, the majority of incorrect recommendations, 60.6%, in conditions where the subject did allow the action to take place occur where the calculated trust is *high*. In these cases the subject had provided an answer that did allow the associated action to take place yet the calculated recommendation did not allow the action to take place. The total percentage of incorrect answers, where the subject did allow the action to take place, accounts for 13.72% of overall recommendations.

Across all 16 possible actor and action combinations, and for all 282 subjects (4512 total recommendations), the total number of incorrect recommendations provided to the subject across all required trust levels was 983, or 21.79%. Therefore, the total percentage of correct recommendations out of 4512 recommendations across all required trust levels was 3529, or 78.21%.

5.3.4.1 Seven Day Follow up Survey

The same subject cachement domain, as used in the original experiment two survey, yielded 208 test subjects in the seven day follow up survey. In this survey the members of the test subject community were presented with each action; *pencil*, *mobile*, *laptop*, *bank pin* and asked to decide which of four people they would **not** grant the action to. The four people were *very low* trusted person, *low* trusted person, *high* trusted person, and *very high* trusted person.

	Bank	Laptop	Mobile	Pencil
Very High Trust	31.7%	4.8%	3.4%	2.4%
High Trust	57.2%	33.2%	8.2%	3.4%
Low Trust	72.1%	71.7%	49.5%	7.7%
Very Low Trust	75.0%	73.6%	76.4%	36.1%

Figure 5-17 Test Subject Community Result Set

This follow up survey enabled the creation of an overall test subject community policy, which was derived from the answers provided by the 208 test subjects. This community policy could be used as a candidate solution that could increase the overall accuracy of trust based recommendations by not providing recommendations where a majority of community members (shaded region in Figure 5-17) have said that they would not allow an action to take place. A total of 93 subjects provided enough personal information to positively identify, anonymously, that they also took part in the original experiment two.

5.3.4.2 Fourteen Day Follow up Survey

In the fourteen day follow up survey ten test subjects from the initial experiment two survey were asked to provide yes and no answers across the same 16 actor and action combinations exactly as per section four of that original survey. It was possible to anonymously correlate the answers that each subject gave in both survey sets over the fourteen day period. This could be used to show the average number of decisions that were changed by a subject over a fourteen day period.

On average, there are approximately 0.6 changes per person out of the possible 16 actor and action combinations. In other words, each subject changed less than 1 answer out of 16 answers after a fourteen day period.

5.3.5 Key Findings

The overall percentage of accurate recommendations, 78.21%, can be categorised into four levels and analysed further. The four levels to be analysed correspond to the four levels of required trust; *very low*, *low*, *high*, and *very high*. In this way it is possible to evaluate the accuracy of calculated trust recommendations with respect to increasing risk (see Figure 5-19). The assumption is made that higher trust requirements equates to higher associated risk.

Required Trust	Number of Recommendations	Accuracy
<i>very low</i>	944	95.55%
<i>low</i>	584	82.36%
<i>high</i>	1124	75.44%
<i>very high</i>	1860	69.78%

Figure 5-18 Accuracy of Recommendations for Likert Scales

Figure 5-18 shows the accuracy of recommendations made at each of the four levels of required trust. For example, at the *very low* level of required trust where 944 recommendations are made the overall accuracy of calculated trust recommendations is 95.55% correct.

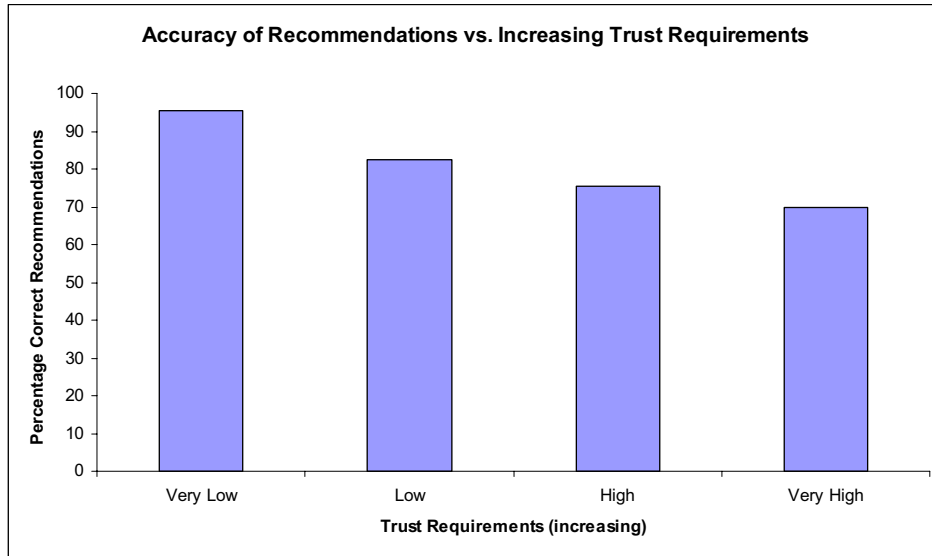


Figure 5-19 Accuracy of Recommendations vs. Increasing Trust Requirements

As illustrated in Figure 5-19 the accuracy of the trust recommendations decreases as the risk associated with the actions increases, which is in line with the original hypothesis for this experiment.

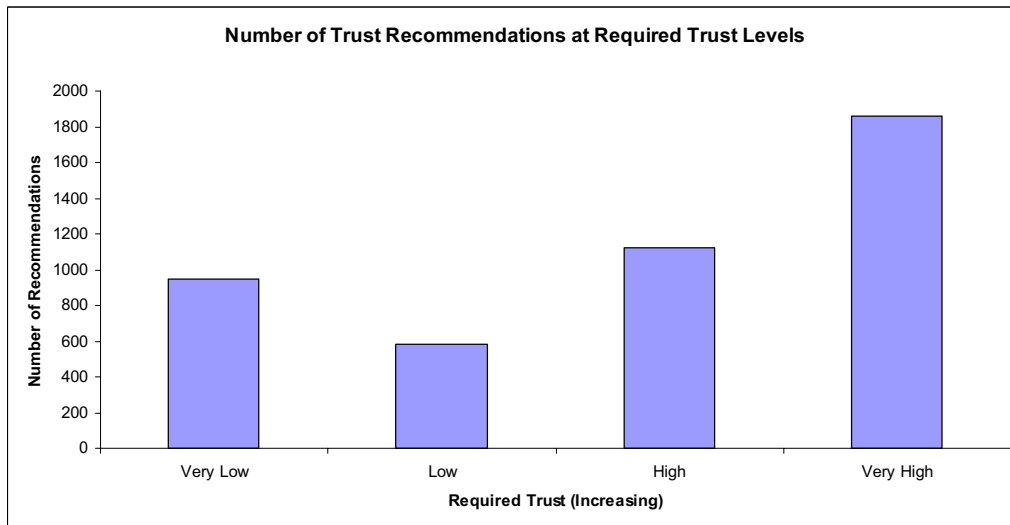


Figure 5-20 Number of Trust Recommendations at Required Trust Levels

Figure 5-20 shows that, independent of accuracy, the number of recommendations made at each required trust level increases as the required trust level increases from *very low* and *low* to *high* and *very high*. The *high* and *very high* required trust levels accounts for 66.1% of all recommendations. The *very high* level accounts for twice as many recommendations as *very low*. In addition, the total amount of recommendations made at the *very high* required trust level accounts for more recommendations than the sum of both *low* and *very low*.

Figure 5-19 shows that at lower levels (*very low* and *low*) of required trust the recommendations are more accurate than at higher levels (*high* and *very high*). However, as per Figure 5-20 more recommendations are made at higher levels of required trust. At lower levels of required trust the recommendations are 90.45% accurate. When the required trust level increases to higher levels the accuracy falls to 71.9%. Therefore, 28.1% of recommendations made at higher levels of trust are incorrect. This 28.1% equates to 838 incorrect recommendations, which accounts for 85% of all incorrect recommendations made across all required trust levels. Improving the accuracy at higher levels of trust is therefore key to providing better overall accuracy.

Note however that the four actions that were presented to the subject are not evenly assigned to each required trust level. Figure 5-20 merely states that the required trust level with the most actions associated with it is *very high*, and that approximately 66% of actions require *high* or *very high* trust.

The analysis has shown that the *very high* required trust level is both the most sought after for recommendations and the most inaccurate. Further analysis within the *very high* required trust level reveals that recommendations made where the actor has *low* or *very low* trust are 86.9% or 94% accurate, respectively. Combining the lower levels of required trust (*very low* and *low*) yields a recommendation set that is 88.85% accurate.

Figure 5-21 illustrates the accuracy results for all 16 actor and action combinations. The grid location where the actor has *very low* trust ('very low trust') and the required trust level is *high* ('high required trust') will be used to explain the presentation format for this accuracy data. At this location in Figure 5-21 it can be seen that 99 recommendations were made, 15 of these recommendations were incorrect, which therefore means that the accuracy of recommendations is 84.8%. This is the presentation format for all other actor and action combinations for the entire grid.

	Very Low Trust Value	Low Trust Value	High Trust Value	Very High Trust Value
Very Low Required Trust	97 91.8% 8	272 93.0% 19	425 96.7% 14	150 99.3% 1
Low Required Trust	55 56.4% 24	174 73.6% 46	269 88.5% 31	86 97.7% 2
High Required Trust	99 84.8% 15	313 60.0% 125	535 77.7% 119	177 90.4% 17
Very High Required Trust	201 94.0% 12	517 86.9% 68	819 54.2% 375	323 66.9% 107

Figure 5-21 Accuracy of Recommendations across Trust and Risk Levels

As per Figure 5-21 it can be seen that recommendations made where the required trust level is *very high* and where the actor has *high* or *very high* trust are 54.2% or 66.9% accurate, respectively. Combining the higher levels of required trust (*very high* and *high*) yields a recommendations set that is 57.8% accurate. As can be seen in Figure 5-21 the *high* required trust level accounts for a significant number of incorrect recommendations; approximately half the number of inaccurate recommendations that were made at the *very high* required trust level where the actor has *high* or *very high* trust.

The test subject community result set shown in Figure 5-17 could be used as an overall community policy that could increase the overall accuracy of recommendations and reduce the levels of inaccuracy in the shaded region in Figure 5-21. The shaded data in Figure 5-17 illustrates the decision of more than 50% of community members. These community members decided to not allow the action associated with the correlating actor to be granted. This information can be used to create a community policy whereby recommendations are not calculated in cases where more than 50% of the community have decided to not allow the action with

respect to the correlating actor to be granted. If this policy is enforced then the overall accuracy of the trust recommendations made with respect to the 93 people who participated in both the original experiment two and the seven day follow up survey improves from 78.21% to 92.2%. Therefore, using an overall community policy increases the overall accuracy of recommendations. However, the associated cost of providing such an increase in accuracy is the dismissal of 6 out of 16 possible actor and action combinations. In addition, these 6 possible actor and action combinations are associated with the higher risk actions such as *bank* and *laptop*, where a large amount of recommendations are made. However, the overall policy still enables the system to provide recommendations that account for a significant proportion of incorrect recommendations, namely recommendations made at (i) *very high* required trust level and actors with *very high* trust, and (ii) *high* required trust level and actors with *high* trust.

The analysis of the overall community policy leads the author to conclude that using an overall community policy is not an effective mechanism to increase overall accuracy and reduce the levels of inaccuracy in the shaded region in Figure 5-21. It is hypothesised that a more useful solution to providing greater overall accuracy lies in ascertaining how to increase the accuracy primarily among actions that have *very high* required trust levels where the actors have *high* or *very high* trust. In addition, increasing the accuracy among actions that have *high* required trust levels where the actors have *low* or *high* trust is also desired. These specific actor and action combinations are shaded in Figure 5-21.

One candidate solution to providing such a mechanism is to provide the subject with additional information that reduces the associated risk and enables the subject to make a more informed decision. The question would move from ‘Would you allow a high trusted person to use your laptop?’ to ‘Would you allow a high trusted person to use your laptop (i) in your office, (ii) under your supervision, and (iii) to reply to an urgent email?’. Therefore, it was decided that the aim of the next experiment would be to investigate whether the provision of additional information will increase the accuracy of trust based recommendations by enabling the subjects to make more informed decisions.

5.4 Experiment Three- Accuracy of Model of Trust with Additional Information

In the introduction chapter of this thesis the fourth goal that was derived to evaluate the research question stated that it was necessary to “Evaluate the accuracy of trust based recommendations calculated using the developed trust management service”. This experiment extends experiment two, which was designed to address this goal. The previous experiment evaluated the accuracy of recommendations made through the multi-faceted model of trust. An analysis of these results illustrated the points at which the accuracy of the recommendations is satisfactorily high and where the accuracy is less than desired. In this experiment additional information is offered to the subject that, it is hypothesised, will result in greater overall accuracy and improved accuracy at the identified weak points in experiment two.

5.4.1 Experiment Goals

The goals of this experiment were to identify (i) what would happen to the accuracy of recommendations if additional information was offered to the test subjects, and (ii) would such additional information be most sought after in the areas where recommendations have been identified as weak in experiment two (see shaded areas in Figure 5-21), and would the accuracy of these recommendations be improved.

5.4.2 Hypotheses

It is hypothesised that offering additional information to the test subject may be required in order for the subject to make a sound decision. The hypothesis of this experiment is that the recommendation accuracy that was achieved in experiment two will alter in some way when the subject is provided with additional information. Secondly, the additional information will be sought most at, and be beneficial to, the identified weak points in experiment two with respect to trust recommendations, where higher levels of risk are present.

5.4.3 Experiment Overview

The web based survey used in experiment three has been derived from web based survey used in experiment two. Section one and two, which correspond to trust model generation and trust annotation, respectively, are re-used verbatim. Section three (rule creation) has been extended to allow the subject to select a set of guarantees that would likely convince them to grant each action to someone. For example, in the instance of the *laptop* a subject could select ‘short term’, ‘under your supervision’, and for ‘official use’ as guarantees that would likely see them lend someone their *laptop*. In experiment two, section four asked the subject to decide whether they would allow, or not allow, an action to be granted for all 16 actor and action combinations. In this experiment and for each of the same 16 combinations the subject could provide answers without any additional information (as per previous experiment) or the subject could request additional information. The additional information consists of two sources. The first source of additional information has been termed ‘Ask the Audience’. If this was selected the subject was presented with the aggregated opinion of the community for that specific actor and action combination. Figure 5-22 illustrates the community’s aggregated opinion for the combination *family member* and *pencil*, where 98.9% of all subjects did allow their *family member* to use their *pencil*. Note that the data for this opinion was gathered in section four of experiment two, where the 282 subject each provided yes or no answers for each of the 16 actor and action combinations.

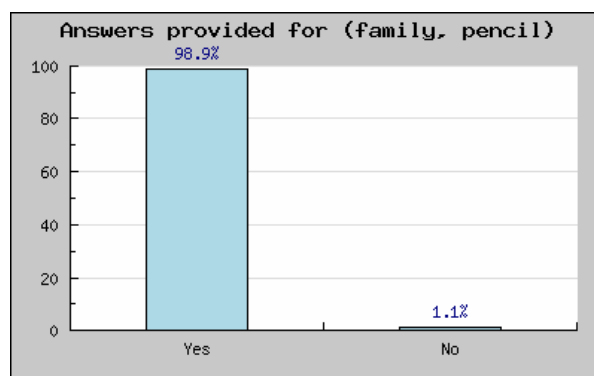


Figure 5-22 'Ask the Audience' Results for *family* and *pencil*

The second source of additional information has been termed ‘Provide Guarantees’. If a subject selected ‘Ask the Audience’ they could either provide an answer after seeing the community opinion or the subject could choose to select ‘Provide Guarantees’.

The 'Provide Guarantees' option reflected the subjects' answers from section three, where they were asked to choose guarantees that would likely convince them to grant each action to someone. Figure 5-23 presents an example of the guarantees provided. It presents to the subject the options that they selected in section three (rule creation), and states that the *work colleague* has guaranteed that borrowing of the *mobile phone* will be for an 'inexpensive call', 'under your supervision', and for a 'short time'. With this additional information the subject may be able to make a more informed decision about this action, which is generally regarded as *high* or *very high* risk.

Questionnaire

Would you allow a **WORK COLLEAGUE** to use your **MOBILE PHONE**?

GUARANTEES PROVIDED

Assume that the **WORK COLLEAGUE** has provided you with the following guarantees regarding use your **MOBILE PHONE**.

- The **WORK COLLEAGUE** will use your **MOBILE PHONE** under your supervision.
- The **WORK COLLEAGUE** will use your **MOBILE PHONE** for an inexpensive call.
- The **WORK COLLEAGUE** will use your **MOBILE PHONE** for a short time.

Please select either 'yes' or 'no' and press the 'Submit Answer' button.

Yes: No:

Submit Answer

Figure 5-23 'Provide Guarantees' Results for *work colleague* and *mobile phone*

By offering the subjects three options when answering each of the 16 actor and action combinations it was possible to evaluate the hypothesis that additional information would increase the accuracy of recommendations, and that the increases will be in areas where accuracy was less than desired in experiment two.

The subjects were offered the opportunity to take part in a competition to win two tickets to Republic of Ireland vs. Chile in Dublin, Ireland as well as €100 gift voucher in order to entice their participation. Advertisement for participation was conducted via email, to a wide range of faculties within Trinity College Dublin, mainly Computer Science at undergraduate, postgraduate, and staff levels. In total 220 fully completed questionnaires were received over a two week period. It can be ascertained from email addresses that 65.0% (143) of these 220 subjects were from Trinity College Dublin. A total of 25.0% (55) of the 220 subjects were from the Computer Science department, Trinity College Dublin. Therefore, 10% (22) subjects did not provide a Trinity College Dublin, or Computer Science, email address.

The experiment three survey can be viewed in APPENDIX II – Research Experiments, Experiment Three; Accuracy Survey with Additional Information. However, in section four it is only the questions regarding *pencil* that are included as *laptop*, *mobile phone*, and *bank pin* all take the same approach.

5.4.4 Results

Figure 5-24 presents the data received from all 220 subjects that took part in experiment three that provided a fully completed questionnaire. As per experiment two the data has been sorted according to the four possible levels of trust that a subject might require ('Required Trust' column) for any given action; *very low*, *low*, *high*, and *very high*. At each of these four trust requirement levels the data is broken down further into the four possible calculated trust levels ('Calculated Trust Value' column). In this way the required trust for an action is correlated to the calculated trust for the associated actor for all 16 actor and action combinations. The number of recommendations made for each of the 16 actor and action combinations are also presented ('Number of Recommendations' column).

The numbers of correct and incorrect recommendations made with respect to answers provided by the subject are presented over four columns for all actor and action 16 combinations. Finally, the percentage of correct recommendation is provided for all 16 actor and action combinations (Percentage Correct column).

An analysis of Figure 5-24 shows that the majority of incorrect recommendations, 82.58% (79.9% in experiment two), occurred in conditions where the subject did not allow the action to take place occur where the required trust is either *high* or *very high*. Here the subject had provided an answer that did not allow the associated action to take place yet the calculated recommendation did allow the action to take place. In both cases the calculated trust was equal to, or greater than, the required trust. The total percentage of incorrect answers, in conditions where the subject did not allow the action to take place, accounts for 6.74% (8.07% in experiment two) of overall recommendations.

Required Trust	Calculated Trust Value	Number of Recommendations	Correct (Allowed)	Correct (Not Allowed)	Incorrect (Allowed)	Incorrect (Not Allowed)	Percentage Correct
Overall Very High	Very High	265	197	6	2	60	76.6%
	High	543	38	301	197	7	62.4%
	Low	349	9	294	44	2	86.8%
	Very Low	155	1	149	5	0	96.8%
Overall High	Very High	178	168	1	1	8	94.9%
	High	391	330	4	3	54	85.4%
	Low	245	19	132	88	6	61.6%
	Very Low	114	5	102	7	0	93.9%
Overall Low	Very High	123	122	0	0	1	99.2%
	High	287	259	2	1	25	90.94%
	Low	186	154	1	2	29	83.3%
	Very Low	60	13	22	25	0	58.3%
Overall Very Low	Very High	126	121	1	0	4	96.8%
	High	271	256	1	0	14	94.8%
	Low	156	142	1	0	13	91.7%
	Very Low	71	55	1	1	14	78.9%
Totals	Very High	692	608	8	3	73	
	High	1492	883	308	201	100	
	Low	936	324	428	134	50	
	Very Low	400	74	274	38	14	
		3520	1889	1018	376	237	
Percentage			53.66%	28.92%	10.68%	6.74%	
Total Percentage			82.58%		17.42%		

Figure 5-24 Total Recommendation Accuracy

From Figure 5-24 the majority of incorrect recommendations, 89.0% (91.7% in experiment two), in conditions where the subject did allow the action to take place occur where calculated trust is *low* or *high*. Furthermore, the majority of incorrect recommendations, 53.5% (60.6% in experiment two), in conditions where the subject did allow the action to take place occur where calculated trust is *high*. In these cases the subject had provided an answer that did allow the associated action to take place yet the calculated recommendation did not allow the action to take place. The total percentage of incorrect answers, where the subject did allow the action to take place, accounts for 10.68% (13.72% in experiment two) of overall recommendations.

Across all 16 possible actor and action combinations, and for all 220 subjects (3520 total recommendations), the total number of incorrect recommendations provided to subjects across all required trust levels was 613, or 17.42% (21.79% in experiment two). Therefore, the total percentage of correct recommendations, out of 3520 recommendations, across all required trust levels was 2907, or 82.58% (78.21% in experiment two).

5.4.5 Key Findings

Required Trust	Number of Recommendations	Accuracy
<i>very low</i>	400	87.0%
<i>low</i>	936	80.0%
<i>high</i>	1492	79.8%
<i>very high</i>	692	89.0%

Figure 5-25 Accuracy of Recommendations for Likert Scales

Figure 5-25 shows the accuracy of recommendations made at each of the four levels of required trust. For example, at the *very low* level of required trust where 400 recommendations are made the overall accuracy of calculated trust recommendations is 87.0% correct.

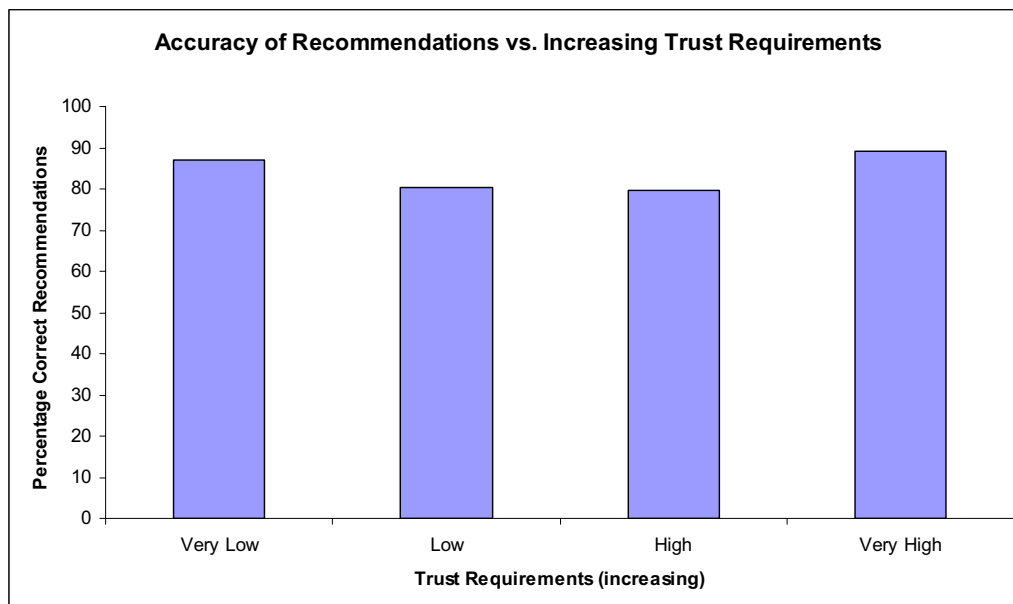


Figure 5-26 Accuracy of Recommendations vs. Increasing Trust Requirements

As illustrated in Figure 5-26 the accuracy of the trust recommendations no longer continually decreases as risk increases as in experiment two (see Figure 5-19).

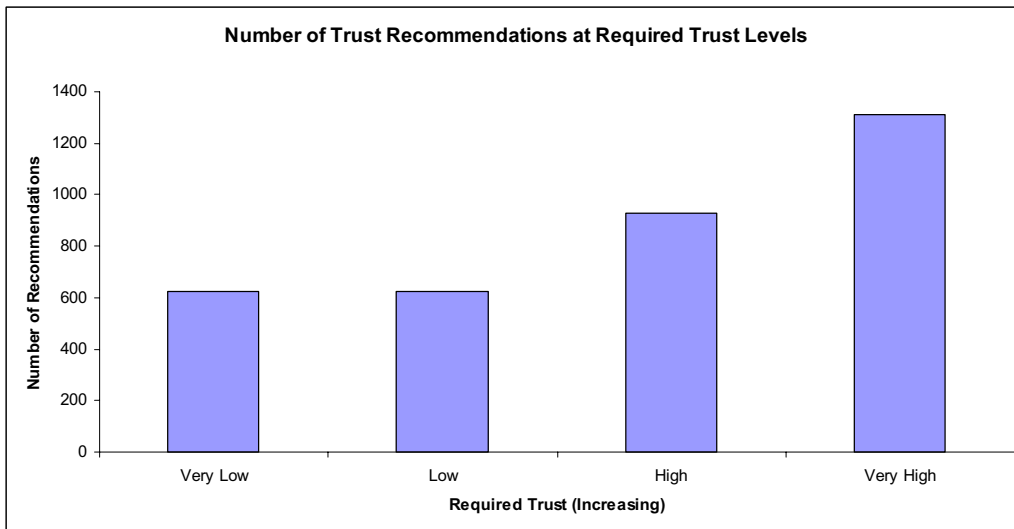


Figure 5-27 Number of Trust Recommendations at Required Trust Levels

Figure 5-27 shows that, independent of accuracy, the number of recommendations made at each required trust level is almost equal at *very low* and *low* required trust and then tends to increase as the required trust level increases from *low* to *high* and *very high*. The *high* and *very high* required trust levels account for 63.6% of all recommendations, which is quite similar to the experiment two's analysis; 66.1%. The *very high* level accounts for twice as many recommendations as *very low*, which was also found in the previous experiment. In addition, like in the previous experiment the total amount of recommendations made at the *very high* required trust level accounts for more recommendations than the addition of both *low* and *very low*.

Figure 5-28 illustrates the percentages of correct recommendations made over increasing trust requirements for both experiment two and experiment three. As described earlier the accuracy of recommendations made in experiment two tends to decrease as risk increases. However, in experiment three this trend is not found. In experiment three the accuracy of recommendations made as risk increases has stabilised in comparison to experiment two. It is suggested that the availability of the option to select additional information that is directly responsible for the increased accuracy of recommendations between the two experiments. The supporting evidence to this claim is provided in the forthcoming analysis.

It is important to note that the number of subjects that participated in both the original experiment two and experiment three is 65. Therefore, out of the 220 subjects that took part in experiment three, 65 also took part in experiment two.

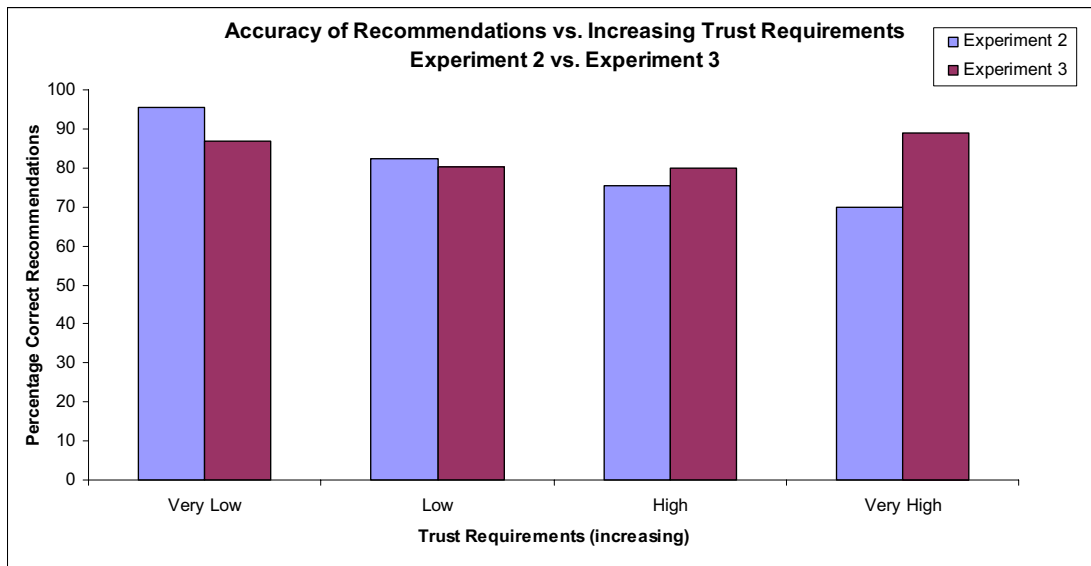


Figure 5-28 Accuracy of Recommendations vs. Increasing Trust Requirements (Experiment two vs. Experiment three)

In Figure 5-28 the most noticeable increase in the accuracy of recommendations is found where *very high* trust is required. At this point the accuracy of recommendations increased significantly from 69% in experiment two to 89% in experiment three. Furthermore, in both experiment two and three the *very high* required trust level is where the single highest number of recommendations are made. Therefore, this approximate 20% increase in accuracy occurs where the single highest number recommendations are made. Where *high* trust is required the accuracy of recommendations increases by approximately 4%. This is where the second single highest number of recommendations is made.

Where either *very low* or *low* trust is required there is a decrease in the accuracy of recommendations across experiment two and experiment three. However, fewer recommendations are made at these levels (see Figure 5-20 and Figure 5-27) in comparison with the *high* and *very high* levels and so the impact is not as significant as the gains which are found at the *high* and *very high* levels.

In general, the accuracy of recommendation made at the *very low* and *low* risk levels has experience a minor decrease, while comparatively the accuracy of recommendation made at the *high* and *very high* risk levels experience much larger increases.

The hypothesis of this experiment was that the accuracy of recommendations would alter in some way when the subject was provided with additional information (experiment three) than when the subject is not provided such additional information (experiment two). This is the primary difference between experiment two and experiment three. The differences between Figure 5-16 and Figure 5-24 illustrates an increase in overall recommendation accuracy for experiment three.

In addition, the hypothesis stated that the benefits of this additional information would be noticed more at identified weak points in experiment two, where higher levels of risk were present. Figure 5-28 illustrates that there are significant gains in accuracy at the *high* and *very high* levels of required trust in experiment three. However, Figure 5-28 also illustrates that there are decreases in the accuracy of recommendations at the *low* and *very low* levels.

It was also hypothesised that the shaded areas in Figure 5-21, the identified weak points for recommendations, would be the combinations of actor and actions that would receive the most sought after for additional information in experiment three. This additional information was not available in experiment two. In addition, these identified weak points would benefit from an increase in the accuracy of recommendations.

	Very Low Trust Value	Low Trust Value	High Trust Value	Very High Trust Value
Very Low Trust Requirement	9	15	13	7
Low Trust Requirement	14	29	22	5
High Trust Requirement	11	41	46	24
Very High Trust Requirement	16	30	89	46

**Figure 5-29 Selection Points for Additional Information,
Across Trust and Risk Levels**

Figure 5-29 illustrates where the additional information is being requested. Where *very low* trust is required 44 requests for additional information are made. This increases to 70 requests where *low* trust is required. When *high* trust is required 122 requests are made. Finally, when *very high* trust is required 181 requests for additional information are made. Therefore, as more trust is required, or as the risk increases, more requests for additional information are being sought. The shaded areas in Figure 5-29 illustrate the four actor and action combinations where the most requests are made. These four regions directly correlate to the four shaded regions in Figure 5-21, experiment two, which were identified as weak points and where the accuracy of recommendations was less than desired. This is where it is hypothesised that additional information would be most sought and would be of benefit. Figure 5-29 confirms the hypothesis that the highest numbers of requests would be made in these shaded regions.

When additional information is sought the accuracy of the recommendations increases with respect to the actor and action combinations in the shaded areas in Figure 5-21, Figure 5-29, and Figure 5-31. When the subject seeks only the ‘Ask the Audience’ opinion the accuracy of recommendations can rise to 89.1%. When ‘Provide Guarantees’ is also selected the accuracy of recommendation can rise to 82.9%.

	Very Low Trust Value	Low Trust Value	High Trust Value	Very High Trust Value
Very Low Trust Requirement	71 78.9% 15	156 91.6% 13	271 94.8% 14	126 96.2% 4
Low Trust Requirement	60 58.3% 25	186 83.3% 31	287 88.4% 57	123 99.2% 1
High Trust Requirement	114 93.9% 7	245 61.6% 94	391 85.4% 57	178 94.9% 9
Very High Trust Requirement	155 96.7% 5	349 86.8% 46	543 62.4% 204	265 76.6% 62

Figure 5-30 Accuracy of Recommendations across Trust and Risk Levels

Figure 5-30 presents the accuracy of recommendations made across all 16 actor and action combinations in experiment three where addition information can be sought. The shaded areas were the actor and action combinations that provided recommendations whose accuracy was less than desired in experiment two.

	Very Low Trust Value	Low Trust Value	High Trust Value	Very High Trust Value
Very Low Trust Requirement	-12.9%	-1.4%	-1.9%	-3.1%
Low Trust Requirement	+1.9%	+9.7%	-0.1%	+1.5%
High Trust Requirement	+9.1%	+1.6%	+7.7%	+4.5%
Very High Trust Requirement	+2.7%	-0.1%	+8.0%	+9.7%

Figure 5-31 Gains & Losses across Trust and Risk Levels

Figure 5-31 illustrates the gains and losses made between experiment two and experiment three for each the 16 actor and action combinations. Overall, the accuracy of recommendations increases from 78.21% in experiment two to 82.58% in experiment three. The provision of additional information in experiment three has clearly increased the overall accuracy of recommendations. However, Figure 5-31 shows that this overall increase is not evident at every actor and action combination or every required trust level,

In general, accuracy increases at *very high* and *high* levels of required trust as shown in Figure 5-31, yet there is one actor and action combination at these high levels of required trust where accuracy reduces. Increases in accuracy are mainly found at the *low* level of required trust, yet there is one actor and action combination that decreases in accuracy. At *very low* levels of required trust there is a decrease in the accuracy of recommendations. These results indicate that the overall increase in recommendation accuracy has benefits for some actor and action combinations, but disadvantages for other actor and action combinations.

At the higher levels (*high* and *very high*) of required trust there are 354 fewer incorrect recommendations. At the lower levels (*low* and *very low*) of required trust there is an additional 13 incorrect recommendations. These results indicate that between experiment two and three there are comparatively more increases in the accuracy than there are decreases in the accuracy of recommendations.

All shaded areas in Figure 5-31 show an increase in recommendation accuracy between experiment two and experiment three. These shaded regions were identified as weak points in experiment two in regards recommendation accuracy. The provision of additional information has benefited these actor and action combinations.

It is speculated by the author that additional information is not required as much at the *very low* level of required trust in comparison to the higher levels of required trust. As such, the accuracy at the *very low* level of required trust may be decreasing due to the additional information clouding the subject's judgement with regards a decision that is normally obvious and straight-forward due to its very low risk.

It can be concluded from Figure 5-31 that, in this experiment, providing additional information should not be offered in situations where *very low* trust is required as the data indicates that this will result in a decrease in accuracy. However, the data also indicates that offering additional information at any other level of required trust will, in this experiment, more than likely result in an increase in accuracy. This is similar to how eBay members make decisions. In regards an item for \$10 a buyer may be satisfied with the seller having a low number of sales and a relatively positive eBay feedback rating. However, a buyer who wishes to buy an item for \$1000, such as a motorbike, would like additional information that could include owner documentation and a limited warranty.

5.5 Experiment Four- Abstract and Concrete Concepts

It was originally hypothesised in experiment one that certain trust concepts were well defined and had a definite scope; these are termed *concrete* concepts in the meta-model. In addition, it was also hypothesised in experiment one that other trust concepts were not as well defined and more open to interpretation and lacked scope; these are termed *abstract* concepts. The first experiment provided evidence that supports this claim (see Section 5.2.5.1). Experiment four examines the set of trust concepts to establish whether there is a link between the ranking of concepts, over a broad population, and the clarity of these concepts. This experiment was carried out after experiment three as by that stage in this PhD research there was sufficient data to include a comparative analysis between experiments (see Section 5.6).

5.5.1 Experiment Goals

The goals were to identify (i) how clear the concepts of trust are to the subjects that participated in experiment three, and (ii) which concepts have the most and least clarity as perceived by the subjects that participated in experiment three.

5.5.2 Hypotheses

In the fourth experiment all eight trust concepts were examined in terms of their clarity, as perceived by a set of subjects who participated in experiment three. The hypothesis for experiment four is that the *abstract* concepts, most notably *belief* and *faith*, will starkly contrast in terms of clarity when compared with the *concrete* concepts such as *reliability*, *honesty*, *competency*, or *reputation*. Furthermore, the hypothesis states that it is the *concrete* concepts that will have the most clarity and *abstract* concepts will be most unclear.

5.5.3 Experiment Overview

In this experiment a proportionally significant number of subjects, at least 30, were sought, at random (40 actually participated) from experiment three to take a short follow up survey, which took place seven days after experiment three. As a reminder the subjects were firstly shown the instructions from experiment three, verbatim. They were then presented with eight screenshots and asked to answer two short questions. Each of the eight screenshots corresponded to each of the eight stages that comprised

section one (Trust Model) of experiment three. The first question asked “How clear was your understanding of *concept x* when you were asked if it influenced other concepts?”, where *concept x* could be *reputation* for example. The subject could answer *very unclear*, *unclear*, *clear*, or *very clear*. This was asked in order to build a view of the clarity subjects’ had in each trust concept, which can help answer the hypotheses. The second question asked “Briefly describe what you took *concept x* to mean.” and the subject could input an answer as they saw fit. The answers provided insight into how subjects describe trust concepts. The full experiment survey can be viewed in APPENDIX II – Research Experiments, Experiment Four; Clarity Survey.

5.5.4 Results

Figure 5-32 shows the results of experiment four for all eight trust concepts for all 40 subjects. Each concept is accompanied by the aggregate number of choices made by the subjects across all four options; *very unclear*, *unclear*, *clear*, or *very clear*. An ‘overall’ clarity value is assigned to each trust concept. This is calculated by assigning a *very unclear* vote a score of 1, an *unclear* vote a score of 2, a *clear* vote a score of 3, and *very clear* vote a score of 4. For each concept, the number of votes received at each level of clarity is multiplied by the score associated with that clarity. These are summed to yield the overall clarity value. The overall clarity value for *belief* is calculated as follows; $(6 \times 1) + (17 \times 2) + (16 \times 3) + (1 \times 4) = 92$.

