

MODELLING AND COMPUTING THE QUALITY OF SCIENTIFIC INFORMATION ON THE WEB OF DATA

A THESIS SUBMITTED TO THE UNIVERSITY OF MANCHESTER
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
IN THE FACULTY OF ENGINEERING AND PHYSICAL SCIENCES

2014

By
Matthew Gamble
School of Computer Science

Contents

List of Tables	7
List of Figures	11
List of Code Listings	13
Abstract	14
Declaration	15
Copyright	16
Table of Artefacts	21
Acknowledgements	24
1 Introduction	26
1.1 Problem	26
1.1.1 A Motivating Question	30
1.1.2 Quality Knowledge and the Information Quality Life-cycle	31
1.2 Examples of Quality Knowledge for Scientific Data on the Web . .	32
1.2.1 Minimum Information Checklists	32
1.2.2 Gene Ontology Annotations Quality Score	34
1.2.3 Online Chemical Structure Repositories	36
1.3 Research Aims and Objectives	37
1.4 Objective, Predictive, and Subjective	37
1.5 Research Hypotheses	39
1.6 Methodology and Approach	40
1.7 Research Contributions	42

1.8	Publications and Research Activity	44
1.8.1	Publications	45
1.8.2	Research Activity	46
1.9	Thesis Organization	46
2	Background	48
2.1	Chapter Introduction	48
2.2	What is Data Quality?	49
2.2.1	The IQ Life-Cycle	53
2.3	Quality Knowledge and the Web of Data	58
2.3.1	The Linked Data Approach	58
2.3.2	Examining Quality Knowledge	63
2.4	OPS IQ: an Objective, Predictive and Subjective Classification . .	69
2.4.1	Using the Classification	72
2.5	The Role of Provenance	75
2.5.1	What is Provenance?	75
2.5.2	State of Provenance in the Web of Data	78
2.6	Related Work in Information Quality Assessment	82
2.7	Summary and Conclusions	95
3	MIM: a Minimum Information Model	97
3.1	Chapter Introduction	97
3.2	Minimum Information Checklists	98
3.3	A WikiProject Chemicals Case-Study	102
3.3.1	Extending the IQ Life-cycle	107
3.3.2	chemmim: A Checklist for Chemical Compound Data . . .	108
3.4	Minimum Information Checklist Structural Meta-Study	112
3.5	The MIM Vocabulary	115
3.5.1	Describing a Checklist	117
3.5.2	Reporting Against a Checklist	120
3.5.3	Checklist Satisfaction	123
3.6	Implementation	125
3.6.1	Checklist Satisfaction using SPIN	127
3.7	Evaluating chembox Data with Chemmim	129
3.7.1	Iterative Assessment - The IQ Life-cycle	133
3.8	Comparison with other approaches	134

3.8.1	minim	134
3.8.2	OWL and OWL Integrity Constraints	136
3.9	Discussion	138
3.10	Future Work for MIM	139
3.11	Chapter Summary	141
4	Quality Fragments: a Probabilistic Approach	142
4.1	Chapter Introduction	142
4.2	Modelling Predictive Quality Knowledge	143
4.3	Bayesian Networks	146
4.3.1	Modelling the Variables of a Metric: Continuous vs. Discrete	150
4.3.2	Modelling the GAQ Metric	151
4.3.3	Modular Metrics	154
4.4	Multi-Entity Bayesian Networks	156
4.5	Implementation	163
4.5.1	Quality Knowledge Encoding with PR-OWL and the Evi- dent Vocabulary.	166
4.6	Assessing Bio2RDF data with Quality Fragments	169
4.6.1	Data Preparation	170
4.6.2	Replicating the GAQ Metric.	171
4.6.3	GAQ Assessment with Missing Metadata	173
4.7	Discussion	179
4.7.1	Some Notes on Implementation	181
4.8	Comparison with Related Work	182
4.9	Future Work for Quality Fragments	183
4.10	Chapter Summary	184
5	Procedurally Building Quality Fragments using Provenance	186
5.1	Chapter Introduction	186
5.2	Provenance-based Quality Knowledge	187
5.2.1	