Concept	Very Unclear	Unclear	Clear	Very Clear	Overall Clarity Value	Overall Clarity Rank
Belief	6	17	16	1	92	8
Competency	0	3	27	10	127	3
Confidence	0	5	26	9	124	4/5
Credibility	0	8	28	4	132	1
Faith	11	8	16	5	95	7
Honesty	0	5	22	13	128	2
Reliability	1	4	25	10	124	4/5
Reputation	2	7	25	6	115	6

Figure 5-32 Clarity of Trust Concepts

The data from Figure 5-32 states that more than 75% of subjects have found the *concrete* concepts *competency*, *confidence*, *credibility*, *honesty*, *reliability*, and *reputation* to be either *clear* or *very clear* in terms of clarity. However, approximately 50% of the subjects found that the hypothesised *abstract* concepts *faith* and *belief* to be either *unclear* or *very unclear* in terms of clarity.

When asked to describe what the subject took *belief* to mean the predominant answers within the *very unclear* and *unclear* categories described *belief* as ability in one-self, as a synonym of *faith*, and with religious connotations. Within the *very unclear* and *unclear* categories *faith* was generally described as a synonym of *belief*, as being quite vague, and had strong religious intonations.

The trust concepts that received the most *clear* and *very clear* results, such as *honesty* and *competency* had very strong and clear descriptions. The predominant answers within the *very clear* and *clear* categories for *honesty* were that *honesty* was as measure of truth; how truthful someone is. The keyword used to describe *competency* within the *very clear* and *clear* categories was ability. The descriptions for *reliability* within the *very clear* and *clear* categories include dependability, assurance, and a common thread of not failing. Descriptions for *reputation* within the *very clear* and *clear* categories include past performance and the opinion of other people.

5.5.5 Key Findings

The trust concepts that were hypothesised as *abstract* have been rated the least positively in terms of clarity. In turn the trust concepts that were hypothesised as *concrete* have been rated relatively positively in terms of clarity.

Figure 5-32 shows that *belief* and *faith* are the trust concepts that have the least clarity as perceived by the subjects that participated in experiment three and the follow up survey seven days later. In addition, *belief* and *faith* have the lowest overall clarity rank of all trust concepts. The descriptions for these concepts, in terms of the *very unclear* and *unclear* categories, further support the hypothesis that *belief* and *faith* should be considered *abstract*. The results show that many subjects described these concepts as synonyms of each other and they both had broad religious meanings associated with them.

In comparison, top ranking concepts, in terms of the *very clear* and *clear* categories, such as *honesty* and *competency* have descriptions that are well scoped and include keywords such as truth and ability, respectively. The same can be said for *reliability* and *reputation*, which subjects described in very narrow and definite terms; assurance, dependability for *reliability*, and past performance and received opinion.

This analysis may suggest a correlation between the clarity of a concept and its overall ranking. The trust concepts *belief* and *faith* have the least clarity and have consistently ranked low in previous experiments. However, *reputation* and *reliability* have consistently ranked high in previous experiments, yet they are not the trust concepts with the highest overall clarity rank. Therefore, it is possible to conclude that concepts do not rank highly because of their high clarity.

Experiment four has shown that the results for concepts that were hypothesised as *abstract* starkly contrast with *concrete* concepts in that *abstract* concepts have the least clarity and description scope amongst subjects.

5.6 Personalised Model Generation; Experiment One in Comparison with Experiment Two and Three

In the introduction chapter of this thesis the fourth goal that was derived to evaluate the research question stated that it was necessary to “Evaluate the ability of the generation mechanism to produce personalised models of trust that accurately reflect users’ ideas of trust”. This experiment has been designed to address this goal.

The trust concepts *belief* and *faith* have consistently been rated poorly in experiment one in terms of Likert scores, rank and influence, and again in experiment four in terms of clarity. The personalised models of trust that were generated in experiment two and three, through the HITS algorithm, contains more evidence that *belief* and *faith* are lowly ranked trust concepts, and thus can be considered *abstract*.

Rank	Concept
Top Three	<i>reputation</i>
Top Three	<i>reliability</i>
Top Three	<i>honesty</i>
Bottom Two	<i>belief</i>
Bottom Two	<i>faith</i>

Figure 5-33 Concept Rankings via Direct Questioning

Figure 5-33 presents the trust concepts that received top three and bottom two rankings in experiment one. Note that these ranking were derived by direct questioning of the subjects in experiment one. In experiment one the trust concepts *belief* and *faith* are ranked in the bottom two.

Rank	Concept
Top Three	<i>reputation</i>
Top Three	<i>reliability</i>
Top Three	<i>credibility</i>
Bottom Two	<i>belief</i>
Bottom Two	<i>faith</i>

Figure 5-34 Concept Rankings via Personalised Model of Trust

Figure 5-34 presents the top three and bottom two ranked concepts in experiment two. These rankings were derived from the aggregation of rankings found in the HITS generated personalised models of trust for subjects that took part in experiment two.

An analysis of the personalised models of trust generated in experiment three provides similar results as experiment two as seen in Figure 5-34. The data from experiment two is used for analysis as it is based on a greater number of subjects. In experiment two and three the trust concepts *belief* and *faith* again rank in the bottom two.

There are notable similarities in the ranking of concepts across experiments one, two, three, and four; the trust concepts *belief* and *faith* are ranked in the bottom two. The data in Figure 5-32 states that *belief* and *faith* are the trust concepts that are most lacking in clarity. It has been previously stated that this may suggest a correlation between the clarity of a concept and its overall ranking, yet this correlation was shown to be incorrect when considering *reputation* and *reliability*. Using (i) experiment one's Likert scores, rankings, and influence data, (ii) experiments two and three HITS generated models of trust, and (iii) experiment four's clarity data it could be possible to conclude that *belief* and *faith* are poor choices to use as concepts that define the term trust. However, further analysis of the data collected in experiment two can provide a more accurate statement regarding *belief* and *faith*. Figure 5-35 present the aggregated rank for *belief* and Figure 5-36 present the aggregated rank for *faith* as derived from personalised models of trust from all subjects in experiment two.

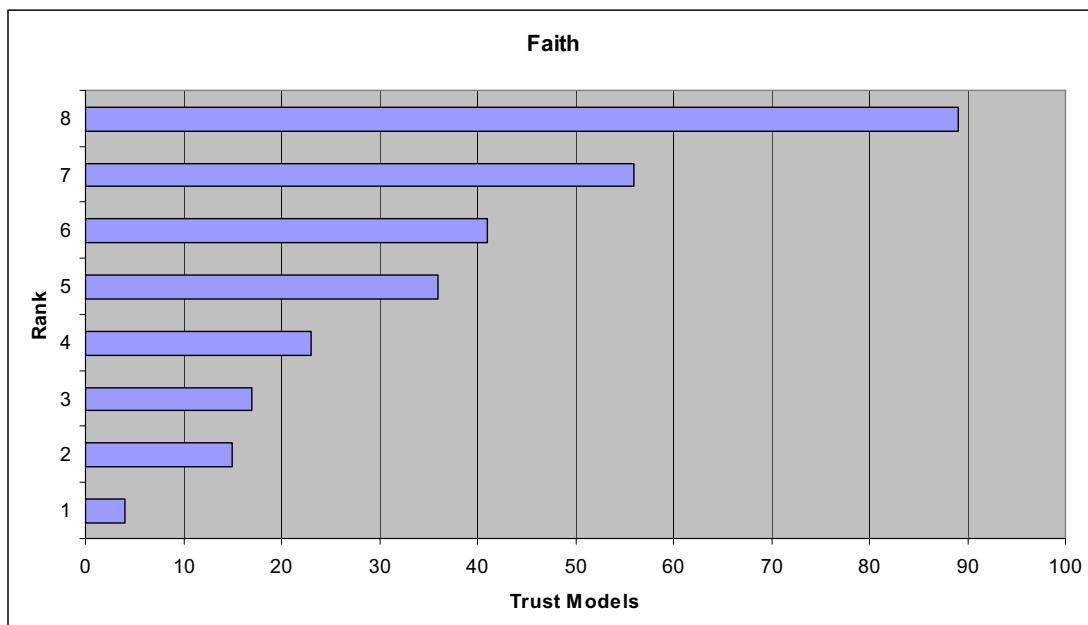


Figure 5-35 Aggregate Rank for *faith* via HITS algorithm

Figure 5-35 shows the aggregate ranking results for *faith*, which has the most number eight (last place) rankings. It is important to note that Figure 5-35 shows that approximately 12.8% of all subjects rank *faith* number one, two, or three.

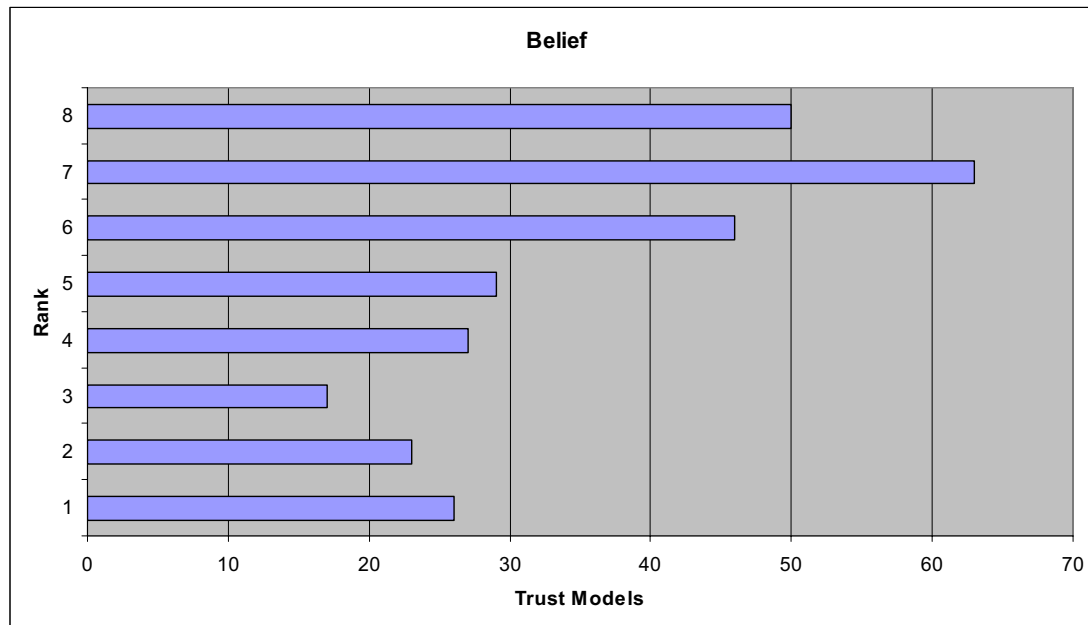


Figure 5-36 Aggregate Rank for *belief* via HITS algorithm

The data in Figure 5-36 illustrates the aggregate ranking results for *belief*, which has the most number of seven (second last place) rankings. However, it is important to note that *belief* also has a significant percentage, 23.4%, of number one, two, or three rankings.

Therefore, the choice of *faith* and *belief* as concepts that define the term trust are not entirely poor choices as significant numbers of subjects have *belief* and/or *faith* in their top three trust concepts. This also further supports the argument that trust should be multi-faceted and personalised. The inclusion of *faith* and *belief* as trust concepts empowers the multi-faceted model to capture the wide and diverse range of ideas found across a population, while also capturing the subjectivity of trust through personalisation.

Figure 5-33 also states that *reputation* and *reliability* rank in the top three when the subject was directly asked to rank the concepts in order from one to three in experiment one.

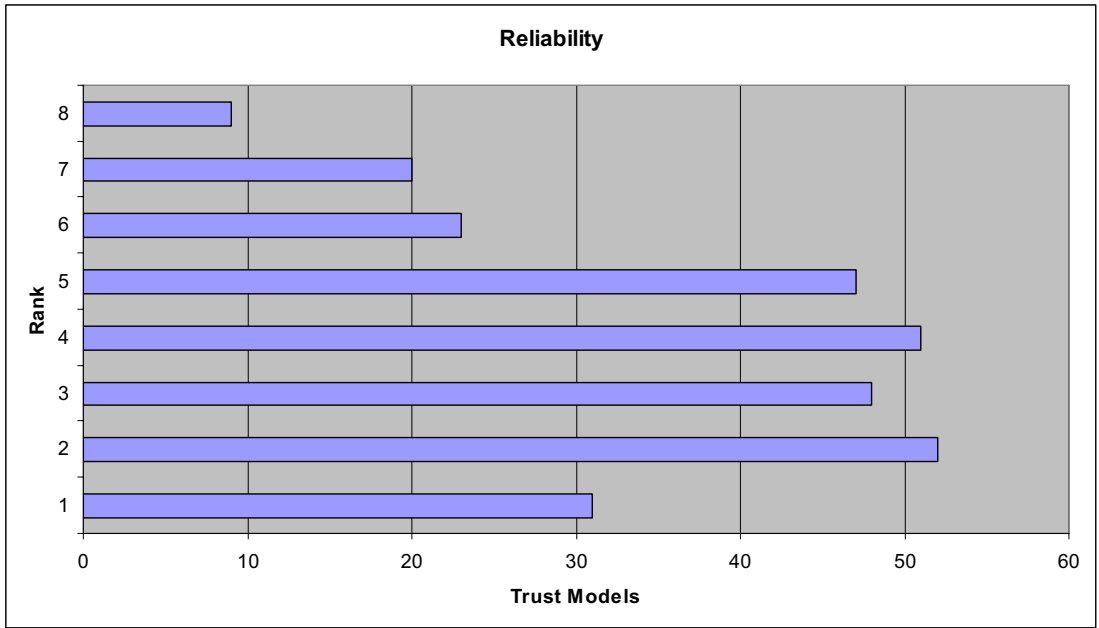


Figure 5-37 Aggregate Rank for *reliability* via HITS algorithm

Figure 5-37 shows the results for *reliability*, which appears in the top three ranked subject in experiment two.

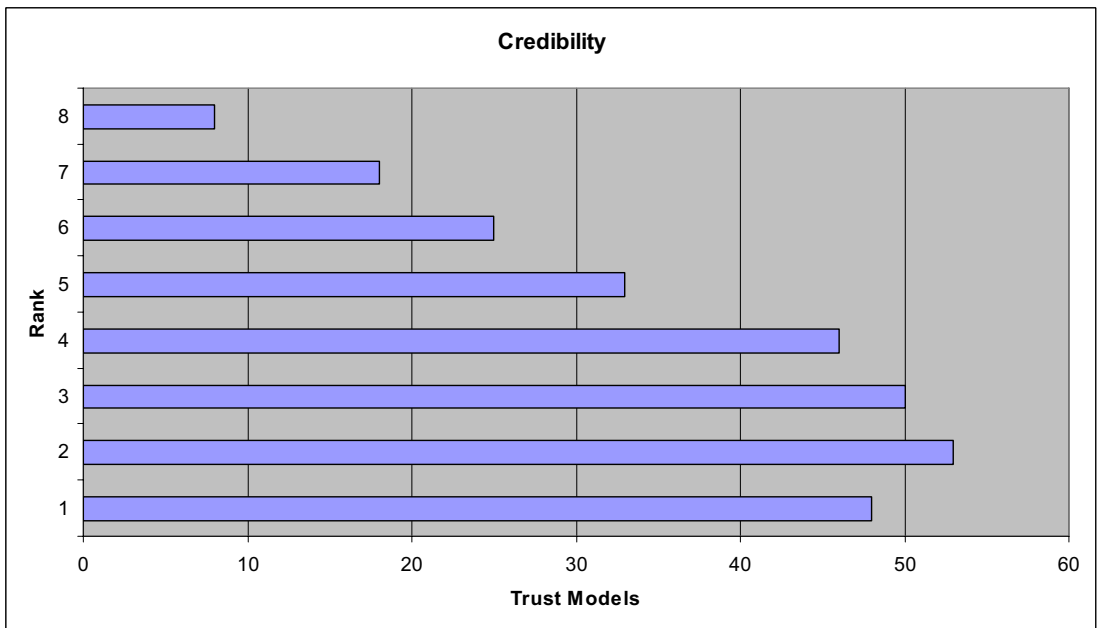


Figure 5-38 Aggregate Rank for *credibility* via HITS algorithm

Figure 5-38 shows the results for *credibility*, which also appears in the top three ranked subject in experiment two. It is *credibility* that is ranked second, instead of *reliability*, as it marginally has more number two ranking and significantly more number one rankings in relation to *reliability*.

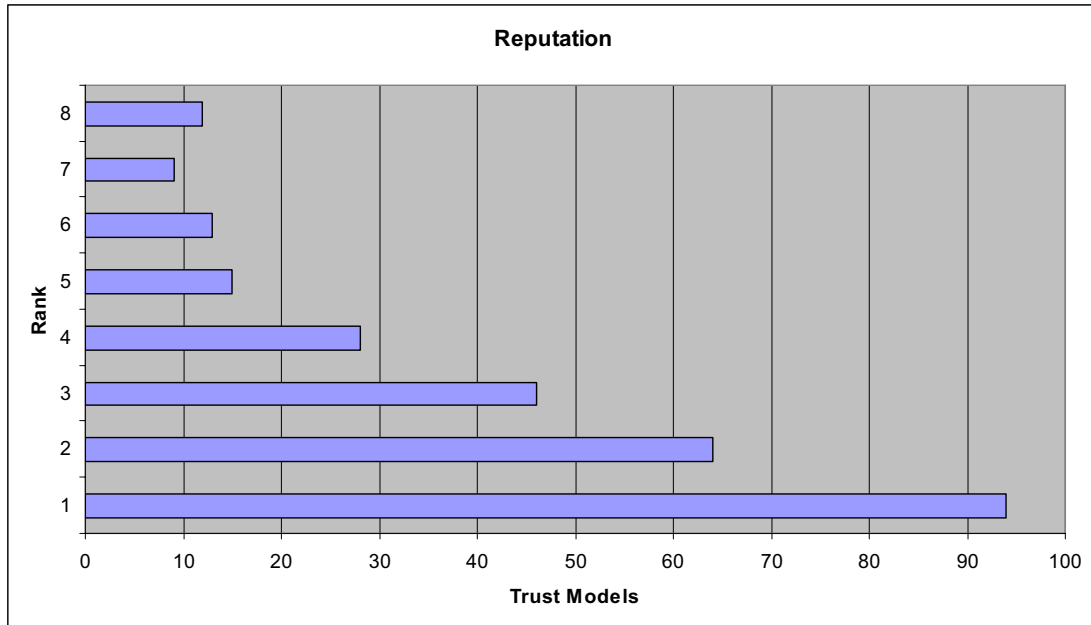


Figure 5-39 Aggregate Rank for *reputation* via HITS algorithm

Figure 5-39 shows the results for *reputation*, which has the most number one rankings in experiment two.

Figure 5-33 illustrates the partial rankings of trust concepts when 279 subjects were directly asked to rank the concepts in order one to three, whereas Figure 5-34 illustrates the partial rankings of the trust concepts when 282 subjects provided information from which the HITS algorithm could generate personalised models of trust. The important correlation between Figure 5-33 and Figure 5-34 is that the aggregated ranking of both methods produce similar overall rankings.

It is possible to conclude from this correlation that generating personalised models of trust, using the HITS algorithm, produces a set of aggregated rankings that generally reflects the aggregated rankings of concepts when the subject was directly asked for rankings. Therefore, the mechanism for generating the personalised models of trust is considered sound as models generated using this mechanism reflect the subjects' models of trust when they were directly asked. This means that it is possible to provide a mechanism for generating a personalised model of trust that yields a comparative ranking of trust concepts. In addition, the benefits of the weighting could see each concept's weight being used as part of a trust calculation algorithm. For example, instead of using the top three ranked concepts the number of concepts could be based on these weightings, which could see the use of one or all eight concepts.

5.7 Summary

This chapter has discussed the evaluation of the multi-faceted model of trust that is personalisable and specialisable. Section 5.2 presented experiment one, which discussed and evaluated the necessity for a multi-faceted model of trust. This evaluation has shown that a single-faceted model of trust can be useful to certain individuals within a wide population. However, the evaluation has also shown that the subjectivity found within trust across a wide population cannot be satisfactorily captured by a single-faceted approach. The evaluation has shown that a multi-faceted model of trust can capture an individual's subjective idea of trust while at the same time capturing the wide and varied range of subjectivity found across a diverse population.

Sections 5.3 and 5.4 presented experiments two and three, which discussed and evaluated the accuracy of recommendations made using the multi-faceted model of trust that was personalised towards each subject. These evaluations showed that (i) there are circumstances where trust recommendations are highly accurate, (ii) there are circumstances where the accuracy of trust recommendations are less than desired, and (iii) by providing the subject with additional information the overall accuracy of recommendations can increase significantly.

Section 5.5 presented experiment four, which investigated the clarity and meaning associated with each of the eight trust concepts. This evaluation showed that concepts categorised as *abstract* had the least clarity and were perceived by many as having religious connotations and as synonyms for each other.

Section 5.6 presented a comparison of personalised models of trust. This evaluation illustrated that the HITS algorithm produces a set of aggregated rankings that generally reflects the aggregated rankings of concepts when the subject was directly asked for rankings. Therefore, it was concluded that the HITS algorithm can be used as a mechanism for generating a personalised model of trust that yields a comparative ranking of trust concepts.

The research question posed in this thesis is *whether a multi-faceted model of trust that is personalisable and specialisable is both necessary and accurate to the user in providing a dynamic and flexible trust based decision support mechanism within Internet environments*. This evaluation chapter has addressed the necessity and accuracy elements of this research question.

The next chapter illustrates the provision of a dynamic and flexible trust based decision support mechanism within Internet environments.

6 TRIALS

6.1 Introduction

This chapter presents the three trials that were undertaken to illustrate the successful provision of a dynamic trust management service that utilises the multi-faceted model of trust that is personalisable and specialisable.

Section 6.2 presents a discussion on background information pertinent to the three trials presented in this chapter. This includes an overview of the systems and applications that were used in the trials, including Community Based Policy Management (CBPM) system, enhanced Instant Messaging (IM) application, and PUDECAS ubiquitous computing simulator [O'Neill et al, 2005].

The three trials are briefly described and compared in Section 6.3. This comparison illustrates the use of different architectures, personalisation algorithms, trust annotation mechanisms, policy specification, and trust calculation algorithms that were used across all three trials.

The first trial is described in Section 6.3. The first trial used a prototype of *myTrust* that is called *deepTrust* [Quinn et al, 2005], which was designed, developed, and trialled in conjunction with Ericsson Research Group, Ireland in 2004. *deepTrust* served as a proof of concept for the subsequent development of the *myTrust* trust management system.

Section 6.5 presents the second trial, which used the implemented *myTrust* trust management service. Trial two combines *myTrust* and CBPM in order to provide dynamic and flexible access control to objects within the PUDECAS environment.

The combined *myTrust* and CBPM system is used further in trial three, which is presented in Section 6.6. In trial three the combination of *myTrust* and CBPM provide dynamic and flexible access control over location information produced in the in the PUDECAS ubiquitous computing simulator for use in an enhanced IM application.

6.2 Background Information for Trials One, Two, and Three

Each trial used a system or application that was not part of the implemented *myTrust* trust management service. In addition, each successive trial made use of a system or application that previous trials did not use. This section describes these systems, namely; CBPM, PUDECAS, and enhanced IM.

6.2.1 Community Based Policy Management (CBPM)

In [Feeney et al., 2004] the CBPM system for the management of policies in organisations with a decentralised and evolving structure is presented. This model builds upon previous work in the field of policy based management, but simplifies specification of the organisation by using a single grouping construct, the community, to model the organisation. It introduces a concept of delegation whereby authority and rights are delegated to dynamically build a model of the organisation. Initially, the most basic structure can be created and the detailed divisions of responsibility and the organisational groupings can subsequently evolve in an organic manner. Thus, a policy based management system can be introduced by merely modelling the entire organisation as a single community, with authority over the full set of resources managed by the system. As the need arises sub-communities can be created and responsibility can be delegated to these sub-communities for specific resources. An analysis of policy conflicts can be used to signal structural problems in the organisation model and in the underlying real-world community, thus providing constant feedback to refine the model.

An academic environment can be used to illustrate this organisation modelling process. A university can have several departments, including a computer science department, which can itself have several research groups. In CBPM terms the university is the community. The provost of the university creates sub-communities, including computer science and can then delegate resources and authority to the computer science head of department. In turn, this head of department creates further sub-communities, research groups, and can delegate resources and authority to the director of these groups. This process models the hierarchy that exists in real world organisations. The director of a research group can delegate a resource, such as a computer, to a student within that group. In addition, authority over this computer is

also delegated to the student. The student now has authority over this computer and they can create policies to state who can and who cannot use this computer. If a conflict arises with respect to this computer it can be resolved through the research group director. If conflict still exists the head of department may resolve it. The provost can have final say if the head of department cannot resolve the policy conflict. This is how CBPM resolves conflicts.

6.2.2 PUDECAS Ubiquitous Computing Simulator

The PUDECAS platform, as presented in [O'Neill et al, 2005] is a modification of the Half-Life 2 game engine which provides a 3D virtual representation of pervasive computing environments. The Half-Life 2 SDK tools are used to create environmental models for use with the platform; the largest model created to date replicates the Lloyd Building in Trinity College Dublin (see Figure 6-2) and contains 104 fully furnished rooms. At runtime, players navigate through the environment encountering objects such as doors, tables and desktop computers.

The primary function of the PUDECAS platform is to supply environmental and user context to real-world ubiquitous computing services for the purpose of experimentation and evaluation. This functionality is provided through the inclusion of embedded sensors in the virtual environment which are activated at runtime by player movements and generate a flow of data/context about the environment, users and devices.



Figure 6-1 Real World Lloyd Building Digital Photograph



Figure 6-2 PUDECAS Simulator Lloyd Building Screenshot

Figure 6-1 presents a digital photograph of the real world Lloyd building, whereas Figure 6-2 presents a screenshot from the PUDECAS ubiquitous computing simulator.

6.2.3 Enhanced Instant Messaging (IM) and Content Based Networking (CBN)

The enhanced IM application presented in [Kenny et al, 2006] is very similar to a standard commercial IM application, such as Microsoft's Windows Live Messenger [WLM]. In a standard commercial IM application an IM user can build and maintain a list of friends, work colleagues, and so on. This list is referred to as a 'buddy-list'. The IM user can then communicate with buddies using the IM application. Such functionality is provided by the enhanced IM application used as part of the trials in this thesis. However, this enhanced IM application also allows each user to request presence information for all other users. For experimental purposes, the PUDECAS ubiquitous computing simulator can be used as the source of such presence information for a user.

The enhanced IM application is based on JABBER IM and uses the Extensible Messaging and Presence Protocol (XMPP) [Saint-Andre, 2004] for streaming XML elements. XMPP establishes a universal messaging address which supports the concept of presence [Saint-Andre, 2004], which allows JABBER clients to ascertain what client applications are online and their status. The JABBER IM platform has been combined with a Content Based Networking (CBN) [Segall et al, 2000] infrastructure, which allows the routing of certain messages between collaborating users to be relayed over a decentralised network of content-based routers rather than from a single centralised IM server. The CBN used in the enhanced IM application is Elvin [Segall et al, 2000]. CBNs provide content-delivery via a publisher/subscriber model [Segall et al, 2000] and routing decisions in a CBN are based on content rather than on traditional physical addresses. Subscribers are allowed to express their interest in event content, which provides a much higher degree of robustness to failure and also reduces performance bottlenecks. In the context of the enhanced IM application the CBN provides access to decentralized trust and policy information, and the PUDECAS simulator.

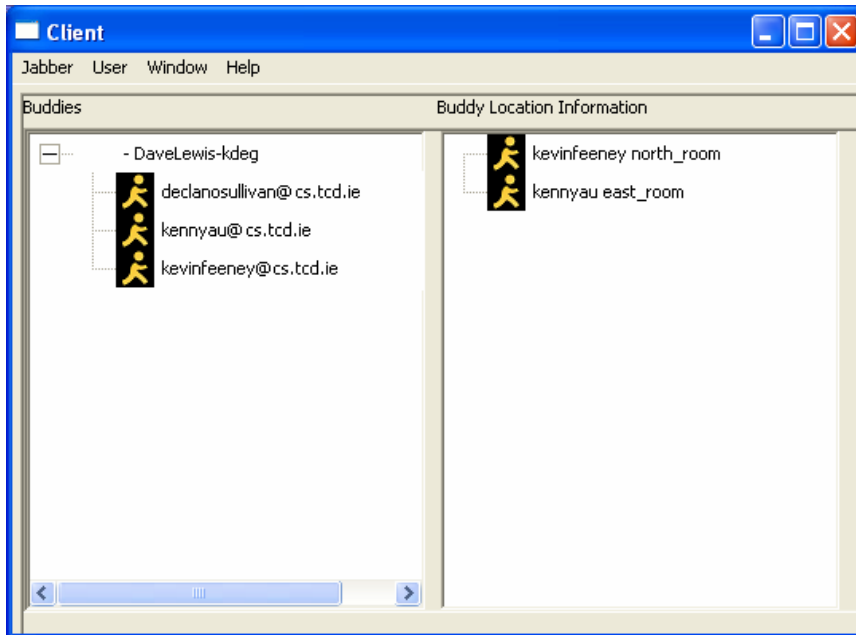


Figure 6-3 Enhanced Instant Messaging Client

Figure 6-3 illustrates the enhanced IM client. The location of a buddy is in fact the location of a virtual user in the PUDECAS ubiquitous computing simulator. Therefore, each user within the simulator has an enhanced IM client and a location.

Figure 6-3 shows the enhanced IM client belonging to *DaveLewis*, a community member of KDEG. The locations of *kevinfeeney* and *kennyau* are *north_room* and *east_room*, respectively. Access control over this location information is carried out by combining *myTrust* and CBPM to provide dynamic and flexible access control to the enhanced IM operating over CBN, and the PUDECAS ubiquitous computing simulator.

6.3 Overview and Comparison of Trials One, Two, and Three

Figure 6-4 presents a comparison chart that describes and compares the three trials under the headings Architecture, Personalisation Algorithm, Trust Annotation, Policy Specification, and Trust Calculation. These heading are also used later in this chapter to describe each of the trials.

	Trial One	Trial Two	Trial Three
Architecture	<i>deepTrust</i>	<i>myTrust</i> , <i>CBPM</i> , <i>PUDECAS</i>	<i>myTrust</i> , <i>CBPM</i> , <i>PUDECAS</i> , <i>enhanced IM</i>
Personalisation Algorithm	Non-HITS	HITS	HITS
Trust Annotation	Ontology Editor	Questionnaire	Questionnaire, GUI
Policy Specification	Internal (text based)	CBPM (text based)	CBPM (text based), GUI
Trust Calculation	Evidence	Opinion	Opinion

Figure 6-4 Trust Systems Comparison Chart

Trial one (see Section 6.3) used a standalone prototype of *myTrust*, called *deepTrust*, which provided trust based recommendations to users in order to allow the user to select Web Services based on the trustworthiness of those Web Services. As per Figure 6-4, a non-HITS based approach to personalisation was used. Instead, personalisation was based on a pre-defined ranking of trust concepts, which resulted in a top three ranking for these trust concepts. An evidence based domain model was specialised towards Web Services and trust annotation was based on it. This trust annotation was carried out using Protégé and many different instances of the Web Service model were annotated. Policy was specified manually and implemented using OWL. Trust data associated with the instances of Web Services were used in calculating an overall trust value.

In trial two (see Section 6.5) the CBPM was combined with *myTrust* to enable access control over resources to be based on trust relationships that existed in the community. A community member could specify a policy for an object that the member had authority over. The policy had trust conditions that must be satisfied in order for a requestor to be granted access to the object. In the PUDECAS ubiquitous computing simulator the objects included doors and other interactive environment objects. Policies were specified for such objects in the simulator, providing dynamic

and flexible access control based on trust. As per Figure 6-4 personalisation was based on the HITS algorithm, trust annotation was made using a web based questionnaire, policy was specified through CBPM, and trust calculations were made using an opinion based domain model.

In trial three the CBPM was combined with *myTrust* as per trial two. However, in this trial policies were specified for location information associated with an enhanced IM user. This location information was provided by PUDECAS. Therefore, this trial extends the research work by provided a dynamic and flexible access control mechanism for location information, based on trust, for a new application domain. As per Figure 6-4 personalisation was again based on the HITS algorithm and trust annotation was made using a web based questionnaire. However, trust annotation could also have been made using a Graphical User Interface (GUI) that was developed for this trial. Policy was specified through CBPM or GUI, and trust calculations were made using the opinion based Instant Messaging domain specific model of trust.

6.4 Trial One - Trustworthy Service Selection

6.4.1 Outline of Trial One

Trial one was conducted in order to examine the feasibility of providing trust recommendations based on a multi-faceted model of trust that is personalisable and specialisable. The goals were to (i) design and develop a Web Services domain specific model and (ii) build a prototype application to illustrate trustworthy service selection, and (iii) create instances of Web Service trust annotation data in order to validate that trust recommendations and policy decisions were as expected with respect to the trust data and trust policy used in the trial. Goals one and two were met through an initial design and development phase in order to carry out goal three.

User	Required Overall Trust Value	Trust Policy Specifics
user1	= <i>very high</i>	<i>downtime < 10.0</i>
user2	>= <i>very low</i>	<i>latency <= 0.005</i>
user3	>= <i>low</i>	<i>transactionTime <1</i>

Figure 6-5 Trust Policies for Trial One

The trust policies used in trial one are presented in Figure 6-5. Policies for three users were created and each had different overall trust requirements and difference trust policy specifics.

Sub-class	Property	Web Service A	Web Service B	Web Service C
Assurance	correctExecution	true	true	true
Availability	MTBF	1,000,000.00	1,000,000.00	1,000,000.00
Availability	downtime	8,500	9	100
Availability	resilience	low	very high	high
Performance	executionTime	0.01	0.001	0.005
Performance	latency	0.5	0.007	0.005
Performance	transactionTime	4	2	0.9
MsgDelivery	atLeastOnce	true	true	true
MsgDelivery	atMostOnce	false	true	false
MsgDelivery	exactlyOnce	false	true	false

Figure 6-6 User *reliability* Trust Data for Web Service A, B, and C

Three instances of Web Service A, B, and C were created and are presented in Figure 6-6. For brevity, *reliability* values are only shown.

User	Web Service	Expected
user1	A	<i>not selected</i>
user1	B	<i>selected</i>
user1	C	<i>not selected</i>
user2	A	<i>selected</i>
user2	B	<i>not selected</i>
user2	C	<i>not selected</i>
user3	A	<i>not selected</i>
user3	B	<i>not selected</i>
user3	C	<i>selected</i>

Figure 6-7 Expected Outcomes for Trial One

The expected outcomes for each user with respect to each Web Service are presented in Figure 6-7. These expectations were derived from the policy data in Figure 6-5, the user trust data in Figure 6-6, and the overall trust values that *deepTrust* was expected to provide for each Web Service.

6.4.2 Architecture and Mechanisms

From May to August 2004 the author of this thesis undertook an internship with Ericsson Research Group, Ireland. In this timeframe a prototype system, *deepTrust*, was developed that would evolve to become *myTrust*. In essence, the system developed at this time was not only a prototype but it was also a proof of concept. The prototype provides a mechanism for the selection of trustworthy Web Services.

The development of *deepTrust* required the design and development of (i) a Web Service domain specific model of trust, (ii) a set of instances that conformed to this domain model, (iii) a set of policies that reflected several users trust requirements for a Web Service, (iv) a trust calculation algorithm, and (v) an application to link these components in order to calculate an overall trust value for a Web Service as per the scenario outlined in Figure 6-8. Figure 6-8 illustrates the four steps required in order for *deepTrust* to provide a trust recommendation from a high level perspective;

- (1) *deepTrust* parses the Web Service domain document for the structure of sub-classes and properties. This refers to sub-classes such as *reliability* and properties such as *meantimeBetweenFailure*.
- (2) One or more sets of trust data, based on other user's experience, are retrieved.
- (3) A user's trust requirements are then parsed from a policy document, which could state for example that the minimum level for *reliability* is *high*.
- (4) *deepTrust* reasons about the domain model, policy document, and trust data to calculate a final trust recommendation. The calculation is based on the user's individual view of trust. *deepTrust* bases its calculations on the user's top three ranked trust concepts.

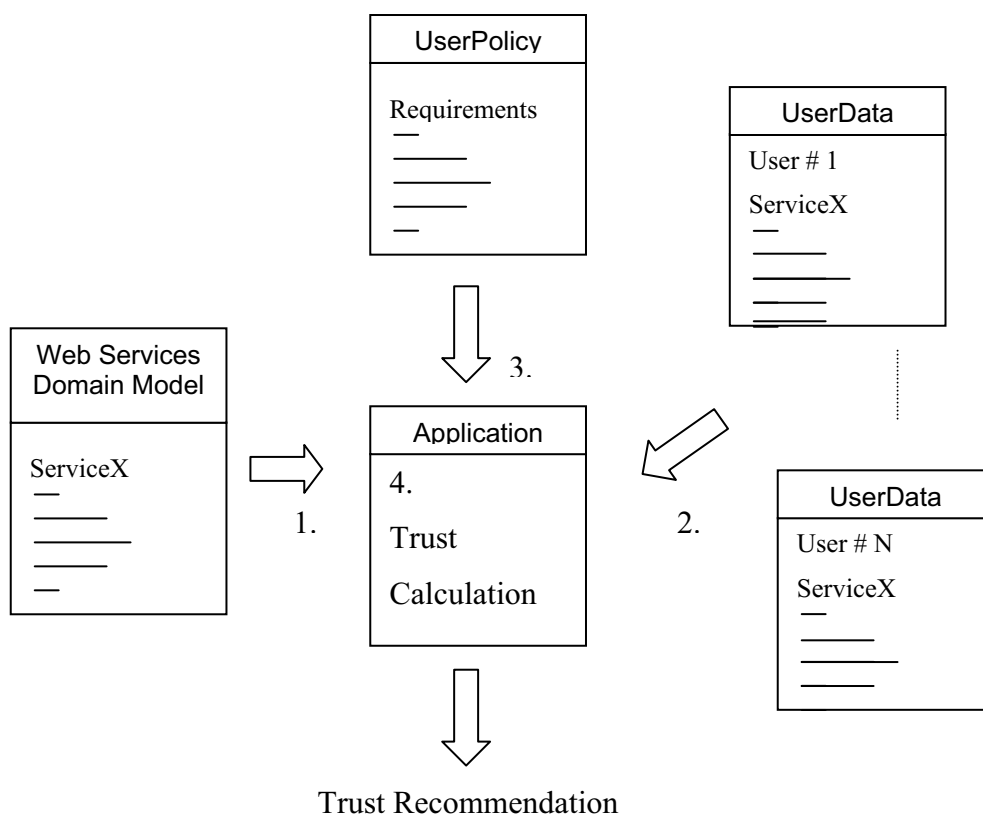


Figure 6-8 Calculation Methodology

The Protégé based trust annotation mechanism that *deepTrust* provided is presented in Section 6.4.2.1. OWL based policy specification is presented in Section 6.4.2.2. The non-HITS based personalisation mechanism is presented in Section 6.4.2.3. Finally, the trust calculation algorithm is described in Section 6.4.2.4.

6.4.2.1 Trust Annotation Mechanism

The domain specific model used in the first trial is the evidence based Web Services domain model (see Section 3.4.4.2). In the Web Services domain model the trust concept *reliability* is made up of the following sub-classes; *assurance*, *availability*, *performance*, and *msgDelivery*. These sub-classes in turn are made up of a set of properties. For example, the sub-class *availability* has a set of properties that includes *downtime*, *meanTimeBetweenFailure*, and *resilience*.

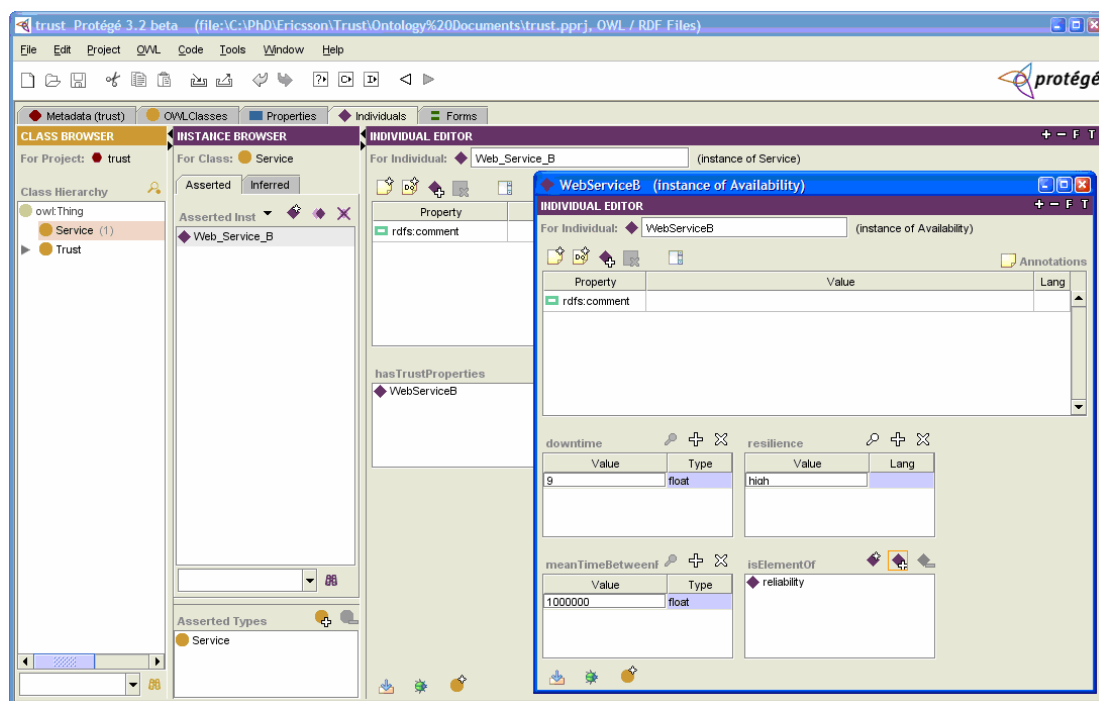


Figure 6-9 Web Service B Protégé based Annotation

A web service can be annotated with respect to the Web Services domain model. In Figure 6-9 the *reliability* of Web Service B is annotated with a *downtime* value of 9 seconds, a *meanTimeBetweenFailure* value of 1,000,000 seconds, and a *resilience* value of *high*. This annotation was carried out manually in Protégé for evaluation purposes in the prototype system. It may be possible to automate this process but such development was never undertaken during the internship period.

6.4.2.2 Trustworthy Service Selection Policy Specification

The policies used by *deepTrust* are developed manually, in Protégé, which results in the output of an OWL document. The policy shown in Figure 6-10 states that the *overallTrust* value of a Web Service must be at least *high*, and *downtime* has to be less than 10.0 seconds. Therefore, in this policy a Web Service will only be selected if it has a *high overallTrust* value and a *downtime* value less than 10.0 seconds.

```
<owl:Class rdf:ID="&trust;UserPolicy">
  <rdfs:subClassOf rdf:resource="&trust;Service"/>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:hasValue
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">high</owl:hasValue>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:about="&trust;overallTrust"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>

  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:hasValue
rdf:datatype="http://www.w3.org/2001/XMLSchema#float">10.0</owl:hasValue>
      <owl:onProperty>
        <owl:DatatypeProperty rdf:about="&trust;downtime"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>

</owl:Class>
</rdf:RDF>
```

Figure 6-10 Example Trust Policy for Selection of a Web Services

6.4.2.3 Personalisation Mechanism

Please note that personalisation was not provided using the HITS algorithm. Instead, in this prototype the trial users chose their top three trust concepts from all eight trust concepts found in the upper ontology.

6.4.2.4 Trust Calculation Algorithm and Example

The trust calculation algorithm uses the three most highly rated trust concepts and associated trust data for these concepts, and policy data for the user. As an example of such a trust calculation consider the data in Figure 6-11 and Figure 6-12.

Sub-class	Property	Web Service B
Assurance	correctExecution	true
Availability	MTBF	1,000,000.00
Availability	downtime	9
Availability	resilience	very high
Performance	executionTime	0.001
Performance	latency	0.007
Performance	transactionTime	2
MsgDelivery	atLeastOnce	true
MsgDelivery	atMostOnce	true
MsgDelivery	exactlyOnce	true

Figure 6-11 User *reliability* Trust Data for Web Service A, B, and C

Figure 6-11 presents a set of user trust data for Web Service B with respect to each of the four sub-classes of the trust concept *reliability*. For example, the property *correctExecution* is part of the *Assurance* sub-class of *reliability* in the Web Services domain specific model of trust.

In order to calculate a *reliability* trust value a default trust calculation was developed that took into account the four sub-classes of *reliability*, and the properties of these classes. It was asserted that a *reliability* score of *very high* would necessitate that (i) the average *downtime* of a Web Service would have to be below 10 seconds, (ii) its *resilience* would be at least *high*, (iii) its *latency* would have to be less than or equal to 0.010 seconds, and (iv) its *MsgDelivery* must state that *exactlyOnce* is true. As per Figure 6-11 it can be seen that a calculation with respect to Web Service B would produce a *very high reliability* score. A similar set of assertions exists for *high*, *low*, and *very low* scores. However, these assertions demand less from a Web Service as

the scores reduce. For example, a Web Service may only need a *downtime* value below 10,000 to be recognised as *very low*. In addition, similar calculation algorithms exist for the remaining trust concepts.

rank	user1	user2	user3
1	<i>reliability</i>	<i>reputation</i>	<i>confidence</i>
2	<i>reputation</i>	<i>competency</i>	<i>Reputation</i>
3	<i>honesty</i>	<i>credibility</i>	<i>Belief</i>

Figure 6-12 User Trust Concept Rankings

Figure 6-12 illustrates the rank of the trust concepts in a top three formation for user1, user2, and user3. It states that *reliability*, *reputation*, and *honesty* are ranked number one, two and three for user1. *deepTrust* will calculate an overall trust value for Web Service B based on these three trust concepts. The trust data in Figure 6-11 was used to calculate a *reliability* value of *very high*. Assuming Web Service B also a *very high* score for *reputation* and *honesty*, then it can be determined by averaging the value for the top three trust concepts that Web Service B has an overall trust value of *very high*.

Assume that user1 owns the policy in Figure 6-10, where the required *overallTrust* value was at least *high* and *downtime* could not exceed 10 seconds. *deepTrust* has calculated that Web Service B has a (i) *very high overallTrust* value, (ii) *downtime* less than 10.0 seconds. Therefore, Web Service B successfully meets the policy requirements of user1, and Web Service B is presented to the user as a trustworthy Web Service.

A semantic difference algorithm [Abdul-Rahman et al, 2000] allows the application to offset each calculated trust value by some degree based on previous experiences and recommendations. For example, if a trust data provider tends to over exaggerate a Web Service, stating a *very high reliability* value rather than a *high reliability* value, then future values from this provider may be offset to reflect this semantic difference.

6.4.3 Results and Conclusions

The overall trust value calculated by *deepTrust* for each Web Service was as follows: Web Service A; *low*, Web Service B; *very high*, and finally Web Service C; *high*.

In the case of user1, only Web Service B was selected as it was the only Web Service that met or exceeded the trust policy requirements. The overall trust value for Web Service B (*very high*) met user1's policy requirement (*very high*). In addition, this policy required a *downtime* of less than 10 seconds and the *downtime* value for Web Service B is 9 seconds. Both Web Services A and C exceeded these requirements.

For user2, only Web Service A was selected as the overall trust value for Web Service A (*low*) met user2's policy requirement (*very low*). In addition, this policy required a *latency* of less than 0.005 seconds and the *latency* value for Web Service A is 0.005 seconds. Both Web Services B and C exceeded these requirements.

For user3, only Web Service C was selected as the overall trust value for Web Service C (*high*) met user3's policy requirement (*low*). In addition, this policy required a *transactionTime* of less than 1 second and the *transactTime* value for Web Service C is 0.9 seconds. Both Web Services A and C exceeded these requirements.

User	Web Service	Expected	Actual
user1	A	<i>not selected</i>	<i>not selected</i>
user1	B	<i>selected</i>	<i>selected</i>
user1	C	<i>not selected</i>	<i>not selected</i>
user2	A	<i>selected</i>	<i>selected</i>
user2	B	<i>not selected</i>	<i>not selected</i>
user2	C	<i>not selected</i>	<i>not selected</i>
user3	A	<i>not selected</i>	<i>not selected</i>
user3	B	<i>not selected</i>	<i>not selected</i>
user3	C	<i>selected</i>	<i>selected</i>

Figure 6-13 Actual Results for Trial One

Figure 6-13 shows the expected and actual *deepTrust* Web Service recommendations. The actual answers reflected the expected answers in 100% of test cases. Therefore, it is possible to conclude that the third goal for this trial has been satisfactorily validated; trust recommendations and policy decisions are being carried out as expected with respect to the trust datasets used in the trial.

6.5 Trial Two - Access Control

Trial two allowed *myTrust* services to be used by the CBPM to provide access control over multiple doors within a simulated virtual environment. This simulated virtual environment is generated by the PUDECAS ubiquitous computing simulator in which users can interact with each other as in the real world. The CBPM system was used to develop policies and relationships that reflect a subject's view of the real world and their relationships in the real world. Thus, combining CBPM and *myTrust* provides dynamic and flexible access control over objects in the simulated environment.

The PUDECAS ubiquitous computing simulator has been configured so that the set of virtual doors within its environment would only open if the policy associated with that door was satisfied. This policy, provided by CBPM, checked the trustworthiness of the virtual user that was attempting to open the virtual door. A trust value for that virtual user was provided for by *myTrust*.

6.5.1 Outline of Trial Two

Trial two was conducted to test the combination of *myTrust* and CBPM. The goals were to (i) validate that trust recommendations and policy decisions were as expected with respect to the trust datasets used in the trial, and (ii) illustrate that the combined *myTrust* and CBPM could provide a dynamic and flexible management system.

Target	Authority	Required Trust
Ground Floor	Custodian	\geq <i>very low</i>
First Floor	Custodian	\geq <i>low</i>
KDEG	Daniel	\geq <i>high</i>
KDEG – Rm. 110	Daniel, Gerard	$>$ <i>high</i>

Figure 6-14 Trust Policies

The CBPM system is used to create the policies for regulating access control to selected doors within the simulated Lloyd building. Figure 6-14 shows the four door targets, the authority over each door, and the trust value required to access that door. The door with the most openness is the ground floor main door. The custodian has authority over it and requires *very low* trust or greater in an actor in order for that actor to gain access. The Custodian also has authority over the first floor door and requires *low* trust or greater in an actor to grant that actor permission to access it. In this way the Custodian has authority over public access doors within the Lloyd

building. The CBPM system has reflected this real world situation in its modelling of the hierarchical structure of the Computer Science department in Trinity College Dublin.

The particular groups and individuals that inhabit the building are delegated authority over their particular domains. Therefore, KDEG has been delegated access to the KDEG room and its inner room (Rm. 110). This authority was directly controlled by the KDEG members Daniel and Gerard. Daniel created a policy for the main door. A policy for the inner room that Daniel and Gerard share was created by both of them. In order to access the KDEG door an actor was required to have at least a *high* trust level or more, as perceived by Daniel. For the inner door an actor is required to have more than a *high* trust level, as perceived by both Gerard and Daniel. Therefore, it is possible for policies to be based on one or more perceptions and sources of trust.

Figure 6-15 illustrates the expected actions that the virtual simulator would effect for all four actors across the four target doors, where applicable. These expected actions are based on the policies that were specified for each door within the virtual Lloyd building and on the trust data that is available for all four trust actors.

Actor	Target	Expected Decision
<i>stranger</i>	Ground Floor	Grant
<i>stranger</i>	First Floor	Deny
<i>stranger</i>	KDEG	Deny
<i>stranger</i>	KDEG - Rm 111	Deny
<i>friend of a friend</i>	Ground Floor	Grant
<i>friend of a friend</i>	First Floor	Grant
<i>friend of a friend</i>	KDEG	Deny
<i>friend of a friend</i>	KDEG - Rm 111	Deny
<i>work colleague</i>	Ground Floor	Grant
<i>work colleague</i>	First Floor	Grant
<i>work colleague</i>	KDEG	Grant
<i>work colleague</i>	KDEG - Rm 111	Deny
<i>family member</i>	Ground Floor	Grant
<i>family member</i>	First Floor	Grant
<i>family member</i>	KDEG	Grant
<i>family member</i>	KDEG - Rm 111	Grant

Figure 6-15 Expected Policy Decisions

The only access that *stranger* has in the Lloyd building should be to the ground floor door, assuming that the Custodian has at least *very low* trust or more in *stranger*.

Access for *friend of a friend* should be limited to only public doors based on the annotated trust data for that actor that was annotated on the Custodians behalf.

Daniel has trust values for *work colleague* and *family member* that states that they are trusted enough to gain entry to KDEG, however *friend of a friend* should be denied.

Access to the inner room, Rm. 111, should only be granted to *family member* as this actor is the only actor that has a trust value that exceeds a *high* trust level.

In the trial a human user logged on as one of the four actors within the PUDECAS ubiquitous virtual simulator. Then the human user guided the virtual user along a set path from the ground floor door, through the first floor and KDEG doors, to the KDEG inner room (Rm 111). Details of where a set of movie files that recorded the trial can be found in APPENDIX III – Implementation Code, Trial Data, and Sundry under Trial Data.

6.5.2 Architecture and Mechanisms

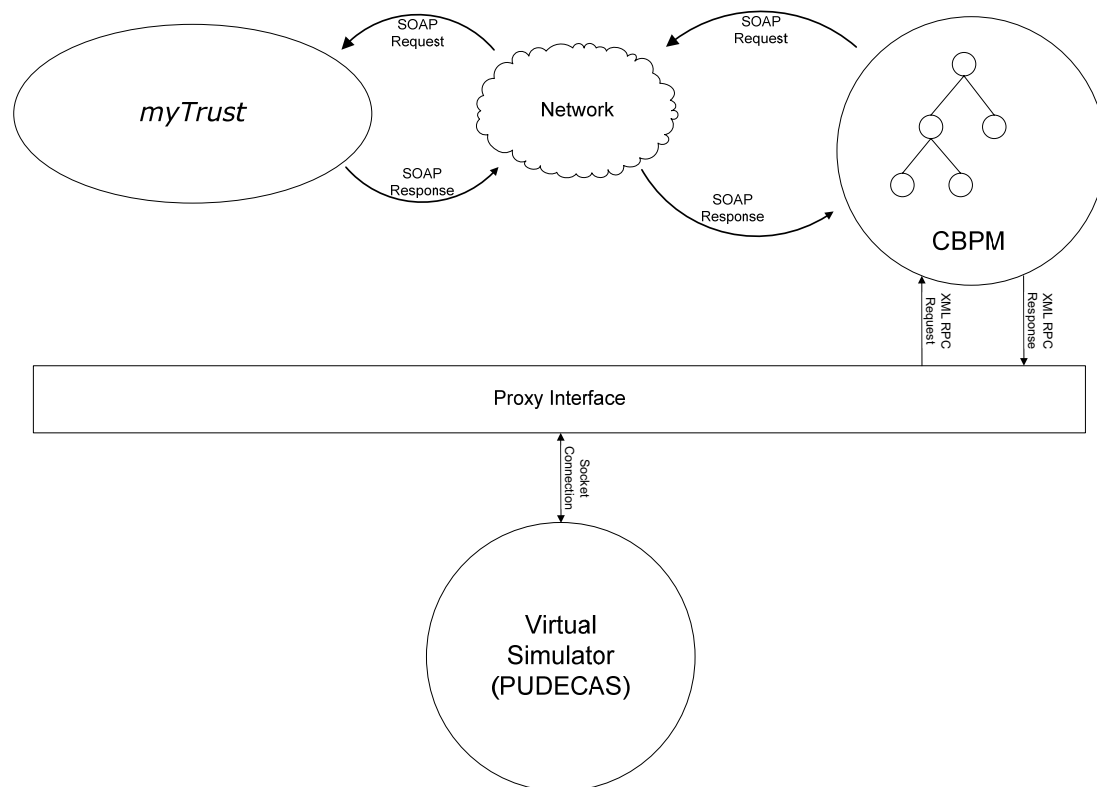


Figure 6-16 Overall Architectural for Trial Two

Figure 6-16 illustrates the overall architecture for trial two. The PUDECAS ubiquitous computing simulator sends and receives information to and from a proxy interface via a socket connection. The CBPM system receives XML-RPC requests from this layer and responds to such requests with XML-RPC. The resulting connection between PUDECAS simulator and the CBPM system enables the CBPM system to provide a policy based access control service to the PUDECAS simulator. The CBPM system uses the exposed web services of *myTrust* to request and receive overall trust values over the network, as described in Section 4.2. For example, CBPM queries *myTrust* with the question ‘How much does Dave trust Austin’. *myTrust* will calculate an overall trust value and return this value to CBPM.

6.5.2.1 Trust Annotation Mechanism

Providing access control to objects within the PUDECAS ubiquitous computing simulator necessitated the development of a mechanism for annotating the simulator users who requested access to these objects.

The domain specific model used in the second trial is an opinion based domain model, which is actually the Instant Messaging domain model (see Section 3.4.4.3). Trust annotation is based on this model as it is simple to achieve and easy to understand for the trial test subjects.

The screenshot shows a web browser window titled "Trust Annotation - Mozilla Firefox". The address bar contains the URL "https://www.cs.tcd.ie/Karl.Quinn/DEG/experiments/surveythree/trustannotationtwo.php". The page content is titled "Work Colleague (2 of 4)". Below the title, there is a horizontal line and a text prompt: "Please assign a particular **WORK COLLEAGUE** with a value for each of the trust concepts below:". The form consists of eight sections, each with a label and four radio button options: "Belief:", "Competency:", "Confidence:", "Credibility:", "Faith:", "Honesty:", "Reliability:", and "Reputation:". Each section has radio buttons for "Very Low:", "Low:", "High:", and "Very High:". A "Next" button is located at the bottom right of the form. The browser's status bar at the bottom shows "Done" and the address "www.cs.tcd.ie".

Figure 6-17 Trust Annotation in Trial Two

Figure 6-17 illustrates the web based trust annotation mechanism that was used in this instance. Each test subject had the ability to annotate one of four pre-defined people with trust data by choosing either *very low*, *low*, *high*, or *very high* for each of the eight trust concepts. In Figure 6-17, the pre-defined *work colleague* is about to be annotated with such trust data. Once annotated, this trust data was stored in a MySQL database, which *myTrust* later used to calculate an overall trust value.

6.5.2.2 Access Control Policy Specification

Policies were created within the CBPM system. These policies had conditions associated with them, which were based on trust values.



Figure 6-18 Resource Tree for Lloyd Building

In Figure 6-18 the resource tree for the Lloyd Building is presented. At the top of the tree are all resources, which are further specialised to a specific target resource called ‘Small Office within office 111’. A target action, ‘open door’, is also associated with ‘Small Office’.

Target Resource	/home/feeneyk/cpms/xml/resources/doorcollection.
Target Path	/all_doors/ff_doors/ff_eastcorridor/ff_kdeg/2_110_1
Target Action	open door
Granted By	kdeg
Granted To	kdeg
Rule Effect	Permit <input type="button" value="v"/>
Conditions	
Description	
SUBJECT(trust[kdeg 1, kdeg 2]) > 3	

Figure 6-19 Policy for 'Small Office within office 111'

Figure 6-19 illustrates the policy for ‘Small Office’. In it a community called ‘kdeg’ have permitted the action ‘open door’ on this ‘Small Office’ to all members of ‘kdeg’. However, this action is only permitted so long as the policy condition is satisfied. The policy condition states that the average trust value of ‘kdeg 1’ and ‘kdeg 2’ (two members of KDEG research group) must be greater than ‘3’, which is *high* trust. Therefore, users with an average trust greater than *high*, as perceived by ‘kdeg 1’ and ‘kdeg 2’, can access ‘Small Office’.

6.5.2.3 Personalisation Mechanism

The second trial used the HITS algorithm to provide personalised trust calculations and a web based mechanism for gathering the personalisation data. Figure 6-20 shows one of the mechanism steps where a user was asked which trust concepts *reliability* influences. In Figure 6-20 the user has stated that *reliability* influences *confidence*, which might be because this user feels that *reliability* gives them a certain level of *confidence*. This was repeated for all eight trust concepts.

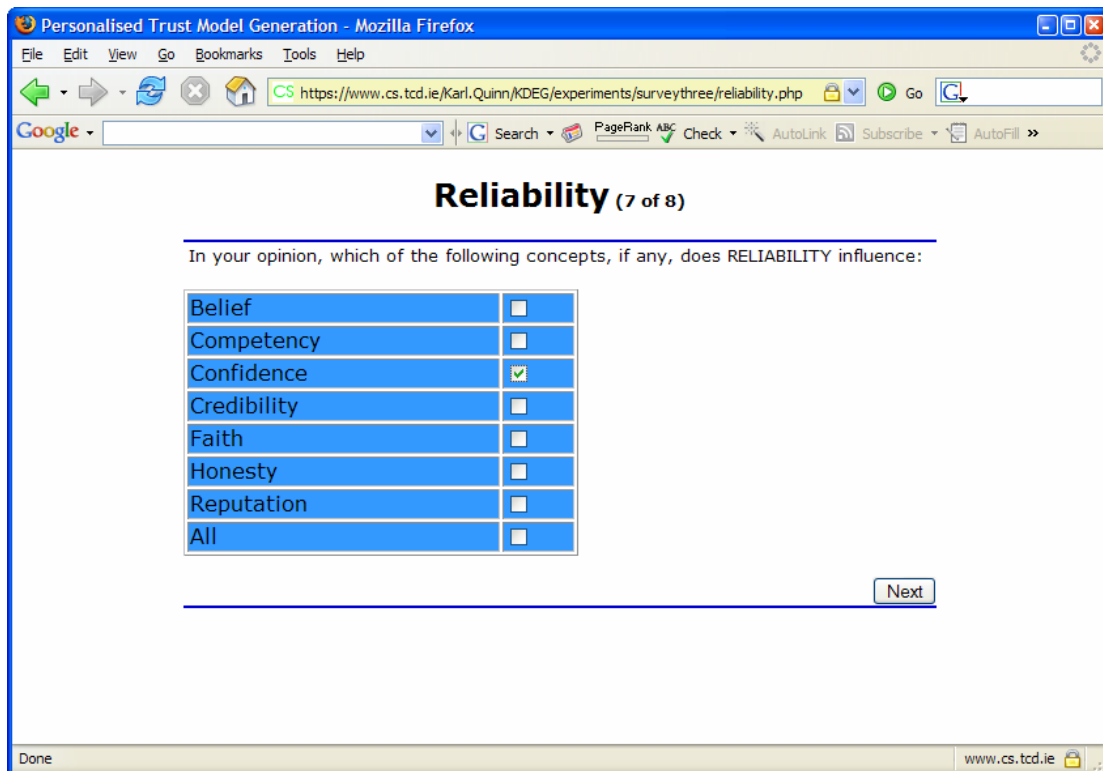


Figure 6-20 Personalisation Mechanism for Trial Two

The full set of personalisation data was then be used to generate a personalised model of trust using the HITS algorithm as outlined in the design chapter (see Section 3.4.3).

6.5.2.4 Trust Calculation Algorithm and Example Operation

The trust calculation algorithm used the opinion based domain model, a personalised model, and a set of trust data to calculate an overall trust value. The calculation algorithm used the three most highly rated trust concepts and associated trust data for these concepts. Figure 6-21 illustrates the weight and rank of trust concepts using the HITS algorithm, and Figure 6-22 provides sample trust data which is used to illustrate the trust calculation algorithm.

	weight	rank
belief	5.91	5/6
competency	5.29	7
confidence	7.32	4
credibility	11.66	1
faith	5.91	5/6
honesty	4.89	8
reliability	10.41	3
reputation	10.91	2

Figure 6-21 HITS Ranked Concepts for User 'kdeg 1'

The trust calculation algorithm will be illustrated with respect to the community member 'kdeg 1'. Figure 6-21 states that 'kdeg 1' ranks *credibility* number one, *reputation* number two, and *reliability* number three.

source user	kdeg 3	kdeg 4	kdeg 5
belief	3	4	3
competency	3	4	3
confidence	4	3	4
credibility	4	4	4
faith	3	3	4
honesty	3	3	4
reliability	3	3	3
reputation	4	4	4
destination user	kdeg 2	kdeg 2	kdeg 2

Figure 6-22 Sample Trust Data for Community Member 'kdeg 2'

Figure 6-22 shows trust data for the community member 'kdeg 2' as annotated by 'kdeg 3', 'kdeg 4', and 'kdeg 5'. The trust calculation algorithm uses the top three concepts of 'kdeg 1' in Figure 6-21 and all three trust data sets from Figure 6-22. The trust calculation would be carried out as follows;

Trust data for the number one ranked trust concept (*credibility*) are added together and divided by the number of trust data sets; $credibility = (4 + 4 + 4)/3 = 4.0$. This occurs for the trust concept ranked number two; $reputation = (4 + 4 + 4)/3 = 4.0$, and again for the trust concept ranked third; $reliability = (3 + 3 + 3)/3 = 3.0$.

The aggregate values for each of the top three trust concepts are then added and the total is divided by 3 (number of trust concepts used), which represents the overall trust value: **Overall Trust Value** = $(4.0 + 4.0 + 3.0)/3 = 3.66$. Therefore, 'kdeg 1' has an overall trust value of 3.66 for 'kdeg 2'. This is not quite *very high* trust, but it is greater than *high* trust and would satisfy the policy presented in Figure 6-19.

6.5.3 Common Data Sets for Trial Two and Trial Three

For both trial two and trial three the trust data, and recommendations were provided by *myTrust*. The trust data itself was sourced from two subjects who took part in experiment two; for anonymity reasons they are called Daniel and Gerard. They granted permission to use their entire survey datasets. In experiment two they both provided sufficient information to generate a personalised model of trust. In addition, they both annotated four actors with trust data. These four actors are the same as those present in trials two and three; *stranger*, *friend of a friend*, *work colleague*, and *family member*. Additional data was instantiated for the fictional user ‘Custodian’. It is *myTrust* that calculated an overall trust value for each of these actors when the CBPM system requested it, which was then reconciled with the conditions found with the policies in order to make an access decision. Figure 6-23 shows the overall trust value that *myTrust* calculated for Daniel and Gerard with respect to each of the four actors. The trust values have been calculated with the personalised model of trust that Daniel and Gerard created in experiment two.

Source	Destination	Trust Value
Gerard	<i>stranger</i>	<i>very low</i>
Gerard	<i>friend of a friend</i>	<i>low</i>
Gerard	<i>work colleague</i>	<i>high</i>
Gerard	<i>family member</i>	<i>high</i>
Daniel	<i>stranger</i>	<i>low</i>
Daniel	<i>friend of a friend</i>	<i>low</i>
Daniel	<i>work colleague</i>	<i>high</i>
Daniel	<i>family member</i>	<i>very high</i>
Custodian	<i>stranger</i>	<i>very low</i>
Custodian	<i>friend of a friend</i>	<i>low</i>
Custodian	<i>work colleague</i>	<i>high</i>
Custodian	<i>family member</i>	<i>very high</i>

Figure 6-23 Trust Values for Trial One and Two

The trust datasets associated with Gerard states that he has a *very low* level of trust in *stranger*. A *low* level of trust is given to *friend of a friend* and both *work colleague* and *family member* are *highly* trusted. Daniel has *low* trust values for *stranger* and *friend of a friend*. A *high* level of trust is present for *work colleague* and a *very high* level of trust is given to *family member*. The fictional Custodian has a *very low* trust value for *stranger* and the trust values increase linearly thereafter to a *very high* trust value in *family member*.

6.5.4 Results and Conclusions

Actor	Target	Expected Action	Actual Action
<i>stranger</i>	Ground Floor	Grant	Grant
<i>stranger</i>	First Floor	Deny	Deny
<i>stranger</i>	KDEG	Deny	Deny
<i>stranger</i>	KDEG - Rm 111	Deny	Deny
<i>friend of a friend</i>	Ground Floor	Grant	Grant
<i>friend of a friend</i>	First Floor	Grant	Grant
<i>friend of a friend</i>	KDEG	Deny	Deny
<i>friend of a friend</i>	KDEG - Rm 111	Deny	Deny
<i>work colleague</i>	Ground Floor	Grant	Grant
<i>work colleague</i>	First Floor	Grant	Grant
<i>work colleague</i>	KDEG	Grant	Grant
<i>work colleague</i>	KDEG - Rm 111	Deny	Deny
<i>family member</i>	Ground Floor	Grant	Grant
<i>family member</i>	First Floor	Grant	Grant
<i>family member</i>	KDEG	Grant	Grant
<i>family member</i>	KDEG - Rm 111	Grant	Grant

Figure 6-24 Expected vs. Actual Actions

Figure 6-24 shows the actions that were expected and the actions that took place within the simulator. The actual answers reflected the expected answers in 100% of test cases. Therefore, it is possible to conclude that the second goal for this trial has been satisfactorily validated; trust recommendations and policy decisions are being effected as expected with respect to the trust datasets used in the trial. The first goal for this trial was to validate that the combined *myTrust* and CBPM could provide a dynamic and flexible management system. In validating the correctness of the operation of the *myTrust* and CBPM combination it has indicated that the system can provide dynamic and flexible management. The arguments to support this claim are based on the CBPM and *myTrust*, respectively. The CBPM system is flexible in that it allows organisations to model their organisational structure and re-organise it as necessary. Authority over resources, such as doors or location information, can then be delegated to reflect the organisational hierarchy. Trust relationship and values that exists between entities within the organisation are created, altered, and removed dynamically. These relationships and associated trust data can dynamically change, which can be captured in *myTrust*. It is possible for *myTrust* to calculate trust values in the face of this dynamic change, which the CBPM system can use when reconciling policies in order to provide access control.

6.6 Trial Three - Instant Messaging

Trial three expanded the use of the *myTrust* and CBPM combination in order to regulate access control over location information within an enhanced IM application. The PUDECAS ubiquitous computing simulator provided a rich set of location information for virtual users within the simulated environment. Initially, access control over this location information was based on policies that used a community membership paradigm, but *myTrust* and CBPM enables access control to be based on policy and trust. This trial illustrated that the CBPM and *myTrust* combination can provide flexible and dynamic access control across multiple scenarios. In addition, trial three showcases alternative trust annotation and policy specification mechanisms.

6.6.1 Outline of Trial Three

Trial three extended the *myTrust* and CBPM combination to include an enhanced Instant Messenger (IM) client. Each enhanced IM client had a ‘buddy list’ for each user, which created a social network for the trust management service to query for trust data if required. Enhanced IM users could request the location of another enhanced IM user within the virtual simulator. The CBPM was used to create a set of policies that regulated access control over this location information. Again, the policies used trust as the condition. In this way access to the location information is again granted or denied based on trust.

Target	Authority	Required Trust
Location Information	Daniel	\geq <i>low</i>
Location Information	Gerard	\geq <i>high</i>

Figure 6-25 Location Information Policy

Figure 6-25 presents the two policies used for location information in the second trial; one for Daniel and one for Gerard. Daniel will allow someone with *low* trust or greater to gain access to his location information. Gerard will require a *high* level of trust, or greater, for access to his location information. As per trial two these policies were created in the CBPM.

The goal of this trial was to validate that trust recommendations and policy decisions were as expected with respect to the trust datasets used in the trial. In addition, successful operation would demonstrate another use case scenario showing how CBPM and *myTrust* combine to provide flexible and dynamic trust management in Internet Environments.

Actor	Target	Expected Decision
<i>stranger</i>	Daniel	Deny
<i>stranger</i>	Gerard	Deny
<i>friend of a friend</i>	Daniel	Grant
<i>friend of a friend</i>	Gerard	Deny
<i>work colleague</i>	Daniel	Grant
<i>work colleague</i>	Gerard	Grant
<i>family member</i>	Daniel	Grant
<i>family member</i>	Gerard	Grant

Figure 6-26 Expected Policy Decisions

Figure 6-26 illustrates the expected decisions for all four actors with respect to Daniel and Gerard. It is expected that *work colleague* and *family member* will be granted access to both their location information. It is also expected that both will deny *stranger* access to their location information. According to the trust data and policy requirements *friend of a friend* will have access to Daniel’s location information but not to Gerard’s location information.

In this trial a human user logged on as either Daniel or Gerard within the PUDECAS ubiquitous virtual simulator. In addition, a set of human users logged on as one of the four actors. The virtual target (Daniel or Gerard) changed location and each of the actors requested the location information of the targets.

6.6.2 Architecture and Mechanisms

The overall architecture for trial three is an extension of the overall architecture for trial two. The way in which *myTrust* is accessed and utilised does not actually change between trial two and trial three. Instead, in this trial the CBPM is accessed by the enhanced IM application, and not the PUDECAS simulator, in order to see if a requestor is allowed to access the location information of a target user.

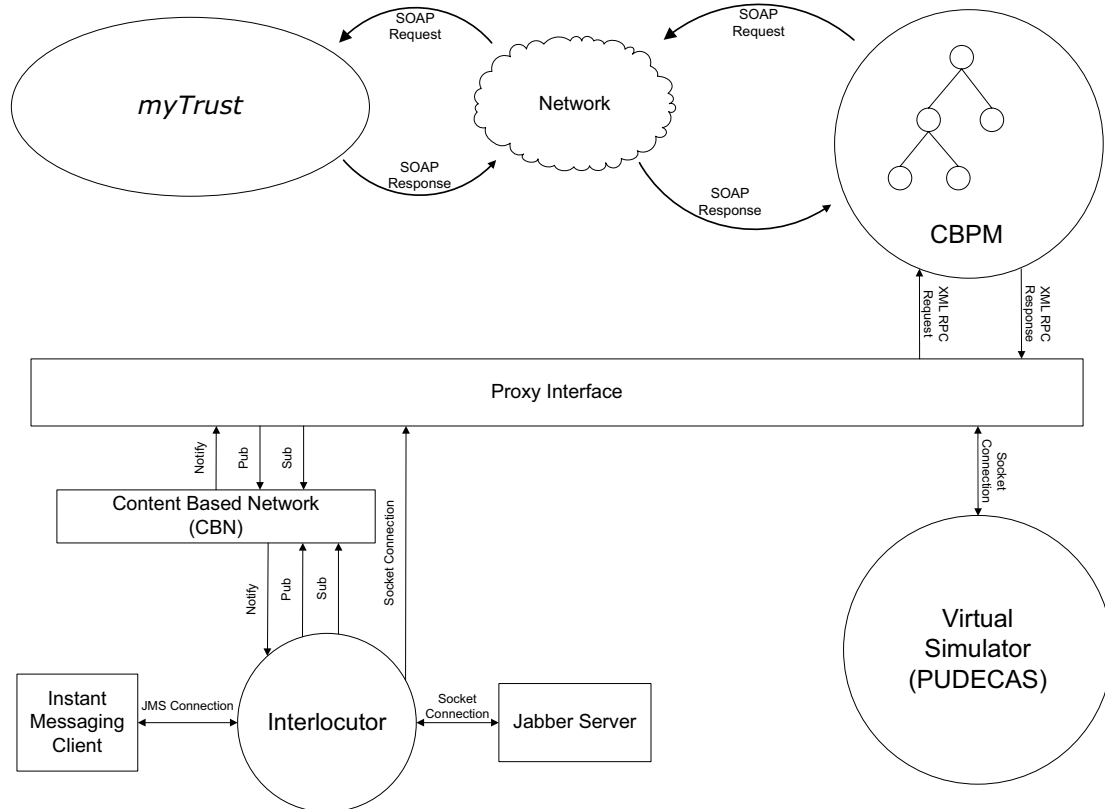


Figure 6-27 Overall Architectural for Trial Three

Figure 6-27 shows the extension to the architecture of trial two (see Figure 6-16). The Interlocutor [Kenny et al, 2006] intercepts requests for access to location information from an enhanced IM client and publishes a request over the Elvin CBN. This request is received by the CBPM system, which subscribes to such requests. The CBPM system then determines whether access should be granted (based on interactions with *myTrust*) and publishes the response, which the Interlocutor receives and adheres to. A virtual user in the virtual simulator has an enhanced IM client through which she can access the location information of other virtual users within the simulator who also have enhanced IM clients. Policies created in the CBPM system regulate access control to such location information, which is in turn based on trust.

The overall trust value was once again calculated using a personalised model of trust, a set of annotated trust data, and a trust calculation algorithm. However, in this trial alternative mechanisms for trust annotation and policy specification are showcased. The personalisation mechanism is the same as the previous instance; a web based gathering of personalisation data and the subsequent HITS based generation of a personalised model of trust.

6.6.2.1 Trust Annotation Mechanism

An alternative method for trust annotation was developed to illustrate that annotation can be carried out in a number of different ways, including via Protégé as in trial one, through a web based interface as in trial two, or as in this trial via a GUI.

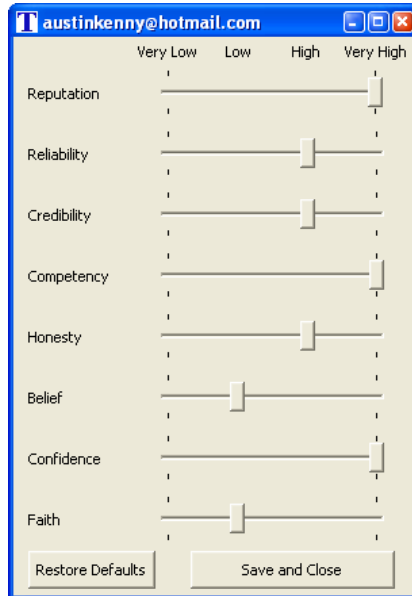


Figure 6-28 Trust Annotation Mechanism in Enhanced IM Application

Figure 6-28 shows sample trust data that a user has annotated about another user, *austinkenny@hotmail.com*. There is a slider associated with each of the eight trust concepts. The slider can be moved from *very low* through *low* or *high* to *very high*. In this way each trust concept can be annotated via a GUI and stored and used in the same way as via the web based mechanism. This GUI, and the policy specification GUI, are implemented using Eclipse's Standard Widget Toolkit (SWT) [SWT].

6.6.2.2 Instant Messaging Policy Specification

The CBPM system is used to specify policy for access control in both trial two and three. However, in order to showcase alternative user interaction mechanisms a GUI was developed that enables users to specify policy for location information. This GUI stored these policies in a MySQL database but a tool was developed to transform this database policy into an OWL document implementation. The CBPM system can be integrated with a range of external policies, and not just its own internal policy implementation. Therefore, it would be possible to specify policy for location information in alternative ways (see Figure 6-29), although CBPM is used in this trial.

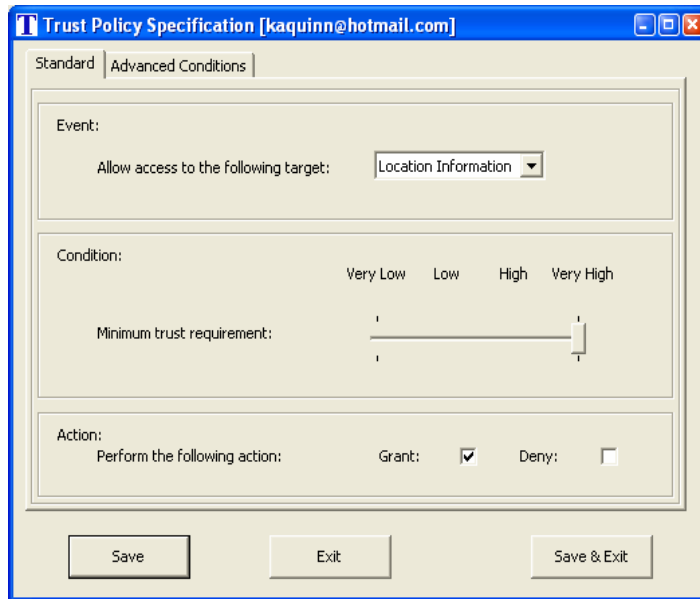


Figure 6-29 Policy Specification in Enhanced IM Application

Figure 6-29 presents an event, condition, action based approach to policy specification. The GUI allows a user to state that access to location information will only be granted if the requesting user meets or exceeds a minimum overall trust value, which is set to *very high* in Figure 6-29.

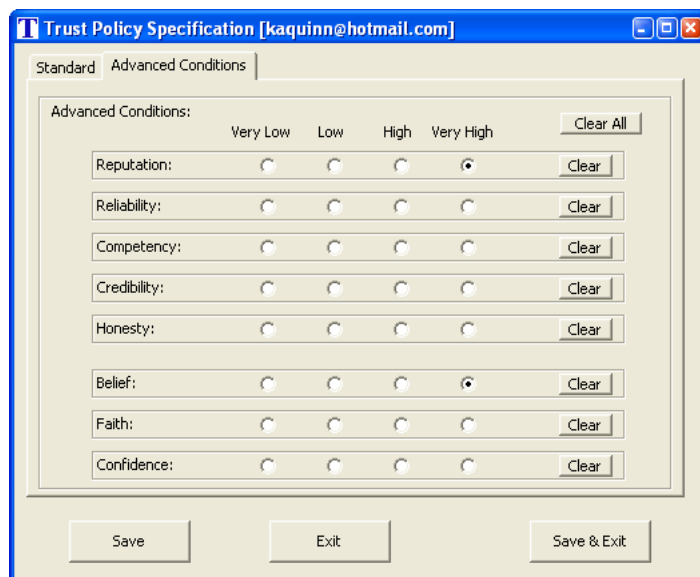


Figure 6-30 Advanced Policy Specification in Enhanced IM Application

A user can also state additional trust conditions. In Figure 6-30 the user has stated that *reputation* must be at least *very high* and *belief* must also be at least *very high*. Both these conditions must be met in addition to the overall trust value.

6.6.2.3 Personalisation Mechanism

Trial three used the HITS algorithm to provide personalised trust calculations. Again a web based mechanism for gathering the personalisation data was used. Once gathered the full set of personalisation data could then be used to generate a personalised model of trust using the HITS algorithm as outlined in the design chapter (see Section 3.4.3).

6.6.2.4 Trust Calculation Algorithm and Example Operation

The trust calculation algorithm uses the Instant Messaging domain model (see Section 3.4.4.2), a personalised model, and trust data to calculate an overall trust value.

	Example 1	Example 2
source user	kdeg 1	kdeg 1
belief	4	3
competency	4	4
confidence	4	4
credibility	4	3
faith	3	4
honesty	4	4
reliability	4	3
reputation	4	4
destination user	kdeg 2	kdeg 3

Figure 6-31 Example Trust Data

By way of explanation three users in an enhanced IM scenario have been created; 'kdeg 1', 'kdeg 2', and 'kdeg 3'. As per Figure 6-31 the source user 'kdeg 1' has trust data for each trust concept for destination users 'kdeg 2' and 'kdeg 3'. The integer trust data representation spans from 1 to 4 where 1 is *very low*, 2 is *low*, 3 is *high*, and is four is *very high*.

	weight	rank
belief	5.91	5/6
competency	5.29	7
confidence	7.32	4
credibility	11.66	1
faith	5.91	5/6
honesty	4.89	8
reliability	10.41	3
reputation	10.91	2

Figure 6-32 Example Rank and Weight Data for 'kdeg 1'

In the enhanced IM scenario the calculation algorithm places the average weight (see Figure 6-32) of the top three ranked concepts in to a 'gold band', the average weight of the concepts ranked four to six into a 'silver band', and the average weight of the

bottom two concepts into a ‘bronze band’. Once again, the weight and rank of a trust concept is provided by the HITS algorithm. This averaging leads to a gold band value of approximately 13, a silver band value of approximately 8, and a bronze band value of approximately 5.

	Band	
Band	Weight	Band Value
Gold	13	4
Silver	8	3.66
Bronze	5	4

Figure 6-33 Band Values for 'kdeg 2'

The trust data that ‘kdeg 1’ holds for ‘kdeg 2’ based on the three bands is now used. The aggregation of the top three trust concepts (see Figure 6-32), which constitutes the gold band, uses *credibility*, *reputation*, and *reliability*, each of which has a trust value of 4. Therefore, the average value is 4, which is the value assigned to the gold band as per Figure 6-33. The same calculation is made for the concepts in the silver and bronze bands, which assigns 3.66 to the silver band and 4 to the bronze.

As per Figure 6-33 the gold band has significantly more weight in relation to the silver band, and in turn the silver band has significantly more weight than the bronze band. The band value in the highly weighted gold band (top three trust concepts) will be used as the overall trust value. Therefore, it can be said that ‘kdeg 2’ is *very highly* trusted. In this example ‘kdeg 2’ meets the minimum overall trust requirement of the policy in Figure 6-29 (*very high*) and also the additional policy condition as per Figure 6-30 (*reputation* and *belief* are *very high*). Therefore, ‘kdeg 2’ would be granted access to the location information of ‘kdeg 1’.

This calculation algorithm is similar to the calculation algorithm used in the second trial. However, the use of the weightings from the HITS algorithm can be used to calculate an overall trust value using trust concepts beyond the top three ranked concepts. There may be cases where the fourth, fifth, and sixth trust concepts may have very similar weight to the top three concepts. In these cases the algorithm could take into account the top six trust concepts to calculate an overall trust calculation that takes into account a larger selection of the trust concepts that the users ranks highest. In addition, there may be cases where the top ranked concept holds the most weight.

6.6.3 Results and Conclusions

Actor	Target	Expected Action	Actual Action
<i>stranger</i>	Daniel	Deny	Deny
<i>stranger</i>	Gerard	Deny	Deny
<i>friend of a friend</i>	Daniel	Grant	Grant
<i>friend of a friend</i>	Gerard	Deny	Deny
<i>work colleague</i>	Daniel	Grant	Grant
<i>work colleague</i>	Gerard	Grant	Grant
<i>family member</i>	Daniel	Grant	Grant
<i>family member</i>	Gerard	Grant	Grant

Figure 6-34 Actual vs. Expected Actions

Figure 6-34 shows the outcome of the third trial. The actual actions, as ascertained via the enhanced IM client, reflected the expected actions in 100% of test cases. Thus, it is possible to conclude that the goal of the trial has been satisfactorily achieved; access to location information has been regulated based on trust and policy and the expected outcomes have been verified.

6.7 Comparisons of myTrust to Related Work

Figure 6-35 presents the comparison framework chart from Chapter 2 (see Figure 2-6) with the addition of the *myTrust* trust management system. This section compares *myTrust* in relation to the selected state of the art systems.

		REFEREE	SULTAN	OpenPrivacy	TRELLIS	Fidellis	FOAF extended	Advogato	eBay	Amazon	Slashdot	Epinions	FilmTrust	<i>myTrust</i>
Model of Trust	Single-Faceted	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	
	Multi-Faceted				✓									✓
	Personalisable													✓
	Specialisable by developer	✓	✓		✓	✓	✓					✓	✓	✓
	Specialisable by user													✓
Trust Annotation	Opinion	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Evidence	✓	✓		✓	✓						✓		✓
Trust Calculation	Simple	✓	✓	✓	✓	✓			✓	✓	✓	✓		✓
	Advanced						✓	✓					✓	✓
Policy	Internal	✓	✓	✓		✓					✓			✓
	Separate													✓
	Not Used				✓		✓	✓	✓	✓		✓	✓	
Architecture	Centralised	✓	✓		✓			✓	✓	✓	✓			
	Distributed			✓		✓	✓					✓	✓	✓
Trust Representation	XML				✓									
	OWL				✓		✓						✓	✓
	Other	✓	✓	✓		✓		✓	✓	✓	✓	✓		

Figure 6-35 Comparison Framework Chart with *myTrust*

The model of trust that *myTrust* uses is multi-faceted, personalisable, and both developer and user specialisable. Such a model of trust is not found in, or used by, any other state of the art trust management system. In this comparison framework TRELLIS provides the closest comparison with respect to the model of trust category. TRELLIS provides both a multi-faceted and specialisable model of trust. However, specialisation is restricted to the developer, whereas *myTrust* enables the developer and user to specialise a model of trust. Enabling the average end user to design and develop specialised models of trust creates an environment for community participation, and community verification, of specialised trust models. TRELLIS, like all systems in Figure 6-35, does not provide personalisation, as defined in this thesis, within the model of trust. This type of personalisation enables the model of trust to capture the subjectivity of trust at the individual level and at the same time capture the

wide and diverse range of views of trust across a large and broad population. This is key to providing a trust management system that can provide a bespoke, tailored or personalised service to a large population. Therefore, it is *myTrust* that is unique across the systems presented in Figure 6-35 in its use of a model of trust that is multi-faceted and is personalisable and both developer and user specialisable.

In chapter two it was asserted that trust annotation and trust calculation were key requirements for a trust management system. As per Figure 6-35 trust annotation can be opinion or evidence based and trust calculations can be simple or advanced. It is important to provide both an evidence and opinion based approach to trust annotation as (i) it can be seen from the state of the art that both approaches are used, and (ii) both approaches can cover the wide range of application domains. *myTrust* provides an evidence based trust annotation with the availability of advanced trust calculations. The complexities of an evidence based approach sometimes require advanced trust calculations, as per the Web Services trust calculation algorithm presented in Section 6.4.2.4. FOAF extended, Advogato, and FilmTrust are the only selected systems that provide advanced trust calculations, yet not one of these systems also provides evidence based trust annotation. Of all the reviewed state of the art systems it is *myTrust* that is alone in providing evidence based trust annotation as well as advanced trust calculations. In addition, *myTrust* that is unique in providing both evidence and opinion based trust annotation, and both simple and advanced trust calculations.

A smaller proportion of reviewed trust management systems use policy. All these systems; REFEREE, SULTAN, OpenPrivacy, Fidelis, and Slashdot all provide policy that is internal to the trust management system. *myTrust* also provides an internal policy mechanism as presented in Section 6.6.2.2. An internal trust management system enables users to state rules that use trust values in order to provide management functionality, such as access control. However, the policy approach used by these systems tends to be based on a basic event, condition, action mechanism or the more advanced Role Based Access Control (RBAC). More advanced approaches can offer additional benefits such as a reduced overhead or easier administration. However, combining the Community Based Policy Management (CBPM) and *myTrust* offers a level of flexibility and dynamicity for management functionality that is unparalleled in the reviewed state of the art trust management systems.

The architecture of *myTrust* has been design as a distributed system. For the purposes of this PhD thesis the *myTrust* implementation has used Enterprise Java Bean's, which are inherently distributed. In addition, the ability to seek trust data from a friend of a friend has been implemented and is available in *myTrust*. For example, in the enhanced Instant Messaging application it is possible for a user to seek trust data not only from a friend in their buddy list but also from a friend in another friends buddy list. In this way the trust data is sought, and calculations are made, in a similar approach to FilmTrust. The benefits of such an approach are (i) the likelihood of retrieving trust data increases as a larger population is available and (ii) trust calculation made using this trust data may mean more to the user who makes a decision based on this trust data, as a path may exists from the requestor of that trust data to the provider of that trust data. However, in must be noted that the trials to date do not retrieve trust data beyond a direct relationships.

With regards to trust representation TRELIS, FOAF extended, and FilmTrust all use an OWL based approach. OWL is also used as the trust representation format for *myTrust*. Some of the benefits of using an OWL approach include; extendibility, reusability, and the easy sharing and understanding of model of trust components and trust data.

An overall analysis of *myTrust* across all reviewed trust management systems that appear in Figure 6-35 illustrates that *myTrust* at least matches, or exceeds, all other trust management systems in relation to provisions made for (i) model of trust, (ii) trust annotation, and (iii) trust calculation categories. However, more research work and development is required to have a fully completed and tested distributed architecture for *myTrust*.

6.8 Summary

The first trial illustrated the initial prototype of *myTrust*, which served as a proof of concept. The first trial saw the design and development of (i) Web Services domain specific model, (ii) manually created personalised model, (iii) Protégé based trust annotation mechanism, (iv) early trust calculation algorithm, and (v) OWL based policy representation. The second trial used the combination of *myTrust* and CBPM to provide access control to objects within the PUDECAS virtual environment. The third trial used the combination of *myTrust* and CBPM to provide access control over location information about users in the enhanced IM application.

The three trials illustrated the feasibility, and successful operation, of the implemented trust management system, *myTrust*, which is based on the multi-faceted model of trust that is personalisable and specialisable. The second and third trials showed that the combination of *myTrust* and CBPM provides a flexible mechanism to manage an organisation while at the same time handle the dynamic nature of trust and the relationships found within trust.

In addition to the three trials this chapter also presented a comparison of *myTrust* and related work. This comparison shows that *myTrust* is unique in using a multi-faceted model of trust that is personalisable and both developer and user specialisable. In addition, it illustrated that *myTrust* is unique in providing both opinion and evidence based trust annotation and both simple and advanced trust calculations.

7 CONCLUSIONS

This chapter presents a discussion of how well the objectives of this thesis were achieved (Section 7.1), the contribution made (Section 7.2), ideas and options for future work (Section 7.3), and concludes with some final remarks (Section 7.4).

7.1 Objectives and Achievements

The research question posed in this thesis is *whether a multi-faceted model of trust that is personalisable and specialisable is both necessary and accurate to the user in providing a dynamic and flexible trust based decision support mechanism within Internet environments.*

In order to investigate the research question the following goals were derived:

1. Research the state of the art in trust, focusing primarily on models of trust and trust management systems in order to identify whether there is a consensus on what trust is and how trust management operates.
2. Design and develop a multi-faceted model of trust that is personalisable and specialisable.
3. Evaluate the necessity for a multi-faceted model of trust that is personalisable and specialisable.
4. Design and develop a trust management service that has a mechanism for generating personalised models of trust, which provides trust based recommendations. Evaluate the ability of the generation mechanism to produce personalised models of trust that accurately reflect users' ideas of trust. Evaluate the accuracy of trust based recommendations calculated using the developed trust management service.
5. Develop two case studies to illustrate and compare specialisation in an evidence based application domain and in an opinion based application domain.
6. Illustrate the ability of the model of trust to provide dynamic and flexible management by providing a trust management service to a policy based management system as part of a use case scenario.

How well each objective was achieved is described in the following discussion.

State of the Art Research

The initial state of the art review in 2003 and the continued review to 2006 was undertaken for objective one and it illustrates that there has been, and there is, no real consensus on the meaning of trust as used in computer science. This view is supported by [Grandison, 2003] and is evident across the research work of [Golbeck & Hendler, 2004], [Shadbolt, 2002], [Golbeck et al, 2003], [Grandison & Sloman, 2000], and [McKnight & Chervany, 1996]. It must be noted that there may be additional trust concepts, such as loyalty, which could be found in the state of art but are not already part of the eight trust concepts used in the trust management system to date. However, the eight trust concepts are fairly representative of what is found within the current state of the art in trust management. The state of the art review of trust management systems found that the most common model of trust is a single-faceted approach, and the dominant trust concept used in this single-faceted approach is reputation. However, one trust management system, TRELIS, uses a multi-faceted approach, which uses the trust concepts reliability and credibility. No further research work in the state of the art has advanced a multi-faceted approach to modelling trust. In addition, the literature review illustrated that people have subjective views of trust, yet no personalisation exists within these models of trust to reflect the subjective nature of trust. Furthermore, it was found that several state of the art trust management systems use specialised models of trust. However, in these cases specialisation is carried out only by the developer of the trust management system.

The model of trust proposed in this thesis addresses several gaps in the state of the art by designing and advancing a multi-faceted approach to modelling trust, which allows all users to personalise their model of trust and provides the option for the users themselves to specialise the model of trust towards multiple application domains.

Design and Development of Model of Trust

This thesis proposes a multi-faceted model of trust that is personalisable and specialisable and was developed in response to the second objective. The model of trust is accomplished through a multi-faceted upper-ontology and meta-model (see chapter 3), personalised model generation mechanism (see chapter 3), and specialised application domain models (see chapter 4). The model of trust was designed and constructed in a way that addressed the gap in the state as outlined above. In addition,

it meets the desired set of properties that were outlined in Section 3.2.1 and also illustrated below:

- Capture the wide and varied views of trust that can exist across a large and broad population,
- Capture the subjectivity of trust at the individual level found within a large and broad population,
- Engineer multiple, specialised, application domain models,
- Build upon current models of trust and have the ability to be extended.

The design of the model of trust is split into four distinct models; (i) upper ontology, (ii) meta-model, (iii) personalised model, and (iv) specialised model. Such a separation of concerns enables each distinct model to be used independently, yet also in various combinations with each other. Separation enables the independent use and extension of the upper ontology. The upper ontology and meta-model are used to generate a personalised model of trust, and to engineer a domain specific model of trust. The specialised model of trust is used as part of a trust annotation mechanism. A personalised model of trust is used in conjunction with trust annotation data to calculate trust values. The model was developed using the OWL ontology language that enabled the specification of a rich set of relationships, classes and properties, which reflects the design of the model of trust while also making the model highly interoperable.

Providing personalisation and specialisation within a multi-faceted model of trust requires more input from the user than single-faceted approaches require. For example, it is necessary for the user to provide input in order to build a personalised model of trust. In addition, a system that would use a multi-faceted model of trust that is personalisable and specialisable would also have additional computational and administration overheads, which have not been empirically measured in this thesis. However, the author of this thesis believes that the advantages of personalisation outweigh the required additional user input and system overheads.

Evaluation of Necessity of Model of Trust

Experiment one was developed to address objective three. The experiment was designed in conjunction with Dr. Deirdre Bonini, Psychology Department, Trinity College Dublin and 279 test subjects' participated. The experiment illustrated that (i) a multi-faceted, personalisable model of trust is required, (ii) trust concepts can be categorised as *abstract* or *concrete*, and that (iii) as risk increases so too does the subjects regard for the usefulness of the trust concepts rise. In addition, the results of experiment one also indicate that there is a wide and varied range of personal views on what constitutes trust, which this experiment has empirically measured. Currently, to this author's knowledge there is no other similar evaluation with regards trust concepts. However, it is important to note that the analysis of this experiment was based on information from subjects who had, in 90% of cases, purchased something online. In addition, approximately 80% of test subjects were aged between 20 and 40 years old. Therefore, readers of the results from first experiment should take into account that the majority of subjects have bought something online and are under 40 years old. To date, no experiments have been developed and deployed that focus on a set of subjects over 40 years old. Such an experiment could be used to investigate whether the results of a large number of people over 40 years old are consistent with, or different to, a large number of people under 40 years old.

Evaluation of Accuracy of Personalised Model of Trust

The personalisation mechanism developed to meet objective four has been evaluated in terms of its accuracy in providing trust based recommendations, and also in terms of its ability to reflect a user's model of trust. Experiments two and three were developed for this evaluation. Each of these experiments was completed by over two hundred subjects and used a personalised model of trust for each one of these individual subjects.

The second experiment (Accuracy of Model of Trust) has illustrated that the accuracy of recommendations decreases from approximately ninety five percent accuracy to seventy percent accuracy as the amount of risk increases from *very low* to *very high*. An analysis of the experiment data also revealed a set of higher risk areas that were of low accuracy. The third experiment (Accuracy of Model of Trust with Additional Information) allowed the user to request addition information. This resulted in greater

overall accuracy and also avoided the steady decline in accuracy from *very low* to *very high* required trust as seen in experiment two. However, it was also illustrated that providing additional information can reduce the accuracy of recommendations in areas of lowest risk. In contrast, there are improved results in the set of higher risk areas that were of low accuracy. It can be concluded that, in this experiment, additional information should not be provided when considering very low risk scenarios. Furthermore, in this experiment, additional information should be provided in areas of higher risk. However, it is too early to state definitively that the provision of additional data will always alter the accuracy of recommendations in these ways. Therefore, it would be beneficial to develop and deploy more experiments of this nature. In particular, further experimentation that provides new and more potent sets of additional information, such as legal guarantees or penalties, could be carried out to investigate the levels of accuracy that could be achieved with such information.

The HITS algorithm was evaluated as a mechanism to generate personalised models of trust. Aggregated experiment data taken from experiment one (where models were built via direct questioning) was analysed against aggregated experiment data taken from experiment two (where models were generated via HITS algorithm). It was found that personalised models of trust, generated using the HITS algorithm, produced a set of aggregated rankings that generally reflected the aggregated rankings of concepts when a broad set of subjects were directly asked for rankings. Thus, this indicates that the two sets of aggregated experiment data produced similar models of trust.

The use of the HITS algorithm has several benefits over asking the user via direct questioning. For example, the HITS algorithm provides a set of weightings that can not only be used to rank trust concepts but they can also be used to determine the relative weightings between the ranked trust concepts. These relative weightings could be used to develop a trust calculation that makes use of more, or less, trust concepts than the top three trust concepts used to date. This would provide an even greater level of calculation personalisation, which may provide more accurate recommendations. However, this personalisation mechanism requires additional user input, which is not found in non-personalised models of trust. Yet, once the process is completed a user gains all the benefits of a personalised approach.

Case Study

Objective five required that two domain specific models of trust were engineered in order to illustrate and compare specialisation across multiple domains, namely an evidence based domain model (Web Services) and an opinion based domain model (Instant Messaging). These two domain models provided the case studies that illustrate the requirement for specialisation within trust, while also allowing a comparison to be made between both domains.

The Web Services domain has a rich set of classes, properties, and relationships in comparison to the Instant Messaging domain. The Web Services domain has 38 classes, 8 object type properties, 46 datatype properties, and 28 restrictions. The Instant Messaging domain is relatively simple when compared with the Web Services domain. The Instant Messaging domain has 12 classes, 4 object properties, 5 datatype properties, and no restrictions.

The design and development of these two domain models is sufficient to demonstrate that different application domain models can be very different in terms of complexity, classes, properties, and relationships. In addition, these two domain models also demonstrate that the multi-faceted model of trust can support specialisation. However, further specialisation has not been carried out across a wider range of application domains. Further specialisation could be used to investigate how representative the current two specialisations are of the wider range of application domains, and to further illustrate the requirement for domain specialisation.

Dynamic and Flexible Management

The implemented trust management service, *myTrust*, was used to address objective six. The Community Based Policy Management (CBPM) system provides a flexible management solution for modelling evolving organisational structures. Independently, *myTrust* can capture the dynamic evolution of trust relationships. The combination of these separate trust and policy systems was used to regulate access control using two distinct projects. Initially, the combination of *myTrust* and CBPM provided access control over doors in the PUDECAS ubiquitous computing simulator. Subsequently, the combination provided access control over location information in an enhanced Instant Messaging (IM) system. The success of these trials illustrates the diversity of

potential applicability of the CBPM and *myTrust* combination. The author is confident that the combination of CBPM and *myTrust* could provide a flexible and dynamic trust management service to a wider range of application and real world scenarios. For example, if people in the real world Lloyd building carried mobile devices that conveyed their location then it would be possible to provide the same service to the real world person as was provided to the virtual user. The real world applications are limitless and could include the provision of a trust management service to football clubs, which tend to be community based and regularly monitored for security reasons.

It is important to note that to date the CBPM system and *myTrust* service have performed satisfactorily in the two trials that provided access control for a small set of trial users within the PUDECAS ubiquitous computing simulator and enhanced IM application. However, in both trials, the trust annotation data remained static for the duration of each trial. In addition, the outcomes of requests were known in advance, and policies did not change. It would be beneficial to conduct a set of trials that take place in a truly dynamic environment where trust relationships and policies change in real-time. This would further illustrate the provision of flexible and dynamic trust management in operation, and could also be used to investigate the CBPM system's and *myTrust* service's performance, operational capacity, throughput, latency, and so on.

7.2 Contribution

The first contribution of this research is the novel strategy in which modelling trust is accomplished through a multi-faceted approach that is personalisable and specialisable. To this author's knowledge such an approach to modelling trust has not been published before and is not found in the state of the art. This is an important contribution as such a model can provide existing, and future, trust management systems with a model of trust that is applicable to a broad population, yet is also personalisable to each individual. Thus, the consensus of trust becomes one of everyone having their own opinion, or individual subjectivity across a large and broad population with wide and diverse views of trust.

In the state of the art the majority of trust management systems that use a single-faceted approach make use of the trust concept reputation. However, trust data that reflects reputation is not easily used by a trust management system that uses a different trust concept such as reliability or credibility. The multi-faceted model of trust that is implemented in OWL can handle such differences and is therefore potentially more interoperable across different trust management systems. This could be of great benefit to well established eBay members who wish to carry over their status to other auction sites, social networks, and so on. However, it must be noted that there needs to be further investigation into semantic interoperability issues that may exist between specialised domain models. This may not be straight forward due to inherent differences found across a variety of specialised models. Therefore, future work is required to address this challenge.

As well as being useful for providing trust management for traditional Internet applications such as e-commerce, Instant Messaging, and so on, the multi-faceted model of trust could be particularly useful for social networks where individual models will need to be shared and reasoned about in order to provide a trust management service. Moreover, the idea of consensus is very useful for social network websites such as Bebo and MySpace as they could take advantage of the multi-faceted model of trust proposed in this thesis to provide personalised trust services, across different domains, to their very large number of members that have developed and formed relationships. It has been shown in this thesis that a single-faceted approach cannot capture the subjective views of trust for a large and broad population, whereas a multi-faceted, personalised model of trust can. Therefore, it is argued that the multi-faceted, personalisable model of trust proposed in this thesis is more appropriate and useful in social networks than a single-faceted approach.

There are also important contributions to be considered in relation to the related work. The model of trust presented in this thesis is multi-faceted, personalisable, and is both developer and user specialisable. Such a model of trust is not found in, or used by, any other trust management system in the reviewed state of the art. Empowering the user with the ability to design and develop specialised models of trust creates an environment for community participation, and community verification, of specialised trust models. This could impact on the uptake and use of domain specific models by

communities in Internet environments. A smaller proportion of the reviewed trust management systems use policy. REFEREE, SULTAN, OpenPrivacy, Fidelis, and Slashdot all provide policy that is internal to the trust management system, which is a feature that *myTrust* also provides. However, by providing *myTrust* services to the CBPM system it is possible to offer a level of flexibility and dynamicity for management functionality that is unparalleled in the reviewed state of the art trust management systems. The impact of such a system is that trust management can be provided on a personalised basis, across multiple application domains, to every member of ever evolving communities. *myTrust* uses a distributed architecture, which has the benefits of (i) a larger available population to search, and (ii) path(s) may exist from the requestor of that trust data to the provider of that trust data. The impact of benefits can mean the larger population would increase the possibility of retrieving trust data and so it could be possible to provide recommendations where it was once not possible. In addition, a path between requestor and provider of trust data may produce trust values that mean more to the user who makes a decision as they may feel that a path strengthens or weakens a recommendation (weakens if the provider is negatively viewed). With regards trust representation TRELIS, FOAF extended, and FilmTrust all use an OWL based approach. OWL is also used as the trust representation format for *myTrust*. Some of the benefits of using an OWL approach include extendibility, reusability, and the easy sharing and understanding of personalised and domain specific models of trust, and also trust data.

The overall analysis of *myTrust* that was conducted across all reviewed trust management systems illustrates that *myTrust* at least matches, or exceeds, all other trust management systems in relation to provisions made for (i) model of trust, (ii) trust annotation, and (iii) trust calculation categories.

A second contribution arose from the experiment and analysis of the accuracy of trust based recommendations. The first experiment illustrated the necessity for a multi-faceted, personalised model of trust. This is a significant contribution as currently, to this author's knowledge there is no other similar evaluation with regards trust concepts. This knowledge is very useful to trust management research as it may help, or accelerate, the adoption of multi-faceted approaches to modelling trust, which will enable the trust model to gain the benefits of a multi-faceted approach. It has also

been shown through experiment that a higher overall level of accuracy can be achieved through the provision of additional information. Additionally, in this experiment, and in circumstances of higher risk, the provision of additional information can result in higher levels of accuracy in comparison to accuracy levels when no additional information was available. This conclusion can aid researchers and developers to decide where and when additional information should be provided in order to achieve higher levels of accuracy. Conversely, the conclusion can also be used to decide where and when additional information should not be provided, which have been the very low risk areas in the experiment to date. However, it must be noted that such increases and decreases may not be indicative of all types of additional information. Finally, clarity experiments show that trust concepts categorised as *concrete* are perceived to have stronger levels of clarity than *abstract* trust concepts. This conclusion can aid researchers and developers in their choice of trust concept for use in a single-faceted model of trust, which could increase the usefulness of that particular trust model to its users.

Peer Reviewed Publications

The most comprehensive publication to date, of the multi-faceted model of trust and its evaluation, was published at a major international ERCIM trust management conference called iTrust:

Quinn, K., O'Sullivan, D., Lewis, D., Wade, V.P., 'The Design, Generation, and Utilisation of a Semantically Rich Personalised Model of Trust', 4th International Conference on Trust Management (iTrust 2006), Pisa, Italy, 15-19 May, 2006.

It is planned to submit a more detailed description of the evaluation presented at iTrust 2006, along with the subsequent accuracy and clarity experimentation and analysis, to selected journals including the International Journal of Information Security (Springer) and the Journal of Computer Security (IOS).

Collaboration with Ericsson Research Group, Ireland resulted in the publication of a paper that described the Web Services specific domain model and a use case that accompanied it. This publication appeared at a major international IEEE network management conference called Integrated Management:

Quinn, K., O'Sullivan, D., Lewis, D., Wade, V.P., 'deepTrust Management Application for Discovery, Selection, and Composition of Trustworthy Services', 9th IFIP/IEEE International Symposium on Integrated Network Management (IM 2005), Nice, France, 15-19 May, 2005.

The *myTrust* service has been used in collaboration with three different research projects to provide trust based recommendations in two use case scenarios. In the M-Zones project *myTrust* services have been provided to the CBPM system in order to provide a dynamic and flexible management solution.

The research work that combines *myTrust* and CBPM has been published over several years at the international IEEE Policies for Distributed Systems and Networks conference:

Feeney, K., Quinn, K., O'Sullivan, D., Lewis, D., Wade, V.P., 'Relationship-Driven Policy Engineering for Autonomic Organisations', IEEE 6th International Workshop on Policies for Distributed Systems and Networks (POLICY 2005), Stockholm, Sweden, 6-8 June, 2005.

Quinn, K., O'Sullivan, D., Lewis, D., Wade, V.P., 'Trust Meta-Policies for Flexible and Dynamic Policy Based Trust Management', 7th International Workshop on Policy (POLICY 2006), London, Ontario, Canada, 5-7 June, 2006.

The research work that combines *myTrust*, CBPM, and the enhanced IM application has been presented at the major international IEEE/IFIP conference for network operation and management:

Quinn, K., Kenny, A., Feeney, K., O'Sullivan, D., Lewis, D., Wade, V.P., 'A Framework for the Decentralisation and Management of Collaborative Applications in Ubiquitous Computing Environments ', 10th IEEE/IFIP Network Operations and Management Symposium (NOMS 2006), Vancouver, Canada, 3-7 April, 2006.

7.3 Future Work

Arising out of the research conducted for this thesis is a number of possibilities for future research and development. Possible future work in relation to the multi-faceted model of trust and the trust management service, *myTrust*, includes:

1. Develop additional enhancements for *myTrust* calculation algorithms,
2. Develop automatic trust annotation mechanisms,
3. Design, develop, and deploy further experiments and trials,
4. Extend the range of application domains for specialisation,
5. Offer a trust management service to emerging online social network paradigms, where no such service currently exists.

These five suggestions are discussed further in the following discussion.

A trust calculation algorithm that is resistant to attacks, such as the Sybil attack [Douceur, 2002], and is applicable to the model of trust proposed in this thesis could be implemented and evaluated. Such an algorithm could enhance the trust management service by providing accurate recommendations even with the retrieval of malicious trust data. The use of weightings that the HITS algorithm produces could also be incorporated into the trust calculation algorithms in order to select the number of trust concepts for use in the calculation. This would provide an alternative to choosing the top three trust concepts, which may provide a further level of personalisation within the model of trust that is embedded into the trust calculation. A trust calculation algorithm that uses trust data that has been retrieved from a social

network might increase the accuracy of trust recommendations, if a path that exists from the requestor of trust data to the provider of trust data. The future work required to test this hypothesis would include some additional implementation and further experiments for evaluation.

In addition to creating a more advanced calculation algorithm it would be beneficial to develop mechanisms that automatically gather trust data in complex, evidence based, application domains. In the Web Services application domain the automated measurement of a services mean time between failure, downtime, and so on would reduce the responsibility on the user to collect trust data, which could lead to a more useful trust management service. Experiments to evaluate what the acceptable burden is for user based trust annotation could provide very useful information for engineering domain specific models of trust. If an acceptable burden was quite high then the domain model could be rich and complex, yet a low level of acceptance may require a simple domain model. In addition, experiments to study a wider range of domain models may reveal further strengths, weaknesses, and limitations of domain specialisation.

As outlined earlier it would be beneficial to develop and deploy a set of experiments that are similar to experiment one but carried out by a larger set of test subjects that are older than 40 years of age. In addition, experiments that evaluate the accuracy of recommendations that use new sources of additional information, such as legal guarantees and penalties, would be interesting and possibly quite beneficial. The development of a trial, or a set of trials, that use the CBPM system and *myTrust* service in a dynamic environment would be beneficial as they could illustrate the truly dynamic operation of the combined system and could also provide valuable performance information. In addition, it would be useful to empirically evaluate the computational and administrative overheads of a personalised approach, which could be compared with other non-personalised approaches.

It would be beneficial to extend the range of application domains for specialisation in order to further investigate the difference, and similarities, found across different application domains. This would further strengthen the argument for specialisation. In addition, further specialisation would also enable an investigation into how

representative the current two domain specialisations are with respect to the wider range of application domains available. The availability of a wider range of specialised models could in turn lead to semantic interoperability experiments, which could evaluate the levels of difficulty, or simplicity, concerning interoperability between multiple application domains.

Further integration of *myTrust* into emerging online social network paradigms is also possible. Currently, large online social networks such as MySpace and Bebo do not offer a trust management mechanism to their respective community members. A trust management service for such social networks would enable its users to annotate each other with trust data so that access to specific areas of an individual's personal website could be regulated based on trust in a manner similar to that of the enhanced Instant Messaging use case scenario. In addition, social networks have very large numbers of members and supplying an excellent trust management service would require a model of trust that can capture the individual subjectivity found in trust within this very large number of members.

7.4 Final Remarks

It is the opinion of the author of this thesis that the state of the art in trust management is dependent on a single-faceted approach. There is a need for a model of trust that can capture the individual's view of trust, which this work shows possesses great diversity across a population. The model of trust proposed in this thesis is multi-faceted and is personalisable and specialisable. This model can accomplish what a single-faceted approach cannot; namely capture both the wide and diverse range of views of trust that exist across a large and broad population as well as the subjectivity found in trust at the individual level.

The author believes that using an ontological approach to representing trust models, data, and values will become the norm over time. This is primarily due to the (i) rich set of properties, classes, and relationships that can be developed using an ontology language, and (ii) level of sharing, reasoning, and interoperability provided by ontology languages.

It is the opinion of the author that the provision of additional information is key to providing trust management. At high levels of risk it is simply not enough to ask a user to rely solely on a trust value calculated from distributed trust data, especially when additional information is available. For example, a member of eBay could quite happily buy an item for \$10 based on the positive feedback from about forty eBay members, but this would not necessarily be the case for an item costing \$1000. For the \$1000 item it would not be unreasonable to assume that the buyer would like to receive documentation or escrow from the seller in an attempt to gauge the validity of the seller. These are the kind of mechanisms that the real world uses to build trust and the online world must reflect this in order to be useful to, and satisfy, the user.

The vision for trust management, in this author's opinion, is that a multi-faceted, personalisable approach must be adopted in order to enable the consensus of trust to become one of everyone having their own opinion, or individual subjectivity across a large and broad population with wide and diverse views of trust. In addition, a model of trust must have the ability to be specialised not only by the developer of a trust management system but also by the end user. It is this level of specialisation that will lead to a much greater uptake and use of specialised trust models, which will in turn lead to more useful trust management systems.

The author's vision for trust management systems is reflected in *myTrust*. A trust management system must be able to work with a large number of individuals (via their personalised models) in order to calculate trust values for multiple application domains (via domain specific models of trust). In this way it is possible for trust management systems to move from providing a 'one size fits all' approach in one application domain to a 'one model for all' approach across multiple application domains.

Finally, it is the opinion of the author that trust will never become the norm, or common place, in Internet environments for the wider range of applications such as Instant Messaging and email unless a generic platform with universal appeal exists to reflect the multi-faceted, subjective, and domain specific nature of trust.

BIBLIOGRAPHY

[Adams et al, 1999] Adams C., and Farrell S., ‘RFC2510 - Internet X.509 Public Key Infrastructure Certificate Management Protocols’, 1999, <http://www.cis.ohio-state.edu/htbin/rfc/rfc2510.html>

[Abdul-Rahman et al, 2000] Abdul-Rahman, A., Hailes, S., ‘Supporting Trust in Virtual Communities’, In Proceedings Hawaii International Conference on System Sciences 33, Maui, Hawaii, 4-7 January 2000.

[Abdul-Rahman, 2005] Abdul-Rahman, A., ‘A Framework for Decentralised Trust Reasoning’, PhD doctoral thesis, Department of Computer Science, University of London, 2005.

[Amazon] Amazon website, <http://www.amazon.com>

[Amazon Auctions] Amazon Auctions website, <http://auctions.amazon.com>

[Amazon Marketplace] Amazon Marketplace website, <http://www.amazon.com>

[Amazon zShops] Amazon zShops website, <http://zshops.amazon.com>

[Barber, 1983] Barber, Bernard. 1983. Logic and Limits of Trust. New Jersey: Rutgers University Press.

[Bebo] Bebo website, <http://www.bebo.com>

[Blaze et al, 1996] M. Blaze, J. Feigenbaum, and J. Lacy. Decentralized Trust Management. In Proc. of the 17th Symposium on Security and Privacy, pages 164-173. IEEE Computer Society Press, Los Alamitos, 1996.

[Blaze et al, 1998a] M. Blaze, J. Feigenbaum, and M. Strauss. Compliance Checking in the PolicyMaker Trust-Management System. In Proc. of the Financial Cryptography '98, Lecture Notes in Computer Science, vol.1465, pages 254-274. Springer, Berlin, 1998.

[Blaze et al, 1998b] M. Blaze, J. Feigenbaum, J. Ioannidis, and A. Keromytis. The KeyNote Trust-Management System. Work in Progress, <http://www.cis.upenn.edu/~angelos/keynote.html> , June 1998.

[Chen et al, 2000] Chen, R., W. Yeager, 'Poblano: A Distributed Trust Model for Peer-to-PeerNetworks', Sun Microsystems, 2000, <http://www.ovmj.org/GNUnet/papers/jxtatrust.pdf>

[Chu et al, 1997] Chu, Y., Feigenbaum, J., LaMacchia, B., Resnick, P., and Strauss, Ma., 'REFEREE: Trust Management for Web Applications.', The World Wide Web Journal, 1997, 2(3), pp. 127-139.

[Corritore et al, 2001] Corritore, C., B. Kracher, and S. Wiedenbeck, 'An Overview of Trust', CHI Workshop Position Paper, 2001. http://cobacourses.creighton.edu/trust/articles/trustpaper2-9-01_final.rtf

[Damianou et al, 2001] Damianou, N., Dulay, N., Lupu, E., Sloman, M., 'The Ponder Specification Language', Workshop on Policies for Distributed Systems and Networks (Policy2001), HP Labs Bristol, 29-31 Jan 2001.

[Deutsch, 1986] Deutsch, M., 'Cooperation and Trust. Some Theoretical Notes', in Jones, M.R. (ed) Nebraska Symposium on Motivation, Nebraska University Press.

[Douceur, 2002] Douceur, J.R., 'The Sybil Attack', In Proceedings of the IPTPS02 Workshop, Cambridge, MA (USA), March 2002.

[Dumbill et al, 2002] Dumbill, E., 'XML Watch: Finding friends with XML and RDF.', IBM Developer Works', <http://www-106.ibm.com/developerworks/xml/library/xfoaf.html>, June 2002.

[eBay] eBay website, <http://www.ebay.com>

[Eclipse] Eclipse IDE website, <http://www.eclipse.org>

[Epinions] Epinions website, <http://www.epinions.com>

[Feeney et al, 2004] Feeney, K., Lewis, D., Wade, V. "Policy-based Management for Internet Communities", in proc of 5th IEEE International Workshop on Policies and Distributed Systems and Networks, IEEE, 2004

[Feeney et al, 2005] Feeney, K., Quinn, K., O'Sullivan, D., Lewis, D., Wade, V.P., 'Relationship-Driven Policy Engineering for Autonomic Organisations', IEEE 6th International Workshop on Policies for Distributed Systems and Networks (POLICY 2005), Stockholm, Sweden, 6-8 June, 2005 .

[Ferraiolo et al, 1992] Ferraiolo, D.F., and Kuhn, D.R., 'Role Based Access Control', 15th National Computer Security Conference (1992).

[Gambetta, 1988] Diego Gambetta, editor. Trust: making and breaking cooperative relations, <http://www.sociology.ox.ac.uk/papers/biographies.html> - dgpp. 158-175, Basil Blackwell, 1988.

[Gerck, 97] Gerck, E., 'Toward Real-World Models of Trust: Reliance on Received Information', <http://www.safevote.com/papers/trustdef.htm>

[Gil et al, 2002] Gil, Y., Ratnakar, V., 'Trusting Information Sources One Citizen at a Time.', Proceedings of the First International Semantic Web Conference (ISWC), Sardinia, Italy, June 2002.

[Golbeck et al, 2003] Golbeck, J., Hendler, J., Parsia, B. 'Trust Networks on the Semantic Web', 12th International Web Conference (WWW03), Budapest, Hungary, May 2003.

[Golbeck et al, 2004] Golbeck, Jennifer, James Hendler, "Accuracy of Metrics for Inferring Trust and Reputation" in Proceedings of 14th International Conference on Knowledge Engineering and Knowledge Management, October 5-8, 2004, Northamptonshire, UK.

[Golbeck & Hendler, 2004] Golbeck, J., Hendler, J., 'Inferring Reputation on the Semantic Web', 13th International Web Conference (WWW2004), New York, NY, USA, May 2004.

[Golbeck, 2005] Golbeck, J., 'Computing and Applying Trust in Web-based Social Networks', PhD thesis, Department of Computer Science, University of Maryland, 2005. <http://trust.mindswap.org/papers/GolbeckDissertation.pdf>

[Golbeck et al, 2006] Golbeck, J., Hendler, J., '2006. FilmTrust: Movie recommendations using trust in web-based social networks', *Proceedings of the IEEE Consumer Communications and Networking Conference*, January 2006.

[Golembiewski et al, 1975] Golembiewski, R.T., and McConkie, M., 'The Centrality of Interpersonal Trust in Group Processes', in *Theories of Group Processes* (Cary Cooper (ed). Hoboken, NJ: Wiley.

[Gollmann, 2005] Gollmann, D., 'Why Trust is Bad for Security', Keynote Speech, IEEE 6th International Workshop on Policies for Distributed Systems and Networks (POLICY 2005), Stockholm, Sweden, 6th-8th June, 2005.

[Grandison & Sloman, 2000] Grandison, T., Sloman, M., 'A Survey of Trust in Internet Applications', *IEEE Communications Surveys*, 3, pp. 2-16, Fourth Quarter 2000.

[Grandison et al, 2001] Grandison, T., Sloman, M., ‘SULTAN - A Language for Trust Specification and Analysis’, Proceedings of the 8th Annual Workshop HP OpenView University Association (HP-OVUA), Berlin, Germany, June 24-27, 2001.

[Grandison et al, 2003] Grandison, T., and Sloman, M., ‘Trust Management Tools for Internet Applications’, proceedings of the 1st International Conference on Trust Management (iTrust 2003), Crete, Greece. May 28-30, 2003.

[Grandison, 2003] Grandison., T., ‘Trust Management for Internet Applications’, Ph.D. thesis, Imperial College of Science, Technology and Medicine, University of London, Department of Computing, 2003.

http://www.doc.ic.ac.uk/~tgrand/PhD_Thesis.pdf

[Hart, 1988] Diego Gambetta, editor. Trust: making and breaking cooperative relations, <http://www.sociology.ox.ac.uk/papers/biographies.html> - khpp. 176-193, Basil Blackwell, 1988.

[Herzberg et al, 2000] Herzberg, A., Mass, Y., Mihaeli, J., Naor, D., and Ravid, Y., ‘Access Control Meets Public Key Infrastructure, or: Assigning Roles to Strangers’, in proceedings IEEE Symposium on Security and Privacy, 2000.

[JBoss] JBoss J2EE Application Server, 2006, <http://www.jboss.org>.

[Jena, HP] Hewlett Packard, <http://www.hpl.hp.com/semweb/rdql.htm>.

[Jonker et al, 2004] Jonker, C.M., Treur, J., ‘Human Experiments in Trust Dynamics’, Proceedings of Second International Trust Management Conference, iTrust 2004, Oxford, UK, March 29 - April 1, 2004.

[JXTA] JXTA Project, <http://www.jxta.org/>

[Kagal et al, 2002] Kagal, L., Undercoffer, J., Perich, F., Joshi, A, Finin, T., ‘A Security Architecture Based on Trust Management for Pervasive Computing Systems’, Proceedings of Grace Hopper Celebration of Women in Computing 2002.

[Kamvar et al, 2003] Kamvar, S.D., Schlosser, M.T., and Garcia-Molina, H., 'The EigenTrust Algorithm for Reputation Management in P2P Networks', In Proceedings International WWW Conference, Budapest, Hungary, 2003.

[Kenny et al, 2006] Kenny, A., Lewis, D., O'Sullivan, D., 'Interlocutor: Decentralised Infrastructure for Adaptive Interaction', 3rd International Workshop on Managing Ubiquitous (MUCS2006), Cork, Ireland, May 4-5, 2006.

[Kleinberg, 1998] J. Kleinberg. Authoritative sources in a hyperlinked environment. Proceedings of the 9th ACM-SIAM Symposium on Discrete Algorithms, 1998.

[Labalme & Burton, 2001] Labalme, F., and Burton, K., 'Enhancing the Internet with Reputations. An OpenPrivacy White Paper', March 2001. <http://www.openprivacy.org/papers/200103-white.html>.

[Lamsal, 2001] Lamsal, P., Understanding Trust and Security. 2001, <http://citeseer.ist.psu.edu/lamsal01understanding.html>

[Levien et al, 1998] Levien, R., Aiken, A., 'Attack resistant trust metrics for public key certification.', 7th USENIX Security Symposium, San Antonio, Texas, January 1998.

[Luhmann, 1979] Luhmann, N., 'Trust and Power', Wiley, Chichester, 1979.

[Marsh, 1994] Marsh, S.: Formalising Trust as a Computational Concept. PhD thesis, University of Stirling, Department of Computer Science and Mathematics (1994)

[McKnight & Chervany, 1996] McKnight, H.D., Chervany, N.L., 'The Meanings of Trust; Technical Report 94-04, Carlson School of Management, University of Minnesota', 1996.

[McGuinness & van Harmelen, 2003] McGuinness, D.L., van Harmelen, F., 'OWL Web Ontology Language Overview', W3C Proposed Recommendation, 15th Dec 2003.

[Misztal, 1996] Misztal, B.A., 'Trust in Modern Societies: The Search for the Bases of Social Order', Cambridge: Polity, 1996.

[Musen et al, 1993] Musen, M. A. , Tu, S. W. , Eriksson, H., Gennari, J. H. , and Puerta, A. R., 'PROTEGE-II: An Environment for Reusable Problem-Solving Methods and Domain Ontologies', International Joint Conference on Artificial Intelligence, Chambery, Savoie, France, . 1993.

[MySpace] MySpace website, <http://www.myspace.com>

[MySQL] MySQL 5.0 Reference Manual, August 2006.
<http://dev.mysql.com/doc/refman/5.0/en/index.html>.

[O'Neill et al, 2005] O'Neill, E., Klepal, M., Lewis, D., O'Donnell, T., O'Sullivan, D., Pesch, D. 'A Testbed for Evaluating Human Interaction with Ubiquitous Computing Environments', Proceedings of IEEE 1st International Conference on Testbeds and Research Infrastructures for the DEvelopment of NeTworks and COMmunities (Tridentcom 2005), Trento, Italy, February 21st - 25th, 2005.

[O'Neill et al, 2006] O'Neill, E., Lewis, D., McGlinn, K., Dobson, S., 'Rapid User-Centred Evaluation for Context-Aware Systems', Proceedings XIII International Workshop on Design, Specification and Verification of Interactive Systems (DSV-IS 2006), Dublin (Ireland), July 26-28, 2006

[OWL-DL] Bechhofer, S., van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D.L., Patel-Schneider, P.F., and Stein, L.A., 'OWL web ontology language reference', <http://www.w3.org/TR/owl-ref/>, 2004.

[Quinn et al, 2004] Quinn, K., O'Sullivan, D., Lewis, Wade, V.P., 'Composition of Trustworthy Web Services', Information Technology & Telecommunications Conference (IT&T 2004), Limerick, Ireland, October 2004.

[Quinn et al, 2005] Quinn, K., O'Sullivan, D., Lewis, D., Wade, V.P., 'deepTrust Management Application for Discovery, Selection, and Composition of Trustworthy Services', 9th IFIP/IEEE International Symposium on Integrated Network Management (IM 2005), Nice, France, 15-19 May, 2005.

[Quinn et al, 2006a] Quinn, K., Kenny, A., Feeney, K., O'Sullivan, D., Lewis, D., Wade, V.P., 'A Framework for the Decentralisation and Management of Collaborative Applications in Ubiquitous Computing Environments ', 10th IEEE/IFIP Network Operations and Management Symposium (NOMS 2006), Vancouver, Canada, 3-7 April, 2006.

[Quinn et al, 2006b] Quinn, K., O'Sullivan, D., Lewis, D., Wade, V.P., 'The Design, Generation, and Utilisation of a Semantically Rich Personalised Model of Trust', 4th International Conference on Trust Management (iTrust 2006), Pisa, Italy, 15-19 May, 2006.

[Quinn et al, 2006c] Quinn, K., O'Sullivan, D., Lewis, D., Wade, V.P., 'Trust Meta-Policies for Flexible and Dynamic Policy Based Trust Management', 7th International Workshop on Policy (POLICY 2006), London, Ontario, Canada, 5-7 June, 2006.

[RDQL, HP] Hewlett Packard, <http://www.hpl.hp.com/semweb/jena.htm>.

[Saint-Andre, 2004] Saint-Andre, P., 'Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence', IETF RFC 3921, The Internet Society, October 2004.

[Segall et al, 2000] Segall, B., Arnold, D., Boot, J., Henderson, M., and Phelps, T., 'Content Based Routing with Elvin4', CRC for Enterprise Distributed Systems Technology (DSTC), The University of Queensland, St Lucia, 4072 Australia, 2000.

[Shadbolt, 2002] Shadbolt, N., 'A Matter of Trust', IEEE Intelligent Systems, pp. 2-3 January/February 2002.

[Shapiro, 1987] Shapiro, Susan P., ‘The Social Control of Impersonal Trust’, *The American Journal of Sociology*, 93(3), 623–658, 1987.

[Sierra] Sierra website, <http://sierra.openprivacy.org>.

[Slashdot] Slashdot website, <http://www.slashdot.com>

[SWT] Standard Widget Toolkit website, <http://www.eclipse.org/swt/>

[W3C] World Wide Web Consortium website, <http://www.w3c.org>

[Winsborough et al, 2002] Winsborough, W., and Li, N., ‘Towards Practical Automated Trust Negotiation’, *Third International Workshop on Policies for Distributed Systems and Networks (Policy 2002)*, Monterrey, CA, June 2002.

[Winslett et al, 2002] Winslett, M., Yu, T., Seamons, K. E., Hess, A., Jacobson, J., Jarvis, R., Smith, B., and Yu, L., ‘Negotiating Trust on the Web’, *IEEE Internet Computing*, Vol. 6, No. 6, November/December 2002.

[WLM] Microsoft’s Windows Live Messenger website, <http://get.live.com/messenger/overview>, 2006.

[X509] Internet X509 Public Key Infrastructure documents, <http://www.ietf.org/ids.by.wg/pkix.html>

[Yao, 2002] Yao, W.T.M., ‘Trust Management for Widely Distributed Systems’, PhD thesis, Computer Laboratory, University of Cambridge, 2002.

[Yao, 2003] Yao, W., ‘*Fidelis*: A Policy-Driven Trust Management Framework’, *First International Conference on Trust Management (iTrust 2003)*, Heraklion, Crete, Greece, May 28-30, 2002.

[Zimmerman, 1995] Zimmerman, P.R., ‘The Official PGP Users Guide’, MIT Press, Cambridge, MA, USA, 1995.

GLOSSARY

Upper Ontology	The upper ontology provides a generic set of trust concepts that can be reused to generate personalised models and domain specific models of trust. Within the upper ontology there are two types of trust concepts; <i>concrete</i> concepts and <i>abstract</i> concepts.
Concrete Trust Concept	The <i>concrete</i> concepts are considered to be more defined and tightly scoped than <i>abstract</i> concepts. The <i>concrete</i> concepts are <i>competency</i> , <i>credibility</i> , <i>honesty</i> , <i>reputation</i> , and <i>reliability</i> .
Abstract Trust Concept	The <i>abstract</i> concepts are considered to be more open to interpretation and loosely scoped than the more defined <i>concrete</i> concepts. The <i>abstract</i> concepts are <i>belief</i> , <i>confidence</i> , and <i>faith</i> .
Trust Meta-Model	The trust meta-model provides a set of relationships that are used to relate trust concepts found in the upper ontology. The relationships are <i>derivedFrom</i> , <i>informedBy</i> , and <i>affectedBy</i> . When the upper ontology and meta-model are used together it is possible to generate personalised models of trust or domain specific model of trust.
Personalised Model of Trust	A personalised model of trust is generated from the upper ontology and trust meta-model to capture the individual's subjective view of trust.
Specialised Model of Trust	A specialised model of trust is engineered to capture classes, properties, and relationships that are found in domains such as Web Services or Instant Messaging.
Community Based Policy Management (CBPM)	CBPM is a technology which is based upon modelling an organisation as a hierarchy of communities rather than as roles. Resource authorities, delegated down the community hierarchy provide an authority scope for each community – specifying what events the community can author policies about.

Resource Authorities (CBPM)	All managed resources are modelled as authority trees, each of which can be divided into an action tree and a target tree. A resource authority is a triple [R,T,A] where R is the resource model, T is a node on the resource's target tree and A is a node on the resource's action tree. Every delegation and policy is associated with a particular resource authority, which defines the scope of the delegation or policy.
------------------------------------	--

APPENDICES

APPENDIX I – Trust Ontology Documents

All trust ontology document can be found on the DVD media that accompanies this thesis. The OWL document for the upper ontology is filed under ‘Trust Ontology Documents’, ‘Upper Ontology’. The Meta-model, example personalised model, and the two specialised models can be found in their respective folders in the ‘Trust Ontology Documents’ folder.

Appendix I contains the OWL documents for the upper ontology, meta-model, a personalised model, and two domain specific models of trust; Web Services and Instant Messaging. These OWL documents have been included in this appendix in order to present to the reader with a comprehensive view of the developed models.

Upper Ontology

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperOntology.owl#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xml:base="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperOntology.owl">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="Reliability">
    <owl:disjointWith>
      <owl:Class rdf:ID="Competency"/>
    </owl:disjointWith>
    <rdfs:subClassOf>
      <owl:Class rdf:ID="ConcreteConcepts"/>
    </rdfs:subClassOf>
    <owl:disjointWith>
      <owl:Class rdf:ID="Credibility"/>
    </owl:disjointWith>
    <owl:disjointWith>
      <owl:Class rdf:ID="Reputation"/>
    </owl:disjointWith>
    <owl:disjointWith>
      <owl:Class rdf:ID="Honesty"/>
    </owl:disjointWith>
  </owl:Class>
  <owl:Class rdf:about="#Reputation">
    <rdfs:subClassOf>
```



```

    <owl:Class rdf:about="#ConcreteConcepts"/>
  </rdfs:subClassOf>
  <owl:disjointWith>
    <owl:Class rdf:about="#Credibility"/>
  </owl:disjointWith>
  <owl:disjointWith>
    <owl:Class rdf:about="#Competency"/>
  </owl:disjointWith>
  <owl:disjointWith>
    <owl:Class rdf:about="#Honesty"/>
  </owl:disjointWith>
  <owl:disjointWith rdf:resource="#Reliability"/>
</owl:Class>
<owl:Class rdf:about="#Credibility">
  <owl:disjointWith rdf:resource="#Reputation"/>
  <owl:disjointWith>
    <owl:Class rdf:about="#Competency"/>
  </owl:disjointWith>
  <owl:disjointWith>
    <owl:Class rdf:about="#Honesty"/>
  </owl:disjointWith>
  <rdfs:subClassOf>
    <owl:Class rdf:about="#ConcreteConcepts"/>
  </rdfs:subClassOf>
  <owl:disjointWith rdf:resource="#Reliability"/>
</owl:Class>
<owl:Class rdf:ID="TrustConcepts"/>
<owl:Class rdf:about="#ConcreteConcepts">
  <owl:disjointWith>
    <owl:Class rdf:ID="AbstractConcepts"/>
  </owl:disjointWith>
  <rdfs:subClassOf rdf:resource="#TrustConcepts"/>
</owl:Class>
<owl:Class rdf:about="#Honesty">
  <owl:disjointWith>
    <owl:Class rdf:about="#Competency"/>
  </owl:disjointWith>
  <owl:disjointWith rdf:resource="#Reliability"/>
  <owl:disjointWith rdf:resource="#Reputation"/>
  <rdfs:subClassOf rdf:resource="#ConcreteConcepts"/>
  <owl:disjointWith rdf:resource="#Credibility"/>
</owl:Class>
<owl:Class rdf:ID="Confidence">
  <owl:disjointWith>
    <owl:Class rdf:ID="Faith"/>
  </owl:disjointWith>
  <owl:disjointWith>
    <owl:Class rdf:ID="Belief"/>
  </owl:disjointWith>
  <rdfs:subClassOf>
    <owl:Class rdf:about="#AbstractConcepts"/>
  </rdfs:subClassOf>

```

```

    </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#Competency">
  <owl:disjointWith rdf:resource="#Reputation"/>
  <owl:disjointWith rdf:resource="#Reliability"/>
  <owl:disjointWith rdf:resource="#Honesty"/>
  <owl:disjointWith rdf:resource="#Credibility"/>
  <rdfs:subClassOf rdf:resource="#ConcreteConcepts"/>
</owl:Class>
<owl:Class rdf:about="#Faith">
  <owl:disjointWith>
    <owl:Class rdf:about="#Belief"/>
  </owl:disjointWith>
  <owl:disjointWith rdf:resource="#Confidence"/>
  <rdfs:subClassOf>
    <owl:Class rdf:about="#AbstractConcepts"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#Belief">
  <owl:disjointWith rdf:resource="#Confidence"/>
  <owl:disjointWith rdf:resource="#Faith"/>
  <rdfs:subClassOf>
    <owl:Class rdf:about="#AbstractConcepts"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:about="#AbstractConcepts">
  <owl:disjointWith rdf:resource="#ConcreteConcepts"/>
  <rdfs:subClassOf rdf:resource="#TrustConcepts"/>
</owl:Class>
<owl:ObjectProperty rdf:ID="informedBy">
  <rdfs:range>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#AbstractConcepts"/>
        <owl:Class rdf:about="#ConcreteConcepts"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:range>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="derivedFrom">
  <rdfs:range rdf:resource="#ConcreteConcepts"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="affectedBy">
  <rdfs:range rdf:resource="#AbstractConcepts"/>
</owl:ObjectProperty>
</rdf:RDF>

```

Meta-Model

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:UpperModel="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:daml="http://www.daml.org/2001/03/daml+oil#"
  xmlns="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/MetaModel.owl#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xml:base="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/MetaModel.owl">
  <owl:Ontology rdf:about="">
    <owl:imports
rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl"/>
  </owl:Ontology>
  <owl:ObjectProperty rdf:ID="hasInformedBy">
    <rdfs:range>
      <owl:Class>
        <owl:unionOf rdf:parseType="Collection">
          <rdf:Description
rdf:about="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#AbstractConce
pts">
            <rdfs:subClassOf>
              <owl:Restriction>
                <owl:allValuesFrom>
                  <rdf:Description
rdf:about="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#ConcreteConce
pts">
                    <rdfs:subClassOf>
                      <owl:Restriction>
                        <owl:onProperty rdf:resource="#hasInformedBy"/>
                        <owl:allValuesFrom
rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#AbstractCo
ncepts"/>
                      </owl:Restriction>
                    </rdfs:subClassOf>
                  </rdf:Description>
                </owl:allValuesFrom>
                <owl:onProperty rdf:resource="#hasInformedBy"/>
              </owl:Restriction>
            </rdfs:subClassOf>
          </rdf:Description>
          <rdf:Description
rdf:about="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#ConcreteConce
pts"/>
        </owl:unionOf>
      </owl:Class>
    </rdfs:range>
```

```

    <rdfs:domain
rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#TrustConce
pts"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="hasDerivedFrom">
    <rdfs:domain
rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#ConcreteCo
ncepts"/>
    <rdfs:range
rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#ConcreteCo
ncepts"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:ID="hasAffectedBy">
    <rdfs:range
rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#AbstractCo
ncepts"/>
    <rdfs:domain
rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#AbstractCo
ncepts"/>
  </owl:ObjectProperty>
</rdf:RDF>

```

Personalised Model

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/PersonalisedModel.owl#"
  xmlns:j.0="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:MetaModel="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/MetaModel.owl#"
  xmlns:daml="http://www.daml.org/2001/03/daml+oil#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xml:base="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/PersonalisedModel.owl">
  <owl:Ontology rdf:about="">
    <owl:imports
rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/MetaModel.owl"/>
  </owl:Ontology>
  <owl:Class rdf:ID="PersonalisedModel">
    <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:ObjectProperty rdf:ID="hasConcept"/>
        </owl:onProperty>
        <owl:allValuesFrom
rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#TrustConce
pts"/>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Person">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:ObjectProperty rdf:ID="hasPersonalisedModel"/>
        </owl:onProperty>
        <owl:allValuesFrom rdf:resource="#PersonalisedModel"/>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
  </owl:Class>
  <owl:ObjectProperty rdf:about="#hasPersonalisedModel">
    <rdfs:domain rdf:resource="#Person"/>
    <rdfs:range
rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#TrustConce
pts"/>
  </owl:ObjectProperty>
  <owl:ObjectProperty rdf:about="#hasConcept">
    <rdfs:domain rdf:resource="#PersonalisedModel"/>
```

```

    <rdfs:range
rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#TrustConce
pts"/>
  </owl:ObjectProperty>
  <owl:DatatypeProperty rdf:ID="hasWeight">
    <rdfs:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
    <rdfs:domain
rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#TrustConce
pts"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="personURI">
    <rdfs:domain rdf:resource="#Person"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="personalisedModelName">
    <rdfs:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:domain rdf:resource="#PersonalisedModel"/>
  </owl:DatatypeProperty>
  <owl:FunctionalProperty rdf:ID="hasRank">
    <rdfs:type rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
    <rdfs:domain
rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#TrustConce
pts"/>
  </owl:FunctionalProperty>
  <j.0:Reliability rdf:ID="Reliability">
    <hasRank rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
>7</hasRank>
    <hasWeight rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
>8.73298</hasWeight>
  </j.0:Reliability>
  <Person rdf:ID="user6_jabber">
    <hasPersonalisedModel>
      <PersonalisedModel rdf:ID="default">
        <hasConcept>
          <j.0:Belief rdf:ID="Belief">
            <MetaModel:hasAffectedBy>
              <j.0:Faith rdf:ID="Faith">
                <hasWeight rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
>14.0895</hasWeight>
                <MetaModel:hasInformedBy>
                  <j.0:Honesty rdf:ID="Honesty">
                    <MetaModel:hasDerivedFrom>
                      <j.0:Reputation rdf:ID="Reputation">
                        <MetaModel:hasInformedBy rdf:resource="#Belief"/>
                        <MetaModel:hasDerivedFrom rdf:resource="#Reputation"/>
                        <hasWeight
rdf:datatype="http://www.w3.org/2001/XMLSchema#float"

```

```

>12.6346</hasWeight>
<MetaModel:hasDerivedFrom>
  <j.0:Credibility rdf:ID="Credibility">
    <MetaModel:hasDerivedFrom>
      <j.0:Confidence rdf:ID="Confidence">
        <MetaModel:hasDerivedFrom rdf:resource="#Credibility"/>
        <hasRank
rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
          >1</hasRank>
        <hasWeight
rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
          >15.1296</hasWeight>
        <MetaModel:hasInformedBy rdf:resource="#Faith"/>
        <MetaModel:hasDerivedFrom rdf:resource="#Honesty"/>
        <MetaModel:hasDerivedFrom rdf:resource="#Confidence"/>
        <MetaModel:hasInformedBy rdf:resource="#Belief"/>
        <MetaModel:hasDerivedFrom rdf:resource="#Reputation"/>
      </j.0:Confidence>
    </MetaModel:hasDerivedFrom>
    <MetaModel:hasInformedBy rdf:resource="#Belief"/>
    <hasWeight
rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
      >14.243</hasWeight>
    <MetaModel:hasDerivedFrom rdf:resource="#Reputation"/>
    <hasRank
rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
      >2</hasRank>
    <MetaModel:hasInformedBy rdf:resource="#Faith"/>
    <MetaModel:hasDerivedFrom rdf:resource="#Credibility"/>
    <MetaModel:hasDerivedFrom rdf:resource="#Honesty"/>
  </j.0:Credibility>
</MetaModel:hasDerivedFrom>
<MetaModel:hasDerivedFrom rdf:resource="#Confidence"/>
<MetaModel:hasInformedBy rdf:resource="#Faith"/>
<hasRank rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
  >5</hasRank>
</j.0:Reputation>
</MetaModel:hasDerivedFrom>
<MetaModel:hasInformedBy rdf:resource="#Faith"/>
<hasWeight rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
  >11.8188</hasWeight>
<MetaModel:hasDerivedFrom rdf:resource="#Honesty"/>
<MetaModel:hasDerivedFrom rdf:resource="#Credibility"/>
<MetaModel:hasDerivedFrom rdf:resource="#Confidence"/>
<hasRank rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
  >6</hasRank>
<MetaModel:hasInformedBy rdf:resource="#Belief"/>
<MetaModel:hasDerivedFrom rdf:resource="#Reliability"/>
</j.0:Honesty>
</MetaModel:hasInformedBy>
<MetaModel:hasAffectedBy rdf:resource="#Faith"/>

```

```

        <MetaModel:hasInformedBy rdf:resource="#Confidence"/>
        <MetaModel:hasAffectedBy rdf:resource="#Belief"/>
        <hasRank rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
        >3</hasRank>
        <MetaModel:hasInformedBy rdf:resource="#Credibility"/>
    </j.0:Faith>
</MetaModel:hasAffectedBy>
<MetaModel:hasInformedBy rdf:resource="#Credibility"/>
<hasRank rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
>4</hasRank>
<hasWeight rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
>13.3808</hasWeight>
<MetaModel:hasAffectedBy rdf:resource="#Belief"/>
<MetaModel:hasInformedBy rdf:resource="#Confidence"/>
</j.0:Belief>
</hasConcept>
<personalisedModelName rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>default</personalisedModelName>
<hasConcept rdf:resource="#Faith"/>
<hasConcept>
    <j.0:Competency rdf:ID="Competency">
        <MetaModel:hasDerivedFrom rdf:resource="#Reliability"/>
        <MetaModel:hasInformedBy rdf:resource="#Faith"/>
        <MetaModel:hasDerivedFrom rdf:resource="#Reputation"/>
        <MetaModel:hasDerivedFrom rdf:resource="#Credibility"/>
        <MetaModel:hasDerivedFrom rdf:resource="#Confidence"/>
        <MetaModel:hasInformedBy rdf:resource="#Belief"/>
        <hasWeight rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
        >8.01482</hasWeight>
        <MetaModel:hasDerivedFrom rdf:resource="#Competency"/>
        <hasRank rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
        >8</hasRank>
    </j.0:Competency>
</hasConcept>
<hasConcept rdf:resource="#Confidence"/>
<hasConcept rdf:resource="#Honesty"/>
<hasConcept rdf:resource="#Credibility"/>
<hasConcept rdf:resource="#Reliability"/>
<hasConcept rdf:resource="#Reputation"/>
</PersonalisedModel>
</hasPersonalisedModel>
<personURI rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>user6@jabber</personURI>
</Person>
</rdf:RDF>

```


Specialised Models

Web Services Domain Specific Model

```
<?xml version='1.0' encoding='ISO-8859-1'?>
<!DOCTYPE uridef[
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns">
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema">
  <!ENTITY owl "http://www.w3.org/2002/07/owl">
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema">
  <!ENTITY service "http://www.daml.org/services/owl-s/0.9/Service.owl">
  <!ENTITY DEFAULT "http://www.cs.tcd.ie/Andrew.Jackson/fire.owl">
]>

<rdf:RDF
  xmlns:rss="http://purl.org/rss/1.0/"
  xmlns:jms="http://jena.hpl.hp.com/2003/08/jms#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:vcard="http://www.w3.org/2001/vcard-rdf/3.0#"
  xmlns:daml="http://www.daml.org/2001/03/daml+oil#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/trust#"
  xml:base="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/trust">
  <owl:Ontology rdf:about="">
  </owl:Ontology>
  <owl:Class rdf:ID="UtmostGoodFaith">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#HonestyElement"/>
    </rdfs:subClassOf>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
      >Good business ethics that can help provide a view of the vendor as
  honest.</rdfs:comment>
  </owl:Class>
  <owl:Class rdf:ID="UserHistory">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#History"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="SLAElement">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:ObjectProperty rdf:about="#isElementOf"/>
        </owl:onProperty>
        <owl:allValuesFrom>
          <owl:Class rdf:about="#Credibility"/>
        </owl:allValuesFrom>
      </owl:Restriction>
```

```

    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Class rdf:about="#TrustElements"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="ConceptualTrust">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#TrustConcepts"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="StandardsElement">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#TrustElements"/>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:allValuesFrom>
          <owl:Class rdf:about="#Competency"/>
        </owl:allValuesFrom>
        <owl:onProperty>
          <owl:ObjectProperty rdf:about="#isElementOf"/>
        </owl:onProperty>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="HonestyElement">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:allValuesFrom>
          <owl:Class rdf:about="#Honesty"/>
        </owl:allValuesFrom>
        <owl:onProperty>
          <owl:ObjectProperty rdf:about="#isElementOf"/>
        </owl:onProperty>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Class rdf:about="#TrustElements"/>
    </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="Honesty">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#ConcreteTrust"/>
    </rdfs:subClassOf>
    <owl:disjointWith>
      <owl:Class rdf:about="#Competency"/>
    </owl:disjointWith>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
      >How 'honest' a vendor/service is in relation to criminality or deceit. Providing
      audit trails, acting in utmost good faith, and providing a service that reflects its

```

description, without deceptive hidden costs, can all help a vendor/service establish a view of honesty among its customers.</rdfs:comment>

```
<owl:disjointWith>
  <owl:Class rdf:about="#Reputation"/>
</owl:disjointWith>
<owl:disjointWith>
  <owl:Class rdf:about="#Credibility"/>
</owl:disjointWith>
<owl:equivalentClass>
  <owl:Restriction>
    <owl:someValuesFrom rdf:resource="#HonestyElement"/>
    <owl:onProperty>
      <owl:ObjectProperty rdf:about="#hasElement"/>
    </owl:onProperty>
  </owl:Restriction>
</owl:equivalentClass>
<owl:disjointWith>
  <owl:Class rdf:about="#Confidence"/>
</owl:disjointWith>
<owl:disjointWith>
  <owl:Class rdf:about="#Belief"/>
</owl:disjointWith>
<owl:disjointWith>
  <owl:Class rdf:about="#Faith"/>
</owl:disjointWith>
<owl:disjointWith>
  <owl:Class rdf:about="#Reliability"/>
</owl:disjointWith>
</owl:Class>
<owl:Class rdf:ID="VendorHistory">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#History"/>
  </rdfs:subClassOf>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >A vendors history in terms of the different catagories of services that it
  provides.</rdfs:comment>
</owl:Class>
<owl:Class rdf:ID="TrustConcepts">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Trust"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Auditing">
  <rdfs:subClassOf rdf:resource="#HonestyElement"/>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Auditing provides both the vendor/service and user with evidence that can supports
  billing, problem, etc, claims. The presence of an auditing system can help
  vendors/services create a view of honesty among its users.</rdfs:comment>
</owl:Class>
<owl:Class rdf:ID="Security">
  <rdfs:subClassOf>
```

```

    <owl:Class rdf:about="#TrustElements"/>
  </rdfs:subClassOf>
<rdfs:subClassOf>
  <owl:Restriction>
    <owl:allValuesFrom>
      <owl:Class rdf:about="#Confidence"/>
    </owl:allValuesFrom>
    <owl:onProperty>
      <owl:ObjectProperty rdf:about="#isElementOf"/>
    </owl:onProperty>
  </owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Faith">
  <owl:disjointWith>
    <owl:Class rdf:about="#Reputation"/>
  </owl:disjointWith>
  <owl:disjointWith rdf:resource="#Honesty"/>
  <owl:equivalentClass>
    <owl:Class rdf:about="#Belief"/>
  </owl:equivalentClass>
  <owl:disjointWith>
    <owl:Class rdf:about="#Reliability"/>
  </owl:disjointWith>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Faith is the same as Belief</rdfs:comment>
  <owl:disjointWith>
    <owl:Class rdf:about="#Competency"/>
  </owl:disjointWith>
  <owl:disjointWith>
    <owl:Class rdf:about="#Credibility"/>
  </owl:disjointWith>
  <owl:disjointWith>
    <owl:Class rdf:about="#Confidence"/>
  </owl:disjointWith>
  <rdfs:subClassOf rdf:resource="#ConceptualTrust"/>
</owl:Class>
<owl:Class rdf:ID="Availability">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#ReliabilityElement"/>
  </rdfs:subClassOf>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >The availability calculation defines the probability that a service will be
available for use.</rdfs:comment>
</owl:Class>
<owl:Class rdf:ID="Reputation">
  <owl:disjointWith>
    <owl:Class rdf:about="#Reliability"/>
  </owl:disjointWith>
  <owl:disjointWith>
    <owl:Class rdf:about="#Competency"/>

```

```

</owl:disjointWith>
<owl:disjointWith>
  <owl:Class rdf:about="#Belief"/>
</owl:disjointWith>
<owl:equivalentClass>
  <owl:Class>
    <owl:intersectionOf rdf:parseType="Collection">
      <owl:Restriction>
        <owl:onProperty>
          <owl:ObjectProperty rdf:about="#basedOn"/>
        </owl:onProperty>
        <owl:someValuesFrom rdf:resource="#Honesty"/>
      </owl:Restriction>
      <owl:Restriction>
        <owl:someValuesFrom>
          <owl:Class rdf:about="#Competency"/>
        </owl:someValuesFrom>
        <owl:onProperty>
          <owl:ObjectProperty rdf:about="#basedOn"/>
        </owl:onProperty>
      </owl:Restriction>
      <owl:Restriction>
        <owl:someValuesFrom>
          <owl:Class rdf:about="#History"/>
        </owl:someValuesFrom>
        <owl:onProperty>
          <owl:ObjectProperty rdf:about="#hasElement"/>
        </owl:onProperty>
      </owl:Restriction>
    </owl:intersectionOf>
  </owl:Class>
</owl:equivalentClass>
<owl:disjointWith>
  <owl:Class rdf:about="#Credibility"/>
</owl:disjointWith>
<rdfs:subClassOf>
  <owl:Class rdf:about="#ConcreteTrust"/>
</rdfs:subClassOf>
<owl:disjointWith>
  <owl:Class rdf:about="#Confidence"/>
</owl:disjointWith>
<rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >A vendors/services reputation is a very important concept within trust. A
reputation is built from a vendors/services history.</rdfs:comment>
  <owl:disjointWith rdf:resource="#Faith"/>
  <owl:disjointWith rdf:resource="#Honesty"/>
</owl:Class>
<owl:Class rdf:ID="ServiceHistory">
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >An individual or composite services history.</rdfs:comment>
  <rdfs:subClassOf>

```

```

    <owl:Class rdf:about="#History"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Service"/>
<owl:Class rdf:ID="Authentication">
  <rdfs:subClassOf rdf:resource="#Security"/>
</owl:Class>
<owl:Class rdf:ID="ServiceLevelAgreement">
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >A Service Level Agreement (SLA) is a 'contract' between a vendor/service and a
    user that establishes precisely what level of performance, security, etc, that will be
    provided by the vendor/service.</rdfs:comment>
  <rdfs:subClassOf rdf:resource="#SLAElement"/>
</owl:Class>
<owl:Class rdf:ID="HonestyDifferential">
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >The honesty differential is a measurement of how different the description of a
    service is to that services terms and conditions. A service description that is too
    good to be believed usually has catches associated with its use. </rdfs:comment>
  <rdfs:subClassOf rdf:resource="#HonestyElement"/>
</owl:Class>
<owl:Class rdf:ID="ReliabilityElement">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#TrustElements"/>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:allValuesFrom>
        <owl:Class rdf:about="#Reliability"/>
      </owl:allValuesFrom>
      <owl:onProperty>
        <owl:ObjectProperty rdf:about="#isElementOf"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="Integrity">
  <rdfs:subClassOf rdf:resource="#Security"/>
</owl:Class>
<owl:Class rdf:ID="TrustElements">
  <rdfs:subClassOf rdf:resource="#Trust"/>
</owl:Class>
<owl:Class rdf:ID="Confidentiality">
  <rdfs:subClassOf rdf:resource="#Security"/>
</owl:Class>
<owl:Class rdf:ID="MsgDelivery">
  <rdfs:subClassOf rdf:resource="#ReliabilityElement"/>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Reliable communications can be asserted by guaranteeing message
    delivering.</rdfs:comment>
</owl:Class>

```

```

<owl:Class rdf:ID="TermsAndConditions">
  <rdfs:subClassOf rdf:resource="#HonestyElement"/>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Used to calculate the honesty differential.</rdfs:comment>
</owl:Class>
<owl:Class rdf:ID="Standards">
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Standards that are used in Web Services.</rdfs:comment>
  <rdfs:subClassOf rdf:resource="#StandardsElement"/>
</owl:Class>
<owl:Class rdf:ID="Performance">
  <rdfs:subClassOf rdf:resource="#ReliabilityElement"/>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >The performance of a service is a measurement of latency, execution time, and
transaction time. Lower values for these properties implies good service
performance.</rdfs:comment>
</owl:Class>
<owl:Class rdf:ID="Confidence">
  <rdfs:subClassOf rdf:resource="#ConceptualTrust"/>
  <owl:disjointWith>
    <owl:Class rdf:about="#Competency"/>
  </owl:disjointWith>
  <owl:disjointWith rdf:resource="#Faith"/>
  <owl:disjointWith>
    <owl:Class rdf:about="#Reliability"/>
  </owl:disjointWith>
  <owl:disjointWith rdf:resource="#Reputation"/>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Security can help create confidence as it mitigates against unauthorized access to
services, service tampering, etc.</rdfs:comment>
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Restriction>
          <owl:onProperty>
            <owl:ObjectProperty rdf:about="#inspiredBy"/>
          </owl:onProperty>
          <owl:someValuesFrom>
            <owl:Class rdf:about="#Competency"/>
          </owl:someValuesFrom>
        </owl:Restriction>
        <owl:Restriction>
          <owl:onProperty>
            <owl:ObjectProperty rdf:about="#inspiredBy"/>
          </owl:onProperty>
          <owl:someValuesFrom>
            <owl:Class rdf:about="#Credibility"/>
          </owl:someValuesFrom>
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>

```

```

        <owl:Class rdf:about="#Reliability"/>
    </owl:someValuesFrom>
    <owl:onProperty>
        <owl:ObjectProperty rdf:about="#assistedBy"/>
    </owl:onProperty>
</owl:Restriction>
<owl:Restriction>
    <owl:onProperty>
        <owl:ObjectProperty rdf:about="#inspiredBy"/>
    </owl:onProperty>
    <owl:someValuesFrom rdf:resource="#Reputation"/>
</owl:Restriction>
<owl:Restriction>
    <owl:someValuesFrom rdf:resource="#Security"/>
    <owl:onProperty>
        <owl:ObjectProperty rdf:about="#hasElement"/>
    </owl:onProperty>
</owl:Restriction>
</owl:intersectionOf>
</owl:Class>
</owl:equivalentClass>
<owl:disjointWith>
    <owl:Class rdf:about="#Belief"/>
</owl:disjointWith>
<owl:disjointWith rdf:resource="#Honesty"/>
<owl:disjointWith>
    <owl:Class rdf:about="#Credibility"/>
</owl:disjointWith>
</owl:Class>
<owl:Class rdf:ID="Reliability">
    <rdfs:subClassOf>
        <owl:Class rdf:about="#ConcreteTrust"/>
    </rdfs:subClassOf>
    <owl:disjointWith>
        <owl:Class rdf:about="#Credibility"/>
    </owl:disjointWith>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >A reliable service will be available, guarantee message delivery, provide good
performance, and carry with it certain assurances.</rdfs:comment>
    <owl:disjointWith>
        <owl:Class rdf:about="#Competency"/>
    </owl:disjointWith>
    <owl:equivalentClass>
        <owl:Class>
            <owl:intersectionOf rdf:parseType="Collection">
                <owl:Restriction>
                    <owl:onProperty>
                        <owl:ObjectProperty rdf:about="#assistedBy"/>
                    </owl:onProperty>
                    <owl:someValuesFrom rdf:resource="#Honesty"/>
                </owl:Restriction>

```



```

    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty rdf:about="#basedOn"/>
      </owl:onProperty>
      <owl:someValuesFrom rdf:resource="#Reputation"/>
    </owl:Restriction>
  </owl:Restriction>
  <owl:Restriction>
    <owl:someValuesFrom rdf:resource="#ReliabilityElement"/>
    <owl:onProperty>
      <owl:ObjectProperty rdf:about="#hasElement"/>
    </owl:onProperty>
  </owl:Restriction>
</owl:intersectionOf>
</owl:Class>
</owl:equivalentClass>
<owl:disjointWith rdf:resource="#Confidence"/>
<owl:disjointWith rdf:resource="#Reputation"/>
<owl:disjointWith>
  <owl:Class rdf:about="#Belief"/>
</owl:disjointWith>
<owl:disjointWith rdf:resource="#Honesty"/>
<owl:disjointWith rdf:resource="#Faith"/>
</owl:Class>
<owl:Class rdf:ID="ConcreteTrust">
  <rdfs:subClassOf rdf:resource="#TrustConcepts"/>
</owl:Class>
<owl:Class rdf:ID="Privacy">
  <rdfs:subClassOf rdf:resource="#Security"/>
</owl:Class>
<owl:Class rdf:ID="History">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:allValuesFrom rdf:resource="#Reputation"/>
      <owl:onProperty>
        <owl:ObjectProperty rdf:about="#isElementOf"/>
      </owl:onProperty>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf rdf:resource="#TrustElements"/>
</owl:Class>
<owl:Class rdf:ID="Competency">
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:someValuesFrom rdf:resource="#StandardsElement"/>
      <owl:onProperty>
        <owl:ObjectProperty rdf:about="#hasElement"/>
      </owl:onProperty>
    </owl:Restriction>
  </owl:equivalentClass>
  <owl:disjointWith rdf:resource="#Honesty"/>
  <owl:disjointWith>

```

```

    <owl:Class rdf:about="#Belief"/>
  </owl:disjointWith>
  <owl:disjointWith>
    <owl:Class rdf:about="#Credibility"/>
  </owl:disjointWith>
  <owl:disjointWith rdf:resource="#Confidence"/>
  <rdfs:subClassOf rdf:resource="#ConcreteTrust"/>
  <owl:disjointWith rdf:resource="#Reputation"/>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Web Service competency can be attained by using standards and producing services
that conform to these standards.</rdfs:comment>
  <owl:disjointWith rdf:resource="#Reliability"/>
  <owl:disjointWith rdf:resource="#Faith"/>
</owl:Class>
<owl:Class rdf:ID="Belief">
  <owl:disjointWith rdf:resource="#Confidence"/>
  <rdfs:subClassOf rdf:resource="#ConceptualTrust"/>
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Restriction>
          <owl:onProperty>
            <owl:ObjectProperty rdf:about="#assistedBy"/>
          </owl:onProperty>
          <owl:someValuesFrom rdf:resource="#Reliability"/>
        </owl:Restriction>
        <owl:Restriction>
          <owl:someValuesFrom rdf:resource="#Confidence"/>
          <owl:onProperty>
            <owl:ObjectProperty rdf:about="#inspiredBy"/>
          </owl:onProperty>
        </owl:Restriction>
        <owl:Restriction>
          <owl:onProperty>
            <owl:ObjectProperty rdf:about="#inspiredBy"/>
          </owl:onProperty>
          <owl:someValuesFrom rdf:resource="#Reputation"/>
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
  <owl:disjointWith rdf:resource="#Competency"/>
  <owl:disjointWith>
    <owl:Class rdf:about="#Credibility"/>
  </owl:disjointWith>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Belief in a service/vendor is an aggregation of many relationships between some
other trust concepts.</rdfs:comment>
  <owl:equivalentClass rdf:resource="#Faith"/>
  <owl:disjointWith rdf:resource="#Honesty"/>
  <owl:disjointWith rdf:resource="#Reliability"/>

```

```

    <owl:disjointWith rdf:resource="#Reputation"/>
  </owl:Class>
  <owl:Class rdf:ID="Credibility">
    <owl:disjointWith rdf:resource="#Reputation"/>
    <owl:disjointWith rdf:resource="#Belief"/>
    <owl:disjointWith rdf:resource="#Faith"/>
    <owl:disjointWith rdf:resource="#Confidence"/>
    <rdfs:subClassOf rdf:resource="#ConcreteTrust"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >A service can be deemed to have a certain credibility if a Service Level Agreement
  supports it.</rdfs:comment>
    <owl:equivalentClass>
      <owl:Class>
        <owl:intersectionOf rdf:parseType="Collection">
          <owl:Restriction>
            <owl:someValuesFrom rdf:resource="#Reputation"/>
            <owl:onProperty>
              <owl:ObjectProperty rdf:about="#basedOn"/>
            </owl:onProperty>
          </owl:Restriction>
          <owl:Restriction>
            <owl:someValuesFrom rdf:resource="#Reliability"/>
            <owl:onProperty>
              <owl:ObjectProperty rdf:about="#basedOn"/>
            </owl:onProperty>
          </owl:Restriction>
          <owl:Restriction>
            <owl:someValuesFrom rdf:resource="#Competency"/>
            <owl:onProperty>
              <owl:ObjectProperty rdf:about="#assistedBy"/>
            </owl:onProperty>
          </owl:Restriction>
          <owl:Restriction>
            <owl:onProperty>
              <owl:ObjectProperty rdf:about="#basedOn"/>
            </owl:onProperty>
            <owl:someValuesFrom rdf:resource="#Honesty"/>
          </owl:Restriction>
          <owl:Restriction>
            <owl:onProperty>
              <owl:ObjectProperty rdf:about="#hasElement"/>
            </owl:onProperty>
            <owl:someValuesFrom rdf:resource="#SLAElement"/>
          </owl:Restriction>
        </owl:intersectionOf>
      </owl:Class>
    </owl:equivalentClass>
    <owl:disjointWith rdf:resource="#Reliability"/>
    <owl:disjointWith rdf:resource="#Competency"/>
    <owl:disjointWith rdf:resource="#Honesty"/>
  </owl:Class>

```

```

<owl:Class rdf:ID="Assurance">
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Assurance is the quality factor that tests a service to see if it maintains its
integrity during interactions.</rdfs:comment>
  <rdfs:subClassOf rdf:resource="#ReliabilityElement"/>
</owl:Class>
<owl:ObjectProperty rdf:ID="inspiredBy">
  <rdfs:domain rdf:resource="#ConceptualTrust"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasAssistedByProperty">
  <rdfs:range rdf:resource="#TrustElements"/>
  <rdfs:domain rdf:resource="#TrustConcepts"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasElement">
  <rdfs:domain rdf:resource="#TrustConcepts"/>
  <owl:inverseOf>
    <owl:ObjectProperty rdf:about="#isElementOf"/>
  </owl:inverseOf>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="isElementOf">
  <owl:inverseOf rdf:resource="#hasElement"/>
  <rdfs:range rdf:resource="#TrustConcepts"/>
  <rdfs:domain rdf:resource="#TrustElements"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasTrustProperties">
  <rdfs:range>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <rdf:Description rdf:about="http://www.w3.org/2002/07/owl#Thing"/>
        <owl:Class rdf:about="#TrustElements"/>
      </owl:unionOf>
    </owl:Class>
  </rdfs:range>
  <rdfs:domain rdf:resource="#Service"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="basedOn">
  <rdfs:domain rdf:resource="#ConcreteTrust"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="assistedBy">
  <rdfs:domain rdf:resource="#TrustConcepts"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasInspiredByProperty">
  <rdfs:domain rdf:resource="#TrustConcepts"/>
  <rdfs:range rdf:resource="#TrustElements"/>
</owl:ObjectProperty>
<owl:DatatypeProperty rdf:ID="termAndTermination">
  <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
  <rdfs:domain rdf:resource="#TermsAndConditions"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"

```

```

    >true</coreProperty>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</assistedByProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >How the user and vendor/service can cease using or providing the
service.</rdfs:comment>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="financialServices">
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</assistedByProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Financial services</rdfs:comment>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <rdfs:domain rdf:resource="#VendorHistory"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</inspiredByProperty>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="limitationOfLiability">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <rdfs:domain rdf:resource="#TermsAndConditions"/>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</assistedByProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >How much liability a vendor/service wishes to expose themselves to.</rdfs:comment>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="MD5">
    <rdfs:domain rdf:resource="#Integrity"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</assistedByProperty>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="tripleDES">
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</assistedByProperty>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >3DES can encrypt and decrypt data using a single secret key.</rdfs:comment>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>

```

```

    <rdfs:domain rdf:resource="#Confidentiality"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="exclusions">
    <rdfs:domain rdf:resource="#ServiceLevelAgreement"/>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</assistedByProperty>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Articles to which the service/vendor is not responsible for.</rdfs:comment>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="payOnTime">
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</assistedByProperty>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</inspiredByProperty>
    <rdfs:domain rdf:resource="#UserHistory"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="meanTimeBetweenFailure">
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >This value represents the average amount of time that will pass between random
failures of the service. It may be aggregated over in a composite
service.</rdfs:comment>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</inspiredByProperty>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</assistedByProperty>
    <rdfs:domain rdf:resource="#Availability"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="remediesForBreeches">
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</inspiredByProperty>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <rdfs:domain rdf:resource="#ServiceLevelAgreement"/>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</assistedByProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Efforts to be made in restoring service to SLA standards.</rdfs:comment>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="downtime">

```

```

<assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</assistedByProperty>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
<inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>false</inspiredByProperty>
<rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>Downtime is the amount of time when the service is non-operational. It is used in
conjunction with MTBF to calculate the availability probability.</rdfs:comment>
<rdfs:domain rdf:resource="#Availability"/>
<coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</coreProperty>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="workAsAdvertised">
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
<assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</assistedByProperty>
<rdfs:domain rdf:resource="#ServiceHistory"/>
<inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</inspiredByProperty>
<rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>Did users find that the service worked as advertised.</rdfs:comment>
<coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</coreProperty>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="correctExecution">
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
<rdfs:domain rdf:resource="#Assurance"/>
<assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</assistedByProperty>
<coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</coreProperty>
<inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>false</inspiredByProperty>
<rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>Assurance stipulates that actions must be executed as planned. It is a boolean
factor reflecting whether or not the vendor/service has assurance
support.</rdfs:comment>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="bpel4ws">
<inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>false</inspiredByProperty>
<rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>Business Process Execution Language 4 Web Services.</rdfs:comment>
<rdfs:domain rdf:resource="#Standards"/>
<assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</assistedByProperty>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
<coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</coreProperty>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="access">

```

```

<assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>false</assistedByProperty>
<rdfs:domain rdf:resource="#Auditing"/>
<inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>false</inspiredByProperty>
<coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</coreProperty>
<rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>This records who accessed a service and when.</rdfs:comment>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="owl">
<rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>Ontology Web Language</rdfs:comment>
<rdfs:domain rdf:resource="#Standards"/>
<inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</inspiredByProperty>
<assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</assistedByProperty>
<coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</coreProperty>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="discloseAllMaterialFacts">
<coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</coreProperty>
<inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>false</inspiredByProperty>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
<rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>The vendor/service must disclose all material facts that would influence a prudent
user in deciding whether to use a service.</rdfs:comment>
<assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</assistedByProperty>
<rdfs:domain rdf:resource="#UtmostGoodFaith"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="exactlyOnce">
<assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</assistedByProperty>
<rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>Guarantees that a message is received exactly once.</rdfs:comment>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
<inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>false</inspiredByProperty>
<rdfs:domain rdf:resource="#MsgDelivery"/>
<coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</coreProperty>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="executionTime">
<rdfs:domain rdf:resource="#Performance"/>
<coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"

```



```

    >true</coreProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Execution time is the time taken by a service to process its sequence of
activities.</rdfs:comment>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="technicalServices">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Technical services</rdfs:comment>
    <rdfs:domain rdf:resource="#VendorHistory"/>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</inspiredByProperty>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</assistedByProperty>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="kerberos">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</assistedByProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Kerberos is a network authentication protocol that provides strong authentication
in client/server environments by using secret-key cryptography.</rdfs:comment>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <rdfs:domain rdf:resource="#Authentication"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="latency">
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</assistedByProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >latency is a measure of the round-trip time between a request being sent and a
response being received.</rdfs:comment>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
    <rdfs:domain rdf:resource="#Performance"/>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</inspiredByProperty>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="PGPbased">
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</assistedByProperty>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"

```

```

    >A peer to peer approach to authenticating a principal via trusted
    parties.</rdfs:comment>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <rdfs:domain rdf:resource="#Authentication"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="wsdl">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Web Services Definition Language</rdfs:comment>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
    <rdfs:domain rdf:resource="#Standards"/>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</inspiredByProperty>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</assistedByProperty>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="qualityOfService">
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</assistedByProperty>
    <rdfs:domain rdf:resource="#ServiceLevelAgreement"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="atLeastOnce">
    <rdfs:domain rdf:resource="#MsgDelivery"/>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Guarantees that a message is received at least once.</rdfs:comment>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</assistedByProperty>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="accessedAsAdvertised">
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</assistedByProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Did users find that the service was accessed as advertised.</rdfs:comment>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>

```

```

    <rdfs:domain rdf:resource="#ServiceHistory"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="resilience">
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</assistedByProperty>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:domain rdf:resource="#Availability"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >resilience in relation to availability implies that the service is kept accessible
  in the face of attacks that attempt to make it unaccessible.</rdfs:comment>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="intellectualProperty">
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</assistedByProperty>
    <rdfs:domain rdf:resource="#TermsAndConditions"/>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >ensures that the service and certain associated articles remains the property of
  the vendor/service.</rdfs:comment>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="disclaimerOfWarranties">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</assistedByProperty>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >For vendors/services that do not accept any responsibility for harm caused by
  using a service.</rdfs:comment>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <rdfs:domain rdf:resource="#TermsAndConditions"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="descriptionAsAdvertised">
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</assistedByProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Did users find that the services description was as advertised.</rdfs:comment>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>

```

```

    <rdfs:domain rdf:resource="#ServiceHistory"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="atMostOnce">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
    <rdfs:domain rdf:resource="#MsgDelivery"/>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</assistedByProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Guarantees that a message is received at most once.</rdfs:comment>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="wsfl">
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Web Services Flow Language.</rdfs:comment>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</assistedByProperty>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <rdfs:domain rdf:resource="#Standards"/>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="uddi">
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Universal Description, Discovery, and Integration.</rdfs:comment>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</assistedByProperty>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
    <rdfs:domain rdf:resource="#Standards"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="generalProvisions">
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</assistedByProperty>
    <rdfs:domain rdf:resource="#TermsAndConditions"/>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >General provisions that are to be included in the terms and
conditions.</rdfs:comment>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
  </owl:DatatypeProperty>

```

```

<owl:DatatypeProperty rdf:ID="thirdPartyClaims">
  <rdfs:domain rdf:resource="#ServiceLevelAgreement"/>
  <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
  >false</assistedByProperty>
  <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
  >false</inspiredByProperty>
  <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
  >true</coreProperty>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Vendor/service won't infringe upon any third party copyrights, trade secrets,
patents, etc.</rdfs:comment>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="owl-s">
  <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
  >true</assistedByProperty>
  <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
  >true</inspiredByProperty>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
  <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
  >true</coreProperty>
  <rdfs:domain rdf:resource="#Standards"/>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >Ontology Web Language - Services.</rdfs:comment>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="nonDisclosure">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
  <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
  >true</inspiredByProperty>
  <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
  >true</assistedByProperty>
  <rdfs:domain rdf:resource="#Privacy"/>
  <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
  >true</coreProperty>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="personalAndLegitimateUse">
  <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
  >false</inspiredByProperty>
  <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
  >For services that are made available to users for personal, non-commercial use
only.</rdfs:comment>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
  >false</assistedByProperty>
  <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
  >true</coreProperty>
  <rdfs:domain rdf:resource="#TermsAndConditions"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="transactionTime">
  <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
  >true</coreProperty>

```

```

<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
<rdfs:domain rdf:resource="#Performance"/>
<rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>Transaction time represents the period of time that passes while the service is
completing one complete transaction.</rdfs:comment>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="soap">
<rdfs:domain rdf:resource="#Standards"/>
<inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</inspiredByProperty>
<coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</coreProperty>
<rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>Simple Object Access Protocol</rdfs:comment>
<assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</assistedByProperty>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="descriptionVsTermsAndConditions">
<assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</assistedByProperty>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
<rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>The honesty differential is a measurement of how different the description of a
service is to that services terms and conditions.</rdfs:comment>
<inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</inspiredByProperty>
<rdfs:domain rdf:resource="#HonestyDifferential"/>
<coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</coreProperty>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="ws-reliability">
<rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>Web Service - Reliability.</rdfs:comment>
<inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>false</inspiredByProperty>
<coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</coreProperty>
<assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</assistedByProperty>
<rdfs:domain rdf:resource="#Standards"/>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="forceMajeure">
<inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>false</inspiredByProperty>
<assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>false</assistedByProperty>
<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
<rdfs:domain rdf:resource="#ServiceLevelAgreement"/>
<coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"

```

```

    >true</coreProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Vendor/service is not responsible for any failure in performance due to reasons
beyond its control.</rdfs:comment>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="RSA-128">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
    <rdfs:domain rdf:resource="#Confidentiality"/>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</assistedByProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >The 128 bit RSA algorithm can be used for both public key encryption and digital
signatures.</rdfs:comment>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="usage">
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</assistedByProperty>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >false</inspiredByProperty>
    <rdfs:domain rdf:resource="#Auditing"/>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Usage is a measurement of some criteria from which billing can be established. It
could be on a time basis, a per kb basis, etc. </rdfs:comment>
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="meteredServices">
    <coreProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</coreProperty>
    <inspiredByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</inspiredByProperty>
    <rdfs:domain rdf:resource="#VendorHistory"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
    <assistedByProperty rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
    >true</assistedByProperty>
    <rdfs:comment rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
    >Metered services.</rdfs:comment>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="inspiredByProperty">
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#AnnotationProperty"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
</owl:DatatypeProperty>
<Availability rdf:ID="availability">
    <meanTimeBetweenFailure rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
    >1.23456792E8</meanTimeBetweenFailure>
    <resilience rdf:datatype="http://www.w3.org/2001/XMLSchema#string"

```

```

>high</resilience>
<isElementOf>
  <Reliability rdf:ID="reliability">
    <hasElement>
      <Performance rdf:ID="performance">
        <latency rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
        >2.0E-11</latency>
        <transactionTime rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
        >4.0E-4</transactionTime>
        <executionTime rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
        >1.0E-5</executionTime>
        <isElementOf rdf:resource="#reliability"/>
      </Performance>
    </hasElement>
    <assistedBy>
      <Honesty rdf:ID="honesty">
        <hasElement>
          <Auditing rdf:ID="auditing">
            <isElementOf rdf:resource="#honesty"/>
            <usage rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
            >50%</usage>
            <access rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
            >always</access>
          </Auditing>
        </hasElement>
        <hasElement>
          <HonestyDifferential rdf:ID="honestyDifferential">
            <descriptionVsTermsAndConditions rdf:datatype="
            "http://www.w3.org/2001/XMLSchema#string">excellent</descriptionVsTermsAndConditions>
            <isElementOf rdf:resource="#honesty"/>
          </HonestyDifferential>
        </hasElement>
        <hasElement>
          <UtmostGoodFaith rdf:ID="utmostGoodFaith">
            <isElementOf rdf:resource="#honesty"/>
            <discloseAllMaterialFacts
            rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
            >always</discloseAllMaterialFacts>
          </UtmostGoodFaith>
        </hasElement>
        <hasElement>
          <TermsAndConditions rdf:ID="termsAndConditions">
            <personalAndLegitimateUse
            rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
            >available</personalAndLegitimateUse>
            <disclaimerOfWarranties
            rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
            >no</disclaimerOfWarranties>
            <termAndTermination
            rdf:datatype="http://www.w3.org/2001/XMLSchema#string"

```



```

        >contract</termAndTermination>
        <intellectualProperty
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
        >shared</intellectualProperty>
        <isElementOf rdf:resource="#honesty"/>
        <generalProvisions
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
        >several</generalProvisions>
        <limitationOfLiability
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
        >finite</limitationOfLiability>
        </TermsAndConditions>
    </hasElement>
</Honesty>
</assistedBy>
<hasElement>
    <MsgDelivery rdf:ID="msgDelivery">
        <isElementOf rdf:resource="#reliability"/>
        <atMostOnce rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
        >false</atMostOnce>
        <exactlyOnce rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
        >false</exactlyOnce>
        <atMostOnce rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
        >true</atMostOnce>
        <atLeastOnce rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
        >true</atLeastOnce>
    </MsgDelivery>
</hasElement>
<basedOn>
    <Reputation rdf:ID="reputation">
        <basedOn>
            <Competency rdf:ID="competency">
                <hasElement>
                    <Standards rdf:ID="standards">
                        <uddi rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
                        >true</uddi>
                        <isElementOf rdf:resource="#competency"/>
                        <wsfl rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
                        >false</wsfl>
                        <owl rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
                        >true</owl>
                        <soap rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
                        >true</soap>
                        <wsdl rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
                        >true</wsdl>
                        <owl-s rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
                        >true</owl-s>
                        <ws-reliability
rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
                        >false</ws-reliability>
                        <bpel4ws rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"

```

```

        >true</bpel4ws>
    </Standards>
</hasElement>
</Competency>
</basedOn>
<basedOn rdf:resource="#honesty"/>
<hasElement>
    <ServiceHistory rdf:ID="serviceHistory">
        <isElementOf rdf:resource="#reputation"/>
        <descriptionAsAdvertised
rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
        >99.0</descriptionAsAdvertised>
        <workAsAdvertised rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
        >99.0</workAsAdvertised>
        <accessedAsAdvertised
rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
        >99.0</accessedAsAdvertised>
    </ServiceHistory>
</hasElement>
<hasElement>
    <UserHistory rdf:ID="userHistory">
        <payOnTime rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
        >true</payOnTime>
        <isElementOf rdf:resource="#reputation"/>
    </UserHistory>
</hasElement>
<hasElement>
    <VendorHistory rdf:ID="vendorHistory">
        <technicalServices
rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
        >9.5</technicalServices>
        <meteredServices rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
        >7.0</meteredServices>
        <isElementOf rdf:resource="#reputation"/>
        <financialServices
rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
        >8.0</financialServices>
    </VendorHistory>
</hasElement>
</Reputation>
</basedOn>
<hasElement>
    <Assurance rdf:ID="assurance">
        <correctExecution rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
        >true</correctExecution>
        <isElementOf rdf:resource="#reliability"/>
    </Assurance>
</hasElement>
<hasElement rdf:resource="#availability"/>
</Reliability>
</isElementOf>

```

```

<downtime rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
>0.05</downtime>
</Availability>
<Privacy rdf:ID="privacy">
  <nonDisclosure rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean"
>true</nonDisclosure>
  <isElementOf>
    <Confidence rdf:ID="confidence">
      <hasElement>
        <Integrity rdf:ID="integrity">
          <MD5 rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>available</MD5>
          <isElementOf rdf:resource="#confidence"/>
        </Integrity>
      </hasElement>
      <hasElement>
        <Authentication rdf:ID="authentication">
          <isElementOf rdf:resource="#confidence"/>
          <kerberos rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>available</kerberos>
          <PGPbased rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>available</PGPbased>
        </Authentication>
      </hasElement>
      <hasElement>
        <Confidentiality rdf:ID="confidentiality">
          <tripleDES rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>available</tripleDES>
          <RSA-128 rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>available</RSA-128>
          <isElementOf rdf:resource="#confidence"/>
        </Confidentiality>
      </hasElement>
      <inspiredBy>
        <Credibility rdf:ID="credibility">
          <assistedBy rdf:resource="#competency"/>
          <basedOn rdf:resource="#reliability"/>
          <basedOn rdf:resource="#reputation"/>
          <hasElement>
            <ServiceLevelAgreement rdf:ID="serviceLevelAgreement">
              <remediesForBreeches
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>fiinite</remediesForBreeches>
              <qualityOfService
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>excellent</qualityOfService>
              <isElementOf rdf:resource="#credibility"/>
              <forceMajeure rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>always</forceMajeure>
              <exclusions rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>none</exclusions>
            </ServiceLevelAgreement>
          </hasElement>
        </Credibility>
      </inspiredBy>
    </Confidence>
  </isElementOf>
</Privacy>

```

```

        <thirdPartyClaims
rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
        >zero</thirdPartyClaims>
    </ServiceLevelAgreement>
</hasElement>
    <basedOn rdf:resource="#honesty"/>
</Credibility>
</inspiredBy>
<hasElement rdf:resource="#privacy"/>
<inspiredBy rdf:resource="#competency"/>
<assistedBy rdf:resource="#reputation"/>
<inspiredBy rdf:resource="#reliability"/>
</Confidence>
</isElementOf>
</Privacy>
<Faith rdf:ID="faith">
    <owl:sameAs>
        <Belief rdf:ID="belief">
            <owl:sameAs rdf:resource="#faith"/>
            <assistedBy rdf:resource="#reliability"/>
            <inspiredBy rdf:resource="#confidence"/>
            <inspiredBy rdf:resource="#reputation"/>
        </Belief>
    </owl:sameAs>
</Faith>
<owl:AnnotationProperty rdf:ID="coreProperty">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
</owl:AnnotationProperty>
<owl:AnnotationProperty rdf:ID="assistedByProperty">
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#boolean"/>
</owl:AnnotationProperty>
<owl:Restriction>
    <owl:allValuesFrom rdf:resource="#Competency"/>
    <owl:onProperty rdf:resource="#isElementOf"/>
</owl:Restriction>
</rdf:RDF>

```

Instant Messaging Domain Specific Model

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:trust="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperOntology.owl#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/InstantMessaging.owl#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:daml="http://www.daml.org/2001/03/daml+oil#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xml:base="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/InstantMessaging.owl">
  <owl:Ontology rdf:about="">
    <owl:imports
rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperOntology.owl"/>
  </owl:Ontology>
  <owl:Class rdf:ID="Person"/>
  <owl:ObjectProperty rdf:ID="hasInstantMessenger">
    <rdfs:domain rdf:resource="#Person"/>
    <rdfs:range
rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperOntology.owl#TrustCo
ncepts"/>
  </owl:ObjectProperty>
  <owl:DatatypeProperty rdf:ID="conceptTrustValue">
    <rdfs:range>
      <owl:DataRange>
        <owl:oneOf rdf:parseType="Resource">
          <rdf:first rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
>1</rdf:first>
          <rdf:rest rdf:parseType="Resource">
            <rdf:rest rdf:parseType="Resource">
              <rdf:rest rdf:parseType="Resource">
                <rdf:rest
rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-
ns#nil"/>
                <rdf:first rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
>4</rdf:first>
              </rdf:rest>
            <rdf:first rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
>3</rdf:first>
          </rdf:rest>
          <rdf:first rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
>2</rdf:first>
        </owl:oneOf>
      </owl:DataRange>
    </rdfs:range>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="annotatedValue">
```

```

    <rdfs:domain
rdf:resource="http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperOntology.owl#TrustCo
ncepts"/>
    <rdfs:range>
      <owl:DataRange>
        <owl:oneOf rdf:parseType="Resource">
          <rdf:rest rdf:parseType="Resource">
            <rdf:first rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
            >2</rdf:first>
            <rdf:rest rdf:parseType="Resource">
              <rdf:rest rdf:parseType="Resource">
                <rdf:first rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
                >4</rdf:first>
                <rdf:rest
                    rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-
ns#nil"/>
              </rdf:rest>
            <rdf:first rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
            >3</rdf:first>
          </rdf:rest>
        </owl:oneOf>
      </owl:DataRange>
    </rdfs:range>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="source">
    <rdfs:domain rdf:resource="#Person"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="name">
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="destination">
    <rdfs:domain rdf:resource="#Person"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  </owl:DatatypeProperty>
  <trust:Reliability rdf:ID="Reliability_2">
    <annotatedValue rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
    >1</annotatedValue>
  </trust:Reliability>
  <Person rdf:ID="Person_4">
    <hasInstantMessenger>
      <trust:Credibility rdf:ID="Credibility_4">
        <annotatedValue rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
        >1</annotatedValue>
      </trust:Credibility>
    </hasInstantMessenger>
    <hasInstantMessenger>
      <trust:Confidence rdf:ID="Confidence_7">
        <annotatedValue rdf:datatype="http://www.w3.org/2001/XMLSchema#int"

```

```

    >1</annotatedValue>
  </trust:Confidence>
</hasInstantMessenger>
<hasInstantMessenger>
  <trust:Competency rdf:ID="Competency_5">
    <annotatedValue rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
    >1</annotatedValue>
  </trust:Competency>
</hasInstantMessenger>
<name rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
>kaquinn</name>
<hasInstantMessenger>
  <trust:Reputation rdf:ID="Reputation_1">
    <annotatedValue rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
    >1</annotatedValue>
  </trust:Reputation>
</hasInstantMessenger>
<destination rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
></destination>
<hasInstantMessenger>
  <trust:Faith rdf:ID="Faith_6">
    <annotatedValue rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
    >1</annotatedValue>
  </trust:Faith>
</hasInstantMessenger>
<hasInstantMessenger rdf:resource="#Reliability_2"/>
<source rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
></source>
<hasInstantMessenger>
  <trust:Belief rdf:ID="Belief_5">
    <annotatedValue rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
    >1</annotatedValue>
  </trust:Belief>
</hasInstantMessenger>
<hasInstantMessenger>
  <trust:Honesty rdf:ID="Honesty_3">
    <annotatedValue rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
    >1</annotatedValue>
  </trust:Honesty>
</hasInstantMessenger>
</Person>
</rdf:RDF>

```

APPENDIX II – Research Experiments

Experiment Data Sets

The anonymously collected data sets for all four experiments can be found on the DVD media that accompanies this thesis. All experiment data for experiment one is filed under 'Experiment Data', 'Experiment One'. Data for experiments two, three, and four can be found in their respective folder in the 'Experiment Data' folder.

The experiments presented in this appendix enable the reader to quickly examine each experiment questionnaire. In addition, these printed questionnaires are presented in a similar format to the actual online experiments, which may be helpful to the reader.

Experiment One – Necessity of Personalisable, Multi-Faceted Approach

Trust Questionnaire – Outline -1

Thank you for taking 3-5 minutes of your time to aid me in my Ph.D. research. If you have any questions please feel free to contact me [here](#). Please note that the U2 competition is no longer available after 14.00 GMT on January 4th 2005.

-Outline-

This questionnaire is comprised of three simple scenarios in which you will be asked to rate a set of characteristics that relate to trust. Each characteristic is rated on the basis of how useful you think it is when determining a level of trust specific to each scenario.

-Directions-

Scenario one presents a low risk scenario (\$10), scenario two is medium risk (\$100), and scenario three is high risk (\$1000). The only difference between each scenario is the level of risk involved. Please work your way through the all three scenarios by following the page instructions. Click start to begin...

-NB-

The questionnaire is best viewed in Internet Explorer. You will probably find the questionnaire slow to begin with but it does speed up dramatically!

[Start ->](#)

Trust Questionnaire - Section THREE - 1

You are interested in purchasing an item online for \$1000 and in deciding to buy this item you need to determine how much you trust the seller. (assume that there is no credit card fraud involved).

For each characteristic below please rate how useful that characteristic is to you when determining trust in this scenario...

BELIEF (...that the seller gives you all the information on the product so that you can make a sound decision on purchase.)

- Very Low
 - Low
 - No Opinion
 - High
 - Very High
-

COMPETENCE (...that the seller is able to fulfill his/her obligations.)

- Very Low
 - Low
 - No Opinion
 - High
 - Very High
-

CONFIDENCE (...that you hold in the seller in regards this transaction.)

- Very Low
 - Low
 - No Opinion
 - High
 - Very High
-

CREDIBILITY (... how credible you think the sellers information is in this transaction.)

- Very Low
 - Low
 - No Opinion
 - High
 - Very High
-

FAITH (... your faith in the seller in regards this transaction.)

- Very Low
 - Low
 - No Opinion
 - High
 - Very High
-

HONESTY (...that the seller is being honest in regards this transaction.)

- Very Low
 - Low
 - No Opinion
 - High
 - Very High
-

RELIABILITY (...that the seller is a reliable person.)

- Very Low
 - Low
 - No Opinion
 - High
 - Very High
-

REPUTATION (...that the seller is a reputable person.)

- Very Low
 - Low
 - No Opinion
 - High
 - Very High
-

Next

Trust Questionnaire - Section THREE - 2

Please rate each trust characteristic (on the Left Hand Side below) in order from first place to third (1-3). First (1) is the most important and the third (3) is the least important to you when determine how much you trust the seller in the current scenario.

	1st	2nd	3rd
Belief	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Competence	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Confidence	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Credibility	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Faith	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Honesty	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Reliability	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Reputation	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Where the above characteristics are described as...

- BELIEF (that the seller gives you all the information on the product so that you can make a sound decision on purchase.)
- COMPETENCE (...that the seller is able to fulfill his/her obligations.)
- CONFIDENCE (...that you hold in the seller in regards the transaction.)
- CREDIBILITY (... how credible you think the sellers information is in the transaction.)
- FAITH (... your faith in the seller in regards the transaction.)
- HONESTY (...that the seller is being honest in regards the transaction.)
- RELIABILITY (...that the seller is a reliable person.)
- REPUTATION (...that the seller is reputable person.)

Next

Trust Questionnaire - Section THREE - 3

The characteristic have now been saved. You will now be asked to rank your highest chosen characteristics...

[Next...](#)

Trust Questionnaire - Section THREE - 4

Please state which ONE of the listed seven characteristics most influences your determination of:

- Trust Concept [1] -

- competence
- confidence
- credibility
- faith
- honesty
- reliability
- reputation

Please state which ONE of the listed seven characteristics most influences your determination of:

- Trust Concept [2] -

- belief
- confidence
- credibility
- faith
- honesty
- reliability
- reputation

Please state which ONE of the listed seven characteristics most influences your determination of:

- Trust Concept [3] -

- belief
- competence
- credibility
- faith
- honesty
- reliability
- reputation

Confirm..

Trust Questionnaire - Section THREE - 5

You have now completed the final scenario. Please continue to the small information and competition page...

[Next...](#)

Trust Questionnaire – Information & Competition - 1

Information & U2 @ Croke Park Competition

Please answer the following questions and (optionally) enter your email address to enter the U2 competition.

-Have you ever purchased something online?

- Yes
 - No
 - Rather not say.
-

Please enter the following personal information...

-Sex

- Male
 - Female
 - Rather not say.
-

-Age

- Under 20
 - 20 – 29
 - 30 – 39
 - 40 – 49
 - Over 50
 - Rather not say.
-

Enter your email address below to be in with a chance to win 2 tickets to see U2 perform live in Croke Park, Ireland, in June 2005!

Enter Email

Finish!

[\[Privacy Policy\]](#)

Trust Questionnaire – Information & Competition - 2

Thank You...

Thank you for taking the time to fill in this questionnaire. Good luck in the U2 competition!!
Winner announced 11th February 2005...

Thank you for taking the time to fill in this questionnaire. Good luck in the U2 competition!!
Winner announced 11th February 2005...

Trust Questionnaire – Privacy Policy - 1

Online Questionnaire Privacy Policy

In order to enter the U2 competition you must specify an email address that will enable us to contact you if you have won. The collection and storage of your email address is solely for use during this one-off competition and will therefore never be disseminated and will be permanently deleted after the competition winner has claimed his/her prize.

In addition to this it is important to note that your email address is not tied to any of the answers you provided whilst filling out this questionnaire.

Experiment Two – Accuracy Survey

Survey Overview - 1

Survey Overview

This questionnaire is divided into five short sections and takes approximately 6-8 minutes in total.

The first section (approx. 60% of the total questionnaire) relates to the generation of a personalised model of trust for you. In the second section you will be asked to annotate some people with trust information. Section three asks you to assign minimum trust requirements to a set of actions. Finally, you will be asked to answer some simple questions relating actions to people.

At the start of each section you will be given specific instructions relevant to that section.

[Start](#)

Section ONE - Trust Model Generation – 1

Trust Model Generation

Section ONE

You are now about to generate a personalised model of trust. During this process eight concepts related to trust will be presented (separately) to you. The eight concepts related to trust are listed alphabetically below:

- Belief
- Competency
- Confidence
- Credibility
- Faith
- Honesty
- Reliability
- Reputation

When each concept is presented to you, please select any of the other seven concepts that you think the presented concept influences. For example, one might think that Credibility is influenced by Reputation.

Begin

Section ONE - Trust Model Generation – 2

Belief (1 of 8)

In your opinion, which of the following concepts, if any, does BELIEF influence:

Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Next

Section ONE - Trust Model Generation – 3

Competency (2 of 8)

In your opinion, which of the following concepts, if any, does COMPETENCY influence:

Belief	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Next

Section ONE - Trust Model Generation – 4

Confidence (3 of 8)

In your opinion, which of the following concepts, if any, does CONFIDENCE influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Next

Section ONE - Trust Model Generation – 5

Credibility (4 of 8)

In your opinion, which of the following concepts, if any, does CREDIBILITY influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Next

Section ONE - Trust Model Generation – 6

Faith (5 of 8)

In your opinion, which of the following concepts, if any, does FAITH influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Next

Section ONE - Trust Model Generation – 7

Honesty (6 of 8)

In your opinion, which of the following concepts, if any, does FAITH influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Section ONE - Trust Model Generation – 8

Reliability (7 of 8)

In your opinion, which of the following concepts, if any, does RELIABILITY influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Section ONE - Trust Model Generation – 9

Reputation (8 of 8)

In your opinion, which of the following concepts, if any, does REPUTATION influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
All	<input type="checkbox"/>

Section TWO - Trust Annotation – 1

Section TWO [55% Total Questionnaire Completed]

You are now about to annotate four different people with trust information. Each of these four people will belong to a specific aspect of your life, e.g. a family member or work colleague. Annotating each person will require you to select a level of trust for each trust concept, e.g. a family member may have 'very high' reliability.

Please think of a real person that you know and who fits the category presented. These people will be used later in the questionnaire. The identity of this person is not required.

Begin

Section TWO - Trust Annotation – 2

Family Member (1 of 4)

Please assign a particular FAMILY MEMBER with a value for each of the trust concepts below:

Belief:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Competency:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Confidence:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Credibility:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Faith:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Honesty:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Reliability:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Reputation:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Next

Section TWO - Trust Annotation – 3

Work Colleague (2 of 4)

Please assign a particular WORK COLLEAGUE with a value for each of the trust concepts below:

Belief:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Competency:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Confidence:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Credibility:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Faith:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Honesty:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Reliability:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Reputation:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Next

Section TWO - Trust Annotation - 4

Friend of a Friend Member (3 of 4)

Please assign a particular FRIEND OF A FRIEND with a value for each of the trust concepts below:

Belief:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Competency:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Confidence:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Credibility:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Faith:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Honesty:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Reliability:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Reputation:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Next

Section TWO - Trust Annotation – 5

Complete Stranger (4 of 4)

Please assign a particular COMPLETE STRANGER with a value for each of the trust concepts below:

Belief:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Competency:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Confidence:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Credibility:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Faith:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Honesty:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Reliability:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Reputation:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Next

Section THREE - Rule Creation - 1

Section THREE [85% Total Questionnaire Completed]

You are now about to create four rules. For each rule please assign a minimum level of trust that you would require in someone in order to allow the action that is presented to take place.

Begin

Section THREE - Rule Creation – 2

Rule Creation

Please assign the minimum level of trust you would require in someone in order to allow the use each of the following:

(1) PENCIL.

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

(2) BANK PIN.

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

(3) LAPTOP.

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

(4) MOBILE PHONE.

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Next

Section FOUR - Survey – 1

Section FOUR [93% Total Questionnaire Completed]

To conclude you will be asked whether or not you would allow a certain action to take place with respect to each of the people you had in mind from earlier.

Next

Section FOUR - Survey – 2

Scenario Survey (1 of 1)

For each of the actions below please state who you would allow to perform that action:

Who would you let use your PENCIL?

	Allow	Not Allow
Family Member	<input type="radio"/>	<input type="radio"/>
Work Colleague	<input type="radio"/>	<input type="radio"/>
Friend of a Friend	<input type="radio"/>	<input type="radio"/>
Stranger	<input type="radio"/>	<input type="radio"/>

Who would you let use your BANK PIN?

	Allow	Not Allow
Family Member	<input type="radio"/>	<input type="radio"/>
Work Colleague	<input type="radio"/>	<input type="radio"/>
Friend of a Friend	<input type="radio"/>	<input type="radio"/>
Stranger	<input type="radio"/>	<input type="radio"/>

Who would you let use your LAPTOP?

	Allow	Not Allow
Family Member	<input type="radio"/>	<input type="radio"/>
Work Colleague	<input type="radio"/>	<input type="radio"/>
Friend of a Friend	<input type="radio"/>	<input type="radio"/>
Stranger	<input type="radio"/>	<input type="radio"/>

Who would you let use your MOBILE PHONE?

	Allow	Not Allow
Family Member	<input type="radio"/>	<input type="radio"/>
Work Colleague	<input type="radio"/>	<input type="radio"/>
Friend of a Friend	<input type="radio"/>	<input type="radio"/>
Stranger	<input type="radio"/>	<input type="radio"/>

Next

Survey Complete - 1

Survey Complete

Enter the competition to win 2 tickets to Robbie Williams in Croke Park, Dublin, Ireland on June 9th 2006 by entering your email address below and press the 'Finish!' button. [\[Privacy Policy\]](#)

Email Address:

Survey Complete - 2

Thank You

Thank you for taking the time to fill in this questionnaire.

Good luck in the Robbie Williams competition!!

You will be notified by email if you have won the competition before the end of February 2006. Check back then at www.karlquinn.com for further information.

Follow Up Survey – 1

Short Survey

Where applicable please choose the person(s) who would **NOT** be allowed to...

1. know your **BANK PIN**

Very Low Trusted Person: <input type="checkbox"/>	Low Trusted Person: <input type="checkbox"/>	High Trusted Person: <input type="checkbox"/>	Very High Trusted Person: <input type="checkbox"/>
--	---	--	---

2. borrow your **LAPTOP**

Very Low Trusted Person: <input type="checkbox"/>	Low Trusted Person: <input type="checkbox"/>	High Trusted Person: <input type="checkbox"/>	Very High Trusted Person: <input type="checkbox"/>
--	---	--	---

3. use your **MOBILE PHONE**

Very Low Trusted Person: <input type="checkbox"/>	Low Trusted Person: <input type="checkbox"/>	High Trusted Person: <input type="checkbox"/>	Very High Trusted Person: <input type="checkbox"/>
--	---	--	---

4. use your **PENCIL**

Very Low Trusted Person: <input type="checkbox"/>	Low Trusted Person: <input type="checkbox"/>	High Trusted Person: <input type="checkbox"/>	Very High Trusted Person: <input type="checkbox"/>
--	---	--	---

Email Address (please use same email as used in Robbie Williams competition [why?]):

Finish

Experiment Three – Accuracy Survey with Additional Information

Survey Overview - 1

Survey Overview

This questionnaire is divided into 4 sections and takes approximately 12-14 minutes in total.

At the start of each section you will be given specific instructions relevant to that section. Below is an overview of the 4 sections;

1. Generation of a personalised model of trust.
2. Annotate some people with trust information.
3. Create a set of rules regarding the set of actions.
4. Answer some simple questions relating actions to people.

Please enter your email address below (competition). If you have completed any of my questionnaires before then please use the email address that you used before again. Then press the 'start' button.

Email Address:

[\[Privacy Policy\]](#)

Start

Section ONE - Trust Model Generation – 1

Trust Model Generation

Section ONE

You are now about to generate a personalised model of trust. During this process eight concepts related to trust will be presented (separately) to you. The eight concepts related to trust are listed alphabetically below:

- Belief
- Competency
- Confidence
- Credibility
- Faith
- Honesty
- Reliability
- Reputation

When each concept is presented to you, please select any of the other seven concepts that you think the presented concept influences. For example, one might think that Credibility is influenced by Reputation.

Begin

Section ONE - Trust Model Generation – 2

Belief (1 of 8)

In your opinion, which of the following concepts, if any, does BELIEF influence:

Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Next

Section ONE - Trust Model Generation – 3

Competency (2 of 8)

In your opinion, which of the following concepts, if any, does COMPETENCY influence:

Belief	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Next

Section ONE - Trust Model Generation – 4

Confidence (3 of 8)

In your opinion, which of the following concepts, if any, does CONFIDENCE influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Next

Section ONE - Trust Model Generation – 5

Credibility (4 of 8)

In your opinion, which of the following concepts, if any, does CREDIBILITY influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Next

Section ONE - Trust Model Generation – 6

Faith (5 of 8)

In your opinion, which of the following concepts, if any, does FAITH influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Next

Section ONE - Trust Model Generation – 7

Honesty (6 of 8)

In your opinion, which of the following concepts, if any, does FAITH influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Section ONE - Trust Model Generation – 8

Reliability (7 of 8)

In your opinion, which of the following concepts, if any, does RELIABILITY influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Section ONE - Trust Model Generation – 9

Reputation (8 of 8)

In your opinion, which of the following concepts, if any, does REPUTATION influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
All	<input type="checkbox"/>

Section TWO - Trust Annotation – 1

Section TWO [40% Total Questionnaire Completed]

You are now about to annotate four different people with trust information. Each of these four people will belong to a specific aspect of your life, e.g. a **FAMILY MEMBER** or **WORK COLLEAGUE**. Annotating each person will require you to select a level of trust for each trust concept, e.g. a **FAMILY MEMBER** may have 'very high' reliability.

Please think of a real person that you know and who fits the category presented. These people will be used later in the questionnaire. The identity of this person is not required.

Begin

Section TWO - Trust Annotation – 2

Family Member (1 of 4)

Please assign a particular FAMILY MEMBER with a value for each of the trust concepts below:

Belief:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Competency:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Confidence:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Credibility:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Faith:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Honesty:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Reliability:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Reputation:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Next

Section TWO - Trust Annotation – 3

Work Colleague (2 of 4)

Please assign a particular WORK COLLEAGUE with a value for each of the trust concepts below:

Belief:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Competency:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Confidence:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Credibility:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Faith:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Honesty:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Reliability:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Reputation:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Next

Section TWO - Trust Annotation - 4

Friend of a Friend Member (3 of 4)

Please assign a particular FRIEND OF A FRIEND with a value for each of the trust concepts below:

Belief:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Competency:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Confidence:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Credibility:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Faith:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Honesty:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Reliability:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Reputation:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Next

Section TWO - Trust Annotation – 5

Complete Stranger (4 of 4)

Please assign a particular COMPLETE STRANGER with a value for each of the trust concepts below:

Belief:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Competency:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Confidence:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Credibility:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Faith:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Honesty:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Reliability:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Reputation:

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

Next

Section THREE - Rule Creation - 1

Section THREE [55% Total Questionnaire Completed]

In this section you will be asked to create a set of four rules.

Next

Section THREE - Rule Creation – 2

Rule Creation (1 of 4) PENCIL

This page should be completed with respect to allowing someone to use your **PENCIL**.

(1) What is the minimum amount of trust that you would have to hold in a person to in order to allow that person to use your **PENCIL**?

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

*If from time to time you think that you may need additional information to make a sound decision regarding allowing the use of your **PENCIL** then please answer question 2. If you feel that you won't need additional information to make this decision then please press the 'next' button to continue.*

(2) Please select one or more options from the list below that is likely to help convince you to allow someone to use your **PENCIL**.

(i) The amount of time that the **PENCIL** will be borrowed matters to you and the following is acceptable:

Short term is acceptable. <input type="checkbox"/>	Long term is acceptable. <input type="checkbox"/>
--	---

(ii) It matters to you where the **PENCIL** will be used and the following is acceptable:

Under your supervision is acceptable. <input type="checkbox"/>	Outside your supervision is acceptable. <input type="checkbox"/>
--	--

(iii) It matters to you what the **PENCIL** is used for and the following is acceptable:

Official use is acceptable. <input type="checkbox"/>	Personal use is acceptable. <input type="checkbox"/>
--	--

(iv) Something else would convince you, please specify:

Next

Section THREE - Rule Creation – 3

Rule Creation (2 of 4) BANK PIN

This page should be completed with respect to allowing someone to use your **BANK PIN**.

(1) What is the minimum amount of trust that you would have to hold in a person to in order to allow that person to use your **BANK PIN**?

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

*If from time to time you think that you may need additional information to make a sound decision regarding allowing the use of your **BANK PIN** then please answer question 2. If you feel that you won't need additional information to make this decision then please press the 'next' button to continue.*

(2) Please select one or more options from the list below that is likely to help convince you to allow someone to use your **BANK PIN**.

(i) The amount of time that the **BANK PIN** will be borrowed matters to you and the following is acceptable:

Short term is acceptable. <input type="checkbox"/>	Long term is acceptable. <input type="checkbox"/>
--	---

(ii) It matters to you where the **BANK PIN** will be used and the following is acceptable:

Under your supervision is acceptable. <input type="checkbox"/>	Outside your supervision is acceptable. <input type="checkbox"/>
--	--

(iii) It matters to you what the **BANK PIN** is used for and the following is acceptable:

Official use is acceptable. <input type="checkbox"/>	Personal use is acceptable. <input type="checkbox"/>
--	--

(iv) Something else would convince you, please specify:

Next

Section THREE - Rule Creation – 4

Rule Creation (3 of 4) LAPTOP

This page should be completed with respect to allowing someone to use your **LAPTOP**.

(1) What is the minimum amount of trust that you would have to hold in a person to in order to allow that person to use your **LAPTOP**?

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

*If from time to time you think that you may need additional information to make a sound decision regarding allowing the use of your **LAPTOP** then please answer question 2. If you feel that you won't need additional information to make this decision then please press the 'next' button to continue.*

(2) Please select one or more options from the list below that is likely to help convince you to allow someone to use your **LAPTOP**.

(i) The amount of time that the **LAPTOP** will be borrowed matters to you and the following is acceptable:

Short term is acceptable. <input type="checkbox"/>	Long term is acceptable. <input type="checkbox"/>
--	---

(ii) It matters to you where the **LAPTOP** will be used and the following is acceptable:

Under your supervision is acceptable. <input type="checkbox"/>	Outside your supervision is acceptable. <input type="checkbox"/>
--	--

(iii) It matters to you what the **LAPTOP** is used for and the following is acceptable:

Official use is acceptable. <input type="checkbox"/>	Personal use is acceptable. <input type="checkbox"/>
--	--

(iv) Something else would convince you, please specify:

Next

Section THREE - Rule Creation – 5

Rule Creation (4 of 4) Mobile Phone

This page should be completed with respect to allowing someone to use your **MOBILE PHONE**.

(1) What is the minimum amount of trust that you would have to hold in a person to in order to allow that person to use your **MOBILE PHONE**?

Very Low: <input type="checkbox"/>	Low: <input type="checkbox"/>	High: <input type="checkbox"/>	Very High: <input type="checkbox"/>
------------------------------------	-------------------------------	--------------------------------	-------------------------------------

*If from time to time you think that you may need additional information to make a sound decision regarding allowing the use of your **MOBILE PHONE** then please answer question 2. If you feel that you won't need additional information to make this decision then please press the 'next' button to continue.*

(2) Please select one or more options from the list below that is likely to help convince you to allow someone to use your **MOBILE PHONE**.

(i) The amount of time that the **MOBILE PHONE** will be borrowed matters to you and the following is acceptable:

Short term is acceptable. <input type="checkbox"/>	Long term is acceptable. <input type="checkbox"/>
--	---

(ii) It matters to you where the **MOBILE PHONE** will be used and the following is acceptable:

Under your supervision is acceptable. <input type="checkbox"/>	Outside your supervision is acceptable. <input type="checkbox"/>
--	--

(iii) It matters to you what the **MOBILE PHONE** is used for and the following is acceptable:

Official use is acceptable. <input type="checkbox"/>	Personal use is acceptable. <input type="checkbox"/>
--	--

(iv) Something else would convince you, please specify:

Next

Section FOUR - Questionnaire – 1

Questionnaire Overview

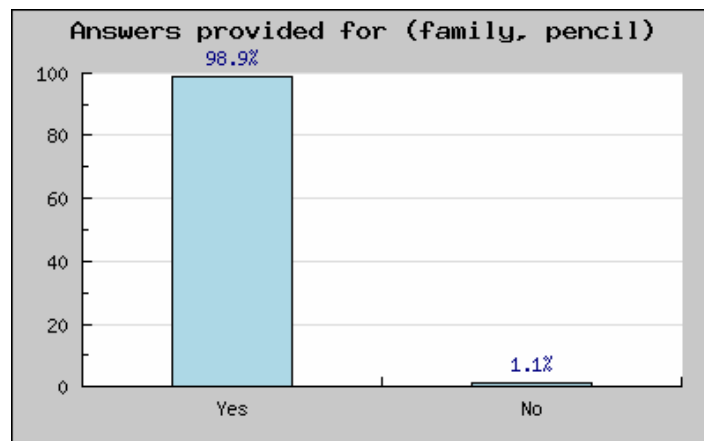
Section FOUR [70% Total Questionnaire Completed]

This is the final section. You will be asked to provide a 'yes' or 'no' answer to questions such as 'Would you allow a **FAMILY MEMBER** use your **PENCIL**?'. From time to time you may feel that you need extra information to help you make a sound decision. An '**Ask the Audience**' button and subsequently a '**Provide Guarantees**' button (see below) provides extra information.

'Ask the Audience' Overview

You can press the '**Ask the Audience**' if desired, and as often as you like, to see the opinion of 282 community members of the Department of Computer Science, Trinity College Dublin. Their opinion was gathered in an identical survey and it may help you make your own decision.

*The example below shows the percentage of yes and no answers provided by the 282 members. Here, approximately 99% of the 282 members that they would allow a **FAMILY MEMBER** use their **PENCIL**. Note: they were given no additional or specific information.*



'Provide Guarantees' Overview

If the '**Ask the Audience**' information does not help you make a decision you can request guarantees by pressing the '**Provide Guarantees**' button. These guarantees relate to the time, location, and usage for the current action. Note: You must first click '**Ask the Audience**' in order to be able to click '**Provide Guarantees**'.

*Example guarantee: **FAMILY MEMBER** will only borrow the **PENCIL** for official purposes and for a short time.*

Begin

Section FOUR - Questionnaire – 2a

Questionnaire

Would you allow a **FAMILY MEMBER** to borrow your **PENCIL**?

Yes: <input type="checkbox"/>	No: <input type="checkbox"/>
-------------------------------	------------------------------

Ask the Audience

Submit Answer

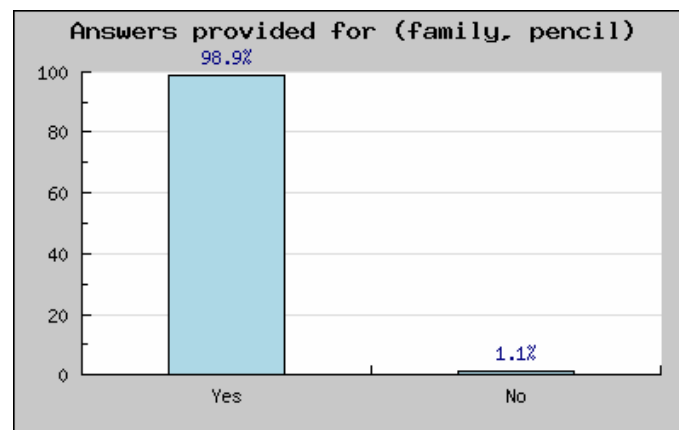
Section FOUR - Questionnaire – 2b

Questionnaire

Would you allow a **FAMILY MEMBER** to borrow your **PENCIL**?

ASK THE AUDIENCE DATA

We asked 282 of the Computer Science community in Trinity College Dublin the question: 'Would you allow a **FAMILY MEMBER** to borrow your **PENCIL**?'. Their opinion is:



- If you can make a decision straight away then please select either 'yes' or 'no' and press the 'Submit Answer' button.

- If you need guarantees (e.g. time, location, usage) to help make a decision then ONLY press the 'Provide Guarantees' button below?

Yes: <input type="checkbox"/>	No: <input type="checkbox"/>
-------------------------------	------------------------------

[Provide Guarantees](#)

[Submit Answer](#)

Section FOUR - Questionnaire – 2c

Questionnaire

Would you allow a **FAMILY MEMBER** to borrow your **PENCIL**?

GUARANTEES PROVIDED

Assume that the **FAMILY MEMBER** has provided you with the following guarantees regarding borrow your **PENCIL**.

- No guarantees are available.

Please select either 'yes' or 'no' and press the 'Submit Answer' button.

Yes: <input type="checkbox"/>	No: <input type="checkbox"/>
-------------------------------	------------------------------

Submit Answer

Section FOUR - Questionnaire – 3a

Questionnaire

Would you allow a **WORK COLLEAGUE** to borrow your **PENCIL**?

Yes: <input type="checkbox"/>	No: <input type="checkbox"/>
-------------------------------	------------------------------

Ask the Audience

Submit Answer

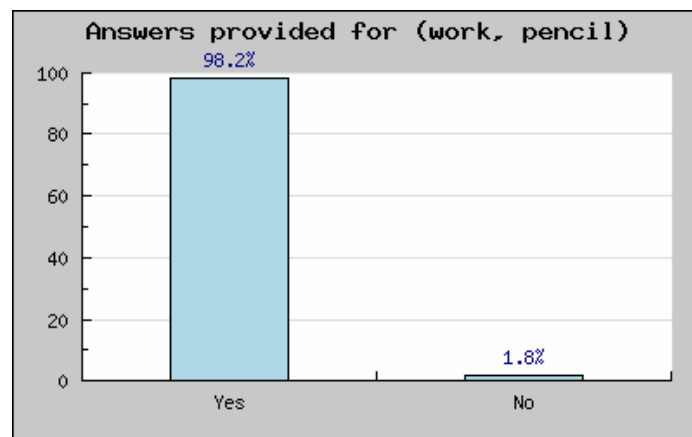
Section FOUR - Questionnaire – 3b

Questionnaire

Would you allow a **WORK COLLEAGUE** to borrow your **PENCIL**?

ASK THE AUDIENCE DATA

We asked 282 of the Computer Science community in Trinity College Dublin the question: 'Would you allow a **WORK COLLEAGUE** to borrow your **PENCIL**?'. Their opinion is:



- If you can make a decision straight away then please select either 'yes' or 'no' and press the 'Submit Answer' button.

- If you need guarantees (e.g. time, location, usage) to help make a decision then ONLY press the 'Provide Guarantees' button below?

Yes: <input type="checkbox"/>	No: <input type="checkbox"/>
-------------------------------	------------------------------

Section FOUR - Questionnaire – 3c

Questionnaire

Would you allow a **WORK COLLEAGUE** to borrow your **PENCIL**?

GUARANTEES PROVIDED

Assume that the **WORK COLLEAGUE** has provided you with the following guarantees regarding borrow your **PENCIL**.

- No guarantees are available.

Please select either 'yes' or 'no' and press the 'Submit Answer' button.

Yes: <input type="checkbox"/>	No: <input type="checkbox"/>
-------------------------------	------------------------------

Submit Answer

Section FOUR - Questionnaire – 4a

Questionnaire

Would you allow a **FRIEND OF A FRIEND** to borrow your **PENCIL**?

Yes: <input type="checkbox"/>	No: <input type="checkbox"/>
-------------------------------	------------------------------

Ask the Audience

Submit Answer

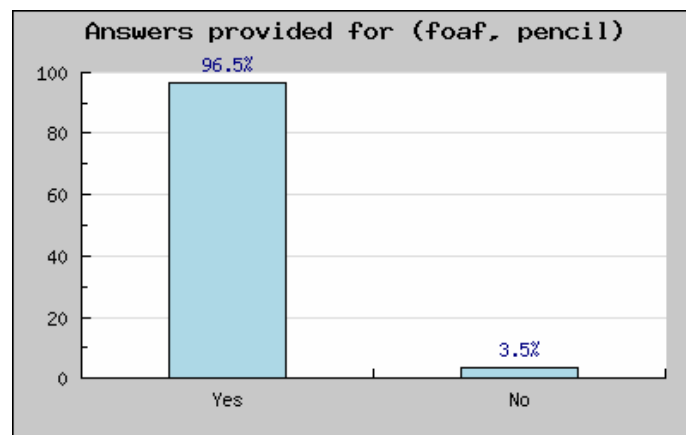
Section FOUR - Questionnaire – 4b

Questionnaire

Would you allow a **FRIEND OF A FRIEND** to borrow your **PENCIL**?

ASK THE AUDIENCE DATA

We asked 282 of the Computer Science community in Trinity College Dublin the question: 'Would you allow a **FRIEND OF A FRIEND** to borrow your **PENCIL**?'. Their opinion is:



- If you can make a decision straight away then please select either 'yes' or 'no' and press the 'Submit Answer' button.

- If you need guarantees (e.g. time, location, usage) to help make a decision then ONLY press the 'Provide Guarantees' button below?

Yes: <input type="checkbox"/>	No: <input type="checkbox"/>
-------------------------------	------------------------------

Section FOUR - Questionnaire – 4c

Questionnaire

Would you allow a **FRIEND OF A FRIEND** to borrow your **PENCIL**?

GUARANTEES PROVIDED

Assume that the **FRIEND OF A FRIEND** has provided you with the following guarantees regarding borrow your **PENCIL**.

- No guarantees are available.

Please select either 'yes' or 'no' and press the 'Submit Answer' button.

Yes: <input type="checkbox"/>	No: <input type="checkbox"/>
-------------------------------	------------------------------

Submit Answer

Section FOUR - Questionnaire – 5a

Questionnaire

Would you allow a **COMPLETE STRANGER** to borrow your **PENCIL**?

Yes: <input type="checkbox"/>	No: <input type="checkbox"/>
-------------------------------	------------------------------

Ask the Audience

Submit Answer

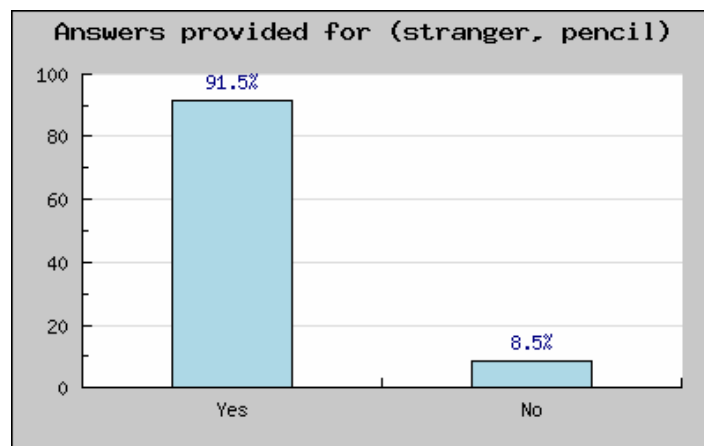
Section FOUR - Questionnaire – 5b

Questionnaire

Would you allow a **COMPLETE STRANGER** to borrow your **PENCIL**?

ASK THE AUDIENCE DATA

We asked 282 of the Computer Science community in Trinity College Dublin the question: 'Would you allow a **COMPLETE STRANGER** to borrow your **PENCIL**?'. Their opinion is:



- If you can make a decision straight away then please select either 'yes' or 'no' and press the 'Submit Answer' button.

- If you need guarantees (e.g. time, location, usage) to help make a decision then ONLY press the 'Provide Guarantees' button below?

Yes: <input type="checkbox"/>	No: <input type="checkbox"/>
-------------------------------	------------------------------

Section FOUR - Questionnaire – 5c

Questionnaire

Would you allow a **COMPLETE STRANGER** to borrow your **PENCIL**?

GUARANTEES PROVIDED

Assume that the **COMPLETE STRANGER** has provided you with the following guarantees regarding borrow your **PENCIL**.

- No guarantees are available.

Please select either 'yes' or 'no' and press the 'Submit Answer' button.

Yes: <input type="checkbox"/>	No: <input type="checkbox"/>
-------------------------------	------------------------------

Submit Answer

Experiment Four – Clarity Survey

Survey Overview - 1

Short Survey

This is a short follow up survey from last weeks survey. Today, we are trying to evaluate the usability of the last weeks survey.

You will be shown some screenshots from last weeks survey and you will be asked to answer two questions associated with each screenshot. Before you start please re-read last weeks instructions (screenshot below) to remind you of what your were asked to do.

Trust Model Generation

Section ONE

You are now about to generate a personalised model of trust. During this process eight concepts related to trust will be presented (separately) to you. The eight concepts related to trust are listed alphabetically below:

- ◆ Belief
- ◆ Competency
- ◆ Confidence
- ◆ Credibility
- ◆ Faith
- ◆ Honesty
- ◆ Reliability
- ◆ Reputation

When each concept is presented to you, please select any of the other seven concepts that you think the presented concept influences. For example, one might think that Credibility is influenced by Reputation.

[Begin](#)

Please provide the email address as used in last weeks survey (anonymity retained): [\[Privacy Policy\]](#)

[Start](#)

Section ONE – Short Survey – 1

Given the screenshot below, please answer the following questions;

Belief (1 of 8)

In your opinion, which of the following concepts, if any, does BELIEF influence:

Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Next

Q: How clear was your understanding of **BELIEF** when you were asked if it influenced other concepts?

Very Unclear: <input type="checkbox"/>	Unclear: <input type="checkbox"/>	Clear: <input type="checkbox"/>	Very Clear: <input type="checkbox"/>
--	-----------------------------------	---------------------------------	--------------------------------------

Q: Briefly describe what you took **BELIEF** to mean.

(Please do not use apostrophes ' in your answer)

Next

Section ONE – Short Survey – 2

Given the screenshot below, please answer the following questions;

Competency (2 of 8)

In your opinion, which of the following concepts, if any, does COMPETENCY influence:

Belief	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Next

Q: How clear was your understanding of **COMPETENCY** when you were asked if it influenced other concepts?

Very Unclear: <input type="checkbox"/>	Unclear: <input type="checkbox"/>	Clear: <input type="checkbox"/>	Very Clear: <input type="checkbox"/>
--	-----------------------------------	---------------------------------	--------------------------------------

Q: Briefly describe what you took **COMPETENCY** to mean.

(Please do not use apostrophes ' in your answer)

Next

Section ONE – Short Survey – 3

Given the screenshot below, please answer the following questions;

Confidence (3 of 8)

In your opinion, which of the following concepts, if any, does CONFIDENCE influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Next

Q: How clear was your understanding of **CONFIDENCE** when you were asked if it influenced other concepts?

Very Unclear: <input type="checkbox"/>	Unclear: <input type="checkbox"/>	Clear: <input type="checkbox"/>	Very Clear: <input type="checkbox"/>
--	-----------------------------------	---------------------------------	--------------------------------------

Q: Briefly describe what you took **CONFIDENCE** to mean.

(Please do not use apostrophes ' in your answer)

Next

Section ONE – Short Survey – 4

Given the screenshot below, please answer the following questions;

Credibility (4 of 8)

In your opinion, which of the following concepts, if any, does CREDIBILITY influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Next

Q: How clear was your understanding of **CREDIBILITY** when you were asked if it influenced other concepts?

Very Unclear: <input type="checkbox"/>	Unclear: <input type="checkbox"/>	Clear: <input type="checkbox"/>	Very Clear: <input type="checkbox"/>
--	-----------------------------------	---------------------------------	--------------------------------------

Q: Briefly describe what you took **CREDIBILITY** to mean.

(Please do not use apostrophes ' in your answer)

Next

Section ONE – Short Survey – 5

Given the screenshot below, please answer the following questions;

Faith (5 of 8)

In your opinion, which of the following concepts, if any, does FAITH influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

[Next](#)

Q: How clear was your understanding of **FAITH** when you were asked if it influenced other concepts?

Very Unclear: <input type="checkbox"/>	Unclear: <input type="checkbox"/>	Clear: <input type="checkbox"/>	Very Clear: <input type="checkbox"/>
--	-----------------------------------	---------------------------------	--------------------------------------

Q: Briefly describe what you took **FAITH** to mean.

(Please do not use apostrophes ' in your answer)

[Next](#)

Section ONE – Short Survey – 6

Given the screenshot below, please answer the following questions;

Honesty (6 of 8)

In your opinion, which of the following concepts, if any, does HONESTY influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

[Next](#)

Q: How clear was your understanding of **HONESTY** when you were asked if it influenced other concepts?

Very Unclear: <input type="checkbox"/>	Unclear: <input type="checkbox"/>	Clear: <input type="checkbox"/>	Very Clear: <input type="checkbox"/>
--	-----------------------------------	---------------------------------	--------------------------------------

Q: Briefly describe what you took **HONESTY** to mean.

(Please do not use apostrophes ' in your answer)

[Next](#)

Section ONE – Short Survey – 7

Given the screenshot below, please answer the following questions;

Reliability (7 of 8)

In your opinion, which of the following concepts, if any, does RELIABILITY influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reputation	<input type="checkbox"/>
All	<input type="checkbox"/>

Next

Q: How clear was your understanding of **RELIABILITY** when you were asked if it influenced other concepts?

Very Unclear: <input type="checkbox"/>	Unclear: <input type="checkbox"/>	Clear: <input type="checkbox"/>	Very Clear: <input type="checkbox"/>
--	-----------------------------------	---------------------------------	--------------------------------------

Q: Briefly describe what you took **RELIABILITY** to mean.

(Please do not use apostrophes ' in your answer)

Next

Section ONE – Short Survey – 8

Given the screenshot below, please answer the following questions;

Reputation (8 of 8)

In your opinion, which of the following concepts, if any, does REPUTATION influence:

Belief	<input type="checkbox"/>
Competency	<input type="checkbox"/>
Confidence	<input type="checkbox"/>
Credibility	<input type="checkbox"/>
Faith	<input type="checkbox"/>
Honesty	<input type="checkbox"/>
Reliability	<input type="checkbox"/>
All	<input type="checkbox"/>

Next

Q: How clear was your understanding of **REPUTATION** when you were asked if it influenced other concepts?

Very Unclear: <input type="checkbox"/>	Unclear: <input type="checkbox"/>	Clear: <input type="checkbox"/>	Very Clear: <input type="checkbox"/>
--	-----------------------------------	---------------------------------	--------------------------------------

Q: Briefly describe what you took **REPUTATION** to mean.

(Please do not use apostrophes ' in your answer)

Next

Survey Complete - 1

Thank You

Thank you for taking part in both surveys, Karl.

APPENDIX III – Implementation Code, Trial Data, and Sundry

Implementation Code

All implementation code can be found on the DVD media that accompanies this thesis, which is filed under ‘Implementation Code’, ‘myTrust’. However, the printed appendix III will present a small selection of source code.

The implementation code found in this appendix has been included to illustrate how the MySQL databases are accessed, how Jena operates, and also presents the precise details of the HITS algorithm. In addition, it may also communicate the scale of the development project to the reader.

Jena MySQL to OWL Converter for Personalised Model of Trust

```
import ie.tcd.cs.kdeg.trust.entity.interfaces.Policy;
import ie.tcd.cs.kdeg.trust.entity.interfaces.PolicyHome;
import ie.tcd.cs.kdeg.trust.entity.interfaces.User;
import ie.tcd.cs.kdeg.trust.entity.interfaces.UserHome;
import ie.tcd.cs.kdeg.trust.session.interfaces.PersonaliseManager;
import ie.tcd.cs.kdeg.trust.session.interfaces.PersonaliseManagerHome;
import ie.tcd.cs.kdeg.trust.session.interfaces.cbpmManager;
import ie.tcd.cs.kdeg.trust.session.interfaces.cbpmManagerHome;
import java.io.PrintWriter;
import java.io.IOException;
import java.rmi.RemoteException;
import java.util.Collection;
import java.util.Iterator;
import java.util.Properties;
import javax.ejb.CreateException;
import javax.ejb.EJBException;
import javax.ejb.FinderException;
import javax.naming.Context;
import javax.naming.InitialContext;
import javax.naming.NamingException;
import com.hp.hpl.jena.ontology.AllValuesFromRestriction;
import com.hp.hpl.jena.ontology.DatatypeProperty;
import com.hp.hpl.jena.ontology.HasValueRestriction;
import com.hp.hpl.jena.ontology.Individual;
import com.hp.hpl.jena.ontology.IntersectionClass;
import com.hp.hpl.jena.ontology.ObjectProperty;
import com.hp.hpl.jena.ontology.OntClass;
import com.hp.hpl.jena.ontology.OntDocumentManager;
import com.hp.hpl.jena.ontology.OntModel;
import com.hp.hpl.jena.ontology.OntModelSpec;
import com.hp.hpl.jena.ontology.OntProperty;
```

```

import com.hp.hpl.jena.ontology.ProfileRegistry;
import com.hp.hpl.jena.ontology.Restriction;
import com.hp.hpl.jena.ontology.SomeValuesFromRestriction;
import com.hp.hpl.jena.rdf.model.Model;
import com.hp.hpl.jena.rdf.model.ModelFactory;
import com.hp.hpl.jena.rdf.model.Property;
import com.hp.hpl.jena.rdf.model.RDFNode;
import com.hp.hpl.jena.rdf.model.Resource;
import com.hp.hpl.jena.rdf.model.Statement;
import com.hp.hpl.jena.vocabulary.VCARD;
import com.hp.hpl.jena.vocabulary.XSD;

public class jenaMySQLtoOWLPersonalisedModel {

    public void begin(String userName) throws FinderException, NamingException,
IOException, CreateException
    {
        //mysql code HERE
        String personalisedURI =
"file:C:/PhD/Ontologies/Trust/PersonalisedModel.owl";
        String personalisedNS =
"http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/PersonalisedModel.owl#";
        //create model from policy.owl
        OntModel model = ModelFactory.createOntologyModel();
        OntDocumentManager dm = model.getDocumentManager();
        dm.addAltEntry(personalisedNS, personalisedURI);
        model.read(personalisedNS);

        mysql2owl(model, personalisedURI, personalisedNS, userName);
        model.write(new FileWriter("C:/PhD/Ontologies/Trust/" + userName +
"PersonalisedModel.owl"), "RDF/XML-ABBREV");

        //queryPolicy(personalisedNS, userURI);
    }

    public void mysql2owl(OntModel model, String personalisedURI, String
personalisedNS, String userName) throws IOException, NamingException, CreateException
    {
        String metaURI = "file:C:/PhD/Ontologies/Trust/MetaModel.owl";
        String metaNS =
"http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/MetaModel.owl#";
        //create model from UpperModel.owl
        OntModel modelmeta = ModelFactory.createOntologyModel();
        OntDocumentManager dmmeta = modelmeta.getDocumentManager();
        dmmeta.addAltEntry(metaNS, metaURI);
        modelmeta.read(metaNS);

        String upperURI = "file:C:/PhD/Ontologies/Trust/UpperModel.owl";
        String upperNS =
"http://www.cs.tcd.ie/Karl.Quinn/KDEG/Ontologies/UpperModel.owl#";
        //create model from UpperModel.owl

```

```

OntModel modelupper = ModelFactory.createOntologyModel();
OntDocumentManager dmupper = modelupper.getDocumentManager();
dmupper.addAltEntry(upperNS, upperURI);
modelupper.read(upperNS);

System.out.println("Creating Person");
//Create classes and datatype properties
//Person
OntClass Person = model.getOntClass(personalisedNS + "Person");
DatatypeProperty personURI = model.getDatatypeProperty(personalisedNS +
"personURI");
Individual person = model.createIndividual(personalisedNS + userName,
Person);

System.out.println(personURI + userName);
person.addProperty(personURI, userName);
ObjectProperty hasPersonalisedModel =
model.getObjectProperty(personalisedNS + "hasPersonalisedModel");

System.out.println("Creating Personalised Model");
//PersonalisedModel Creation
OntClass PersonalisedModel = model.getOntClass(personalisedNS +
"PersonalisedModel");
Individual personalisedIndividual =
model.createIndividual(personalisedNS + "default", PersonalisedModel);

//name the PersonalisedModel
DatatypeProperty personalisedName =
model.getDatatypeProperty(personalisedNS + "personalisedModelName");
personalisedIndividual.addProperty(personalisedName, "default");

//Create a hasConcept ObjectProperty
ObjectProperty hasConcept = model.getObjectProperty(personalisedNS +
"hasConcept");

System.out.println("Creating the Ranks and Weights");
//Concepts (Rank and Weight)
Individual Belief = addHasConcept("Belief", model, modelupper,
personalisedIndividual, hasConcept, upperNS, personalisedNS, userName);
Individual Competency = addHasConcept("Competency", model, modelupper,
personalisedIndividual, hasConcept, upperNS, personalisedNS, userName);
Individual Confidence = addHasConcept("Confidence", model, modelupper,
personalisedIndividual, hasConcept, upperNS, personalisedNS, userName);
Individual Credibility = addHasConcept("Credibility", model, modelupper,
personalisedIndividual, hasConcept, upperNS, personalisedNS, userName);
Individual Faith = addHasConcept("Faith", model, modelupper,
personalisedIndividual, hasConcept, upperNS, personalisedNS, userName);
Individual Honesty = addHasConcept("Honesty", model, modelupper,
personalisedIndividual, hasConcept, upperNS, personalisedNS, userName);
Individual Reputation = addHasConcept("Reputation", model, modelupper,
personalisedIndividual, hasConcept, upperNS, personalisedNS, userName);
Individual Reliability = addHasConcept("Reliability", model, modelupper,
personalisedIndividual, hasConcept, upperNS, personalisedNS, userName);

```

```

//Concepts (Interwined Relationships)
ObjectProperty hasAffectedBy = modelmeta.getObjectProperty(metaNS +
"hasAffectedBy");
ObjectProperty hasInformedBy = modelmeta.getObjectProperty(metaNS +
"hasInformedBy");
ObjectProperty hasDerivedFrom = modelmeta.getObjectProperty(metaNS +
"hasDerivedFrom");
String concept = "";
System.out.println("Creating the Relationships");
//find relationships
conceptRelationships(userName, "reputation", Reputation, Reputation,
Reliability, Belief, Credibility, Honesty, Faith, Confidence, Competency,
hasAffectedBy, hasInformedBy, hasDerivedFrom);
conceptRelationships(userName, "reliability", Reliability, Reputation,
Reliability, Belief, Credibility, Honesty, Faith, Confidence, Competency,
hasAffectedBy, hasInformedBy, hasDerivedFrom);
conceptRelationships(userName, "belief", Belief, Reputation,
Reliability, Belief, Credibility, Honesty, Faith, Confidence, Competency,
hasAffectedBy, hasInformedBy, hasDerivedFrom);
conceptRelationships(userName, "credibility", Credibility, Reputation,
Reliability, Belief, Credibility, Honesty, Faith, Confidence, Competency,
hasAffectedBy, hasInformedBy, hasDerivedFrom);

conceptRelationships(userName, "honesty", Honesty, Reputation,
Reliability, Belief, Credibility, Honesty, Faith, Confidence, Competency,
hasAffectedBy, hasInformedBy, hasDerivedFrom);
conceptRelationships(userName, "faith", Faith, Reputation, Reliability,
Belief, Credibility, Honesty, Faith, Confidence, Competency, hasAffectedBy,
hasInformedBy, hasDerivedFrom);
conceptRelationships(userName, "confidence", Confidence, Reputation,
Reliability, Belief, Credibility, Honesty, Faith, Confidence, Competency,
hasAffectedBy, hasInformedBy, hasDerivedFrom);
conceptRelationships(userName, "competency", Competency, Reputation,
Reliability, Belief, Credibility, Honesty, Faith, Confidence, Competency,
hasAffectedBy, hasInformedBy, hasDerivedFrom);
System.out.println("Adding Relationships");
//Add hasConcepts
personalisedIndividual.addProperty(hasConcept, Belief);
personalisedIndividual.addProperty(hasConcept, Competency);
personalisedIndividual.addProperty(hasConcept, Confidence);
personalisedIndividual.addProperty(hasConcept, Credibility);
personalisedIndividual.addProperty(hasConcept, Faith);
personalisedIndividual.addProperty(hasConcept, Honesty);
personalisedIndividual.addProperty(hasConcept, Reputation);
personalisedIndividual.addProperty(hasConcept, Reliability);
System.out.println("Finalising Personalised Model");
//link person with personalised model
person.addProperty(hasPersonalisedModel, personalisedIndividual);
System.out.println("Finished");
//TEST HERE WITH QUERY.
}

```

```

private void conceptRelationships(String userName, String c, Individual Concept,
Individual reputation, Individual reliability, Individual belief, Individual
credibility, Individual honesty, Individual faith, Individual confidence, Individual
competency, ObjectProperty hasAffectedBy, ObjectProperty hasInformedBy, ObjectProperty
hasDerivedFrom) {
    if(c.equals("faith") || c.equals("belief"))
    {
        if(findRelationships(userName, c,
"belief")==true){Concept.addProperty(hasAffectedBy, belief);System.out.println(c + ", "
+ "belief");};
        if(findRelationships(userName, c,
"faith")==true){Concept.addProperty(hasAffectedBy, faith);System.out.println(c + ", " +
"faith");};

        if(findRelationships(userName, c,
"reputation")==true){Concept.addProperty(hasInformedBy, reputation);
System.out.println(c + ", " + "reputation");};
        if(findRelationships(userName, c,
"reliability")==true){Concept.addProperty(hasInformedBy,
reliability);System.out.println(c + ", " + "reliability");};
        if(findRelationships(userName, c,
"credibility")==true){Concept.addProperty(hasInformedBy,
credibility);System.out.println(c + ", " + "credibility");};
        if(findRelationships(userName, c,
"honesty")==true){Concept.addProperty(hasInformedBy, honesty);System.out.println(c + ",
" + "honesty");};
        if(findRelationships(userName, c,
"confidence")==true){Concept.addProperty(hasInformedBy,
confidence);System.out.println(c + ", " + "confidence");};
        if(findRelationships(userName, c,
"competency")==true){Concept.addProperty(hasInformedBy,
competency);System.out.println(c + ", " + "competency");};
    }

    if(c.equals("reputation") || c.equals("reputation") ||
c.equals("credibility") || c.equals("honesty") || c.equals("confidence") ||
c.equals("competency"))
    {
        if(findRelationships(userName, c,
"belief")==true){Concept.addProperty(hasInformedBy, belief);System.out.println(c + ", "
+ "belief");};
        if(findRelationships(userName, c,
"faith")==true){Concept.addProperty(hasInformedBy, faith);System.out.println(c + ", " +
"faith");};

        if(findRelationships(userName, c,
"reputation")==true){Concept.addProperty(hasDerivedFrom, reputation);
System.out.println(c + ", " + "reputation");};
        if(findRelationships(userName, c,
"reliability")==true){Concept.addProperty(hasDerivedFrom,
reliability);System.out.println(c + ", " + "reliability");};
    }
}

```

```

        if(findRelationships(userName, c,
"credibility")==true){Concept.addProperty(hasDerivedFrom,
credibility);System.out.println(c + ", " + "credibility");};
        if(findRelationships(userName, c,
"honesty")==true){Concept.addProperty(hasDerivedFrom, honesty);System.out.println(c +
", " + "honesty");};
        if(findRelationships(userName, c,
"confidence")==true){Concept.addProperty(hasDerivedFrom,
confidence);System.out.println(c + ", " + "confidence");};
        if(findRelationships(userName, c,
"competency")==true){Concept.addProperty(hasDerivedFrom,
competency);System.out.println(c + ", " + "competency");};
    }
}
private boolean findRelationships(String userName, String conceptA, String
conceptB) {
    boolean influence = false;
    try{
        Properties properties = new Properties();
        properties.put("java.naming.factory.initial",
"org.jnp.interfaces.NamingContextFactory");
        properties.put("java.naming.factory.url.pkgs",
"org.jboss.naming:org.jnp.interfaces");
        properties.put("java.naming.provider.url",
"jnp://localhost:1099");
        properties.put("jnp.disableDiscovery", "true");

        Context context;
        context = new InitialContext(properties);
        Object Pobject = context.lookup(cbpmManagerHome.JNDI_NAME);

        cbpmManagerHome cbpmManagerHome = (cbpmManagerHome)
javax.rmi.PortableRemoteObject.narrow(Pobject, cbpmManagerHome.class);
        cbpmManager cbpmManager = cbpmManagerHome.create();

        int inf = cbpmManager.getInfluenceByUserSourceConcept(userName,
conceptA, conceptB);

        if(inf==1)
        {influence=true;};

        System.out.println("Checking relationship: " + conceptA + ", " +
conceptB + ": " + inf);
        //if(cbpmManager.getInfluenceByUserSourceConcept(userName, conceptA,
conceptB)==0){influence=false;};

    }catch (CreateException e1) {e1.printStackTrace();}
    }catch (RemoteException e1) {e1.printStackTrace();}
    }catch (EJBException e) {e.printStackTrace();}
    }catch (NamingException e) {e.printStackTrace();}
}

```

```

        return influence;
    }

    private Individual addHasConcept(String concept, OntModel model, OntModel
modelu, Individual personalisedIndividual, ObjectProperty hasConcept, String upperNS,
String personalisedNS, String userName) {
        OntClass C = modelu.getOntClass(upperNS + concept);
        Individual c = model.createIndividual(personalisedNS + concept, C);

        try{
            Properties properties = new Properties();
            properties.put("java.naming.factory.initial",
"org.jnp.interfaces.NamingContextFactory");
            properties.put("java.naming.factory.url.pkgs",
"org.jboss.naming:org.jnp.interfaces");
            properties.put("java.naming.provider.url",
"jnp://localhost:1099");
            properties.put("jnp.disableDiscovery", "true");

            Context context;
            context = new InitialContext(properties);

            Object                Pobject                =
context.lookup(PersonaliseManagerHome.JNDI_NAME);

            PersonaliseManagerHome personaliseManagerHome = (PersonaliseManagerHome)
javax.rmi.PortableRemoteObject.narrow(Pobject, PersonaliseManagerHome.class);
            PersonaliseManager personaliseManager = personaliseManagerHome.create();
            int                conceptRank                =
personaliseManager.getRankByUserSourceConcept(userName, concept);
            float                conceptWeight                =
personaliseManager.getWeightByUserSourceConcept(userName, concept);

            DatatypeProperty beliefRank = model.getDatatypeProperty(personalisedNS +
"hasRank");
            c.addProperty(beliefRank, conceptRank);
            DatatypeProperty beliefWeight = model.getDatatypeProperty(personalisedNS
+ "hasWeight");
            c.addProperty(beliefWeight, conceptWeight);
        }catch (CreateException e1) {e1.printStackTrace();}
        }catch (RemoteException e1) {e1.printStackTrace();}
        }catch (EJBException e) {e.printStackTrace();}
        }catch (NamingException e) {e.printStackTrace();}

        return c;
    }

    private void queryPolicy(String policyNS, String userURI)
    {
        String policyURI = "file:C:/PhD/Ontologies/Trust/newPolicy.owl";
    }

```



```

policyNS = "http://www.cs.tcd.ie/Karl.Quinn/policy.owl#";

//create model from policy.owl
OntModel newmodel = ModelFactory.createOntologyModel();
OntDocumentManager newdm = newmodel.getDocumentManager();
newdm.addAltEntry(policyNS, policyURI);
newmodel.read(policyNS);

Individual ind= newmodel.getIndividual(policyNS + userURI);
//Iterator it = ind.listProperties();

DatatypeProperty   puri   =   newmodel.getDatatypeProperty(policyNS   +
"personURI");
ObjectProperty     hasPolicy =   newmodel.getObjectProperty(policyNS   +
"hasPolicy");

//if the userURI exists...
if (ind.hasProperty(puri))
    {
        System.out.println("\n-- INDIVIDUAL DATA --");
        System.out.println("-PERSON-");
        //personURI
        Statement st = ind.getProperty(puri);
        Object ob = (Object)st.getObject();
        System.out.println("personURI: " + ob.toString());

        //hasPolicy
        Iterator it = ind.listProperties(hasPolicy);
        String selectedPolicy = "";
        int policyNumber = 0;
        while(it.hasNext())
            {
                policyNumber++;

//System.out.println(ind.getProperty(hasPolicy).getObject().toString());

                Object inter = (Object) it.next();
                selectedPolicy = policySelector(inter.toString());

//Print policy
                System.out.println("\n-----");
                System.out.println("-POLICY #" + policyNumber + "-");
                Individual Policy= newmodel.getIndividual(selectedPolicy);
                DatatypeProperty   policyName   =
newmodel.getDatatypeProperty(policyNS + "policyName");
                if (Policy.hasProperty(policyName))
                    {
                        //policyName
                        st = Policy.getProperty(policyName);
                        ob = (Object)st.getObject();

```

```

        System.out.println("policyName: " + ob.toString());

        //hasEvent
        System.out.println(" -EVENT-");
        ObjectProperty          hasEvent          =
newmodel.getObjectProperty(policyNS + "hasEvent");
        st = Policy.getProperty(hasEvent);
        Object eventObject = (Object)st.getObject();
        //System.out.println("          hasEvent:    "    +
eventObject.toString());
        Individual              Event=
newmodel.getIndividual(eventObject.toString());
        DatatypeProperty        eventName        =
newmodel.getDatatypeProperty(policyNS + "eventName");
        if (Event.hasProperty(eventName))
        {
            st = Event.getProperty(eventName);
            ob = (Object)st.getObject();
            System.out.println("          eventName:    "    +
ob.toString());
        }

        //hasCondition
        System.out.println(" -CONDITION-");
        ObjectProperty          hasCondition      =
newmodel.getObjectProperty(policyNS + "hasCondition");
        st = Policy.getProperty(hasCondition);
        Object conditionObject = (Object)st.getObject();
        //System.out.println("          hasCondition:    "    +
conditionObject.toString());
        Individual              Condition=
newmodel.getIndividual(conditionObject.toString());
        DatatypeProperty        trustValue       =
newmodel.getDatatypeProperty(policyNS + "trustValue");
        if (Condition.hasProperty(trustValue))
        {
            st = Condition.getProperty(trustValue);
            ob = (Object)st.getObject();
            System.out.println("          trustValue:    "    +
ob.toString());
        }
        System.out.println(" -ADVANCED CONDITIONS-");
        ObjectProperty          hasAdvancedConditions =
newmodel.getObjectProperty(policyNS + "hasAdvancedConditions");
        st = Condition.getProperty(hasAdvancedConditions);
        Object advancedConditionObject = (Object)st.getObject();
        Individual              advancedConditions =
newmodel.getIndividual(advancedConditionObject.toString());
        //System.out.println("          advancedConditions:    "    +
advancedConditionObject.toString());

```

```

        DatatypeProperty      reputationValue      =
newmodel.getDatatypeProperty(policyNS + "reputationValue");
        DatatypeProperty      competencyValue     =
newmodel.getDatatypeProperty(policyNS + "competencyValue");
        DatatypeProperty      reliabilityValue     =
newmodel.getDatatypeProperty(policyNS + "reliabilityValue");
        DatatypeProperty      credibilityValue     =
newmodel.getDatatypeProperty(policyNS + "credibilityValue");
        DatatypeProperty      honestyValue         =
newmodel.getDatatypeProperty(policyNS + "honestyValue");
        DatatypeProperty      beliefValue         =
newmodel.getDatatypeProperty(policyNS + "beliefValue");
        DatatypeProperty      faithValue          =
newmodel.getDatatypeProperty(policyNS + "faithValue");
        DatatypeProperty      confidenceValue      =
newmodel.getDatatypeProperty(policyNS + "confidenceValue");

        //
        st = advancedConditions.getProperty(reputationValue);
        ob = (Object)st.getObject();
        if(ob.toString().equals("not
used")!=true){System.out.println("      reputationValue: " + ob.toString());}

        st = advancedConditions.getProperty(reliabilityValue);
        ob = (Object)st.getObject();
        if(ob.toString().equals("not
used")!=true){System.out.println("      reliabilityValue: " + ob.toString());}

        st = advancedConditions.getProperty(competencyValue);
        ob = (Object)st.getObject();
        if(ob.toString().equals("not
used")!=true){System.out.println("      competencyValue: " + ob.toString());}

        st = advancedConditions.getProperty(credibilityValue);
        ob = (Object)st.getObject();
        if(ob.toString().equals("not
used")!=true){System.out.println("      credibilityValue: " + ob.toString());}

        st = advancedConditions.getProperty(honestyValue);
        ob = (Object)st.getObject();
        if(ob.toString().equals("not
used")!=true){System.out.println("      honestyValue: " + ob.toString());}

        st = advancedConditions.getProperty(beliefValue);
        ob = (Object)st.getObject();
        if(ob.toString().equals("not
used")!=true){System.out.println("      beliefValue: " + ob.toString());}

        st = advancedConditions.getProperty(faithValue);
        ob = (Object)st.getObject();

```

```

        if (ob.toString().equals("not
used") != true) {System.out.println("    faithValue: " + ob.toString());}

        st = advancedConditions.getProperty(confidenceValue);
        ob = (Object)st.getObject();
        if (ob.toString().equals("not
used") != true) {System.out.println("    confidenceValue: " + ob.toString());}

        //hasAction
        System.out.println("    -ACTION-");
        ObjectProperty          hasAction          =
newmodel.getObjectProperty(policyNS + "hasAction");
        st = Policy.getProperty(hasAction);
        Object actionObject = (Object)st.getObject();
        //System.out.println("    hasAction:    "    +
actionObject.toString());
        Individual              Action=
newmodel.getIndividual(actionObject.toString());
        DatatypeProperty          action          =
newmodel.getDatatypeProperty(policyNS + "grant");
        if (Action.hasProperty(action))
        {
            st = Action.getProperty(action);
            ob = (Object)st.getObject();
            System.out.println("    grant: " + ob.toString());
        }
        System.out.println("-----");
    }
}
}

private String policySelector(String inter) {
    String selected = "";
    int comma = inter.lastIndexOf(",");
    comma += 2;
    int len = inter.length() - 1;
    selected = inter.substring(comma, len);
    return selected;
}
}

```

HITS Algorithm for Generating Personalised Models of Trust

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.Vector;
import Jama.Matrix;

public class AuthorityHubRank {
    private Vector ranked = new Vector();
    private int x, y = 0;

    public AuthorityHubRank(){}

    public Vector Rank(String userSource, Connection con, float inspiredBy, float
assistedBy, float derivedFrom) throws InstantiationException, IllegalAccessException,
ClassNotFoundException, SQLException{

        int numberOfConcepts = 8;
        int iter = 3;
        double[][] array = new double[numberOfConcepts][numberOfConcepts];

        for(x=0; x < numberOfConcepts; x++)
            {
                for(y=0; y < numberOfConcepts; y++)
                    {
                        array[x][y] = 0.0;
                    }
            }

        //TODO
        //take from DB and insert into array

        //The newInstance() call is a work around for some broken Java implementations
        Class.forName("com.mysql.jdbc.Driver").newInstance();

        Statement stmt = con.createStatement();

        String query = "select * from trustmodel where userSource = '"+userSource+"'";
        ResultSet rs = stmt.executeQuery(query);
        boolean go = true;

        /*
        while (rs.next()) {
            int rank = rs.getInt("personalisedRank");
            if(rank!=0)
                {
                    go = false;
                    //System.out.println("personalisedRank != 0");
                }
        }
        */
    }
}
```

```

        }
        else{
            //System.out.println("personalisedRank == 0");
        }
    }

    */
    if(go==true)
    {
        x=0;
        y=0;

        query = "Select * from trustmodel where userSource = '"+userSource+"' and
concept = 'reputation'";
        retrieve(stmt, query, array, x, y);
        x++;

        query = "Select * from trustmodel where userSource = '"+userSource+"' and
concept = 'reliability'";
        retrieve(stmt, query, array, x, y);
        x++;

        query = "Select * from trustmodel where userSource = '"+userSource+"' and
concept = 'competency'";
        retrieve(stmt, query, array, x, y);
        x++;

        query = "Select * from trustmodel where userSource = '"+userSource+"' and
concept = 'credibility'";
        retrieve(stmt, query, array, x, y);
        x++;

        query = "Select * from trustmodel where userSource = '"+userSource+"' and
concept = 'honesty'";
        retrieve(stmt, query, array, x, y);
        x++;

        query = "Select * from trustmodel where userSource = '"+userSource+"' and
concept = 'belief'";
        retrieve(stmt, query, array, x, y);
        x++;

        query = "Select * from trustmodel where userSource = '"+userSource+"' and
concept = 'faith'";
        retrieve(stmt, query, array, x, y);
        x++;

        query = "Select * from trustmodel where userSource = '"+userSource+"' and
concept = 'confidence'";
        retrieve(stmt, query, array, x, y);
        x++;
    }
}

```

```

//legacy 0 to 1, or 1 to 0,
x=1;
y=1;

//personalised algorithm
Matrix A = new Matrix(array);

//survey reversal; that are influenced by -> influences
A = A.transpose();

for (int z=0; z<1; z++)
{
Matrix Atrans = A.transpose();

//Hubs diagonal
Matrix AAt = A.times(Atrans);

//Authorities diagonal
Matrix AtA = Atrans.times(A);

Matrix test = hubRank(A, AAt, numberOfConcepts, iter);
Matrix testtwo = authorityRank(AAt, A, AtA, numberOfConcepts, iter);

double lowest = 1.0;
for(int j=0; j < numberOfConcepts; j++)
{
if(test.get(j,0) < lowest && test.get(j,0) > 0.0)
lowest = test.get(j,0);
}

for(int j=0; j < numberOfConcepts; j++)
{
if(testtwo.get(j,0) <= lowest && testtwo.get(j,0) > 0.0)
lowest = testtwo.get(j,0);
}

for(int j=0; j < numberOfConcepts; j++)
{
double normH = test.get(j,j)*(1/lowest);
int k = j+1;
//System.out.print("[ " + j + " ] " + normH + " \n");
test.set(j,j,normH);
}

for(int j=0; j < numberOfConcepts; j++)
{
double normA = testtwo.get(j,j)*(1/lowest);
int k = j+1;
testtwo.set(j,j,normA);
}

```

```

//Iterate through concepts and apply weights to hubs and authorities
Matrix hubValues = test.copy();
Matrix authorityValues = testtwo.copy();

Matrix hubMarker = A.copy();
Matrix authorityMarker = Atrans.copy();

Matrix conceptsRanked = rankConcepts(hubMarker, hubValues, authorityMarker,
authorityValues, numberOfConcepts, inspiredBy, assistedBy, derivedFrom);

for(int i=0; i<8; i++)
    {
    System.out.println();
    for(int j=0; j<8; j++)
        {
        System.out.print(conceptsRanked.get(i,j) + " ");
        }
    }

//update db with personalised ranks.
int diag = 0;
for(int j=0; j < numberOfConcepts; j++)
    {
    int k = j+1;

    if(j==0)
    {
    //System.out.print("Belief - " + conceptsRanked.get(diag,diag) + " \n");
    String insert = "Update trustmodel set personalisedRank =
'" + conceptsRanked.get(diag,diag) + "' where userSource = '" + userSource + "' and concept =
'reputation'";
    stmt.executeUpdate(insert);
    diag++;
    }

    if(j==1)
    {
    //System.out.print("Competency - " + conceptsRanked.get(diag,diag) + "
\n");
    String insert = "Update trustmodel set personalisedRank =
'" + conceptsRanked.get(diag,diag) + "' where userSource = '" + userSource + "' and concept =
'reliability'";
    stmt.executeUpdate(insert);
    diag++;
    }

    if(j==2)
    {
    //System.out.print("Confidence - " + conceptsRanked.get(diag,diag) + "
\n");

```



```

        String insert = "Update trustmodel set personalisedRank =
'" + conceptsRanked.get(diag,diag) + "'where userSource = '" + userSource + "' and concept =
'competency'";
        stmt.executeUpdate(insert);
        diag++;
    }

    if(j==3)
    {
        //System.out.print("Credibility - " + conceptsRanked.get(diag,diag) + "
\n");
        String insert = "Update trustmodel set personalisedRank =
'" + conceptsRanked.get(diag,diag) + "'where userSource = '" + userSource + "' and concept =
'credibility'";
        stmt.executeUpdate(insert);
        diag++;
    }

    if(j==4)
    {
        //System.out.print("Faith - " + conceptsRanked.get(diag,diag) + " \n");
        String insert = "Update trustmodel set personalisedRank =
'" + conceptsRanked.get(diag,diag) + "'where userSource = '" + userSource + "' and concept =
'honesty'";
        stmt.executeUpdate(insert);
        diag++;
    }

    if(j==5)
    {
        //System.out.print("Honesty - " + conceptsRanked.get(diag,diag) + "
\n");
        String insert = "Update trustmodel set personalisedRank =
'" + conceptsRanked.get(diag,diag) + "'where userSource = '" + userSource + "' and concept =
'belief'";
        stmt.executeUpdate(insert);
        diag++;
    }

    if(j==6)
    {
        //System.out.print("Reliability - " + conceptsRanked.get(diag,diag) + "
\n");
        String insert = "Update trustmodel set personalisedRank =
'" + conceptsRanked.get(diag,diag) + "'where userSource = '" + userSource + "' and concept =
'faith'";
        stmt.executeUpdate(insert);
        diag++;
    }

    if(j==7)

```

```

        {
            //System.out.print("Reputation - " + conceptsRanked.get(diag,diag) + "
\n");
            String insert = "Update trustmodel set personalisedRank =
'" + conceptsRanked.get(diag,diag) + "' where userSource = '" + userSource + "' and concept =
'confidence'";
            stmt.executeUpdate(insert);
            diag++;
        }
    }
}
return ranked;
}

private void retrieve(Statement stmt, String query, double[][] array, int a, int b)
throws SQLException {
    // TODO Auto-generated method stub
    ResultSet rs = stmt.executeQuery(query);

    while (rs.next()) {
        String concept = rs.getString("concept");
        int belief = rs.getInt("belief");
        int confidence = rs.getInt("confidence");
        int competency = rs.getInt("competency");
        int credibility = rs.getInt("credibility");

        int faith = rs.getInt("faith");
        int honesty = rs.getInt("honesty");
        int reliability = rs.getInt("reliability");
        int reputation = rs.getInt("reputation");
        System.out.println(concept + " - " + reputation + " " + reliability + " "
+ competency + " " + credibility + " " + honesty + " " + belief + " " + faith + " " +
confidence);

        array[a][b] = reputation ;
        b++;
        array[a][b] = reliability ;
        b++;
        array[a][b] = competency ;
        b++;
        array[a][b] = credibility;
        b++;

        array[a][b] = honesty;
        b++;
        array[a][b] = belief ;
        b++;
        array[a][b] = faith ;
        b++;
        array[a][b] = confidence;
    }
}

```

```

        b++;

    }

}

private Matrix rankConcepts(Matrix hubMarker, Matrix hubValues, Matrix
authorityMarker, Matrix authorityValues, int numberOfConcepts, float inspiredBy, float
assistedBy, float derivedFrom)
{

double[][] rankArray = new double[numberOfConcepts+1][numberOfConcepts+1];

for(x=0; x <= numberOfConcepts; x++)
{
for(y=0; y <= numberOfConcepts; y++)
{
rankArray[x][y] = 0.0;
}
}

Matrix RA = new Matrix(rankArray);

//HUB
for(int i=0; i < numberOfConcepts; i++)
{
double immediateValue = hubValues.get(i,i);
System.out.println("immediateValue: " + immediateValue);
int diag = 0;

for(int j=0; j < numberOfConcepts; j++)
{
//cycle thru hub marker
if(hubMarker.get(i,j) == 1)
{
double val = RA.get(diag,diag);

val = metamodel(i, j, val, immediateValue,inspiredBy,
assistedBy, derivedFrom);

//val = val + immediateValue;
System.out.println("Hub Marker: ["+i+"] ["+j+"] -> " +
val);

RA.set(diag,diag,val);
}
diag++;
}
}

//AUTHORITY
for(int i=0; i < numberOfConcepts; i++)
{
double immediateValue = authorityValues.get(i,i);

```

```

System.out.println("immediateValue: " + immediateValue);
int diag = 0;

for(int j=0; j < numberOfConcepts; j++)
{
    //cycle thru hub marker
    if(authorityMarker.get(i,j) == 1)
        {
            double val = RA.get(diag,diag);
            val = metamodel(i, j, val, immediateValue, inspiredBy,
assistedBy, derivedFrom);
            //val = val + immediateValue;
            System.out.println("Authority Marker: ["+i+"] ["+j+"] ->
" + val);

            RA.set(diag,diag,val);
        }
    diag++;
}

return RA;
}

private double metamodel(int i, int j, double val, double immediateValue, float
inspiredBy, float assistedBy, float derivedFrom)
{
    if(i == j)
    {
        val = val + immediateValue;
    }
    else
    {
        //derivedFrom
        if(i<=4 && j <=4)
            {val = val + (immediateValue*derivedFrom);}

        //assistedBy
        if(i<=4 && j >=5)
            {val = val + (immediateValue*assistedBy);}
        if(i>=5 && j <=4)
            {val = val + (immediateValue*assistedBy);}

        //inspiredBy
        if(i>=5 && j >=5)
            {val = val + (immediateValue*inspiredBy);}
    }

    return val;
}

public Matrix hubRank(Matrix A, Matrix AAt,int numberOfConcepts, int iterations)

```

```

        {
Matrix test = A.times(AAt);
test = AAt.times(test);
test = AAt.times(test);

double[][] returnArray = new double[numberOfConcepts][numberOfConcepts];

double sum=0.0;
for(int i=0; i < numberOfConcepts; i++)
    {
        for(int j=0; j < numberOfConcepts; j++)
            {
                sum += test.get(i,j)*test.get(i,j);
                returnArray[i][j] = 0.0;
            }
    }
Matrix hubRank = new Matrix(returnArray);

sum = Math.sqrt(sum);

//System.out.println("HUBS:");
for(int i=0; i < numberOfConcepts; i++)
    {
        for(int j=0; j < numberOfConcepts; j++)
            {
                test.set(i,j,test.get(i,j)/sum);
                //System.out.print("" + test.get(i,j) + " - ");
            }
        //System.out.println();
    }

//System.out.print("-----\n");
//System.out.print("          HUB Finder          \n");
//System.out.print("-----\n");
//
int concept = 0;

for(int i=0; i < numberOfConcepts; i++)
    {
        boolean stop = false;
        //System.out.println("\nTrust Concept Cycle: " + concept);

        for(int j=0; j < numberOfConcepts; j++)
            {
                //System.out.println("j: " + j + " concept " + concept);
                if(test.get(concept,j) > 0.0 && stop == false)
                    {
                        //System.out.println("Concept [" + j + "][" + concept +
" ] = " + test.get(concept,j));
                        hubRank.set(concept,concept,test.get(concept,j));
                        stop = true;
                    }
            }
    }

```

```

        }
    }
    concept++;
}
//System.out.println("-----\n");

return hubRank;
}

public Matrix authorityRank(Matrix AAt, Matrix Atrans, Matrix AtA, int
numberOfConcepts, int iterations)
{
double[][] returnArray = new double[numberOfConcepts][numberOfConcepts];
Matrix testtwo = Atrans.times(AAt);
testtwo = AtA.times(testtwo);
testtwo = AtA.times(testtwo);

double sum=0.0;
for(int i=0; i < numberOfConcepts; i++)
{
for(int j=0; j < numberOfConcepts; j++)
{
sum += testtwo.get(i,j)*testtwo.get(i,j);
returnArray[i][j] = 0.0;
}
}
Matrix authRank = new Matrix(returnArray);

sum = Math.sqrt(sum);
//System.out.print("SUM ROOT: " + sum + " \n");
//System.out.println("AUTHORITIES: ");
for(int i=0; i < numberOfConcepts; i++)
{
for(int j=0; j < numberOfConcepts; j++)
{
testtwo.set(i,j,testtwo.get(i,j)/sum);
//System.out.print(" " + testtwo.get(i,j) + " - ");
}
}
//System.out.println();
}
//System.out.print("-----\n");
//System.out.print("          AUTH Finder          \n");
//System.out.print("-----\n");

int concept = 0;

for(int i=0; i < numberOfConcepts; i++)
{

```

```

        boolean stop = false;
        //System.out.println("\nTrust Concept Cycle: " + concept);

        for(int j=0; j < numberOfConcepts; j++)
            {
                //System.out.println("j: " + j + " concept " + concept);
                if(testtwo.get(concept,j) > 0.0 && stop == false)
                    {
                        //System.out.println("Concept [" + j + "][" + concept +
                        "]" = " + testtwo.get(concept,j));
                        authRank.set(concept,concept,testtwo.get(concept,j));
                        stop = true;
                    }
                }
            concept++;
        }
        //System.out.println("-----\n");

        return authRank;
    }

    public class Concept{
        String conceptName = "";
        Vector inLinks = new Vector();
        Vector outLinks = new Vector();

        double hubValue = 1.0;
        double authorityValue = 1.0;
    }
}

```

Trial Data

All trial data can be found on the DVD media that accompanies this thesis. Trial data for testing the integration of *myTrust*, the Community Based Policy Management (CBPM) system, and the PUDECAS ubiquitous computing simulator is filed under ‘Trial Data’, ‘Trial One’. Trial data for testing the integration of *myTrust*, the Community Based Policy Management (CBPM) system, the Instant Messaging (IM) system, and the PUDECAS ubiquitous computing simulator is filed under ‘Trial Data’, ‘Trial Two’.

The trial data has been included on the DVD media in order to provide a full and comprehensive store of all data used in the trials presented in this Ph.D. thesis.

Sundry

Only an explanation of HITS algorithm is provided in sundry in this appendix. This explanation may serve to inform the reader of how the algorithm operates.

HITS Algorithm

Kleinberg's Hypertext Induced Topic Selection (HITS) algorithm is used to rank web pages. The HITS algorithm is used to find web pages specific to a given topic, such as cars, cameras, etc. An *authority* is a web page that offers information on a specific topic. A *hub* is a web page that provides a URL link to a web page that offers information on a specific topic. Therefore, an *authority* is a source of content, whereas a *hub* is a source of links. Each web page has an *authority* value and a *hub* value.

These values are calculated using the following steps;

- (1) Use query terms to retrieve a root set of web pages, approximately 200.
- (2) Create a base set S by adding all web pages the root set links too, approximately 1000.
- (3) Associate non-negative authority weights a_p and hub weights h_p to each web page.
- (4) These weights can be updated as follows:

$$a_p = \sum_{q \in S | q \rightarrow p} h_q \qquad h_p = \sum_{q \in S | q \leftarrow p} a_q$$

- (5) Introduce an adjacency matrix \mathbf{A}
 - $\mathbf{A}(\mathbf{i}, \mathbf{j}) = \mathbf{1}$ if web page i links to web page j .
 - The *authority* and *hub* weight vectors are:
$$h = \mathbf{A} \cdot a; a = \mathbf{A}^T \cdot h$$
$$h = (\mathbf{A}\mathbf{A}^T)^k h; a = (\mathbf{A}^T\mathbf{A})^k a$$
 - h can be initialised with a random value.
 - According to linear algebra these two equations converge to the principle eigenvectors of $\mathbf{A}\mathbf{A}^T$ and $\mathbf{A}^T\mathbf{A}$ respectively.

Note: Special thanks to Professor Pdraig Cunningham, Computer Science, Trinity College Dublin for the use of his lecture notes used in the creation of this explanation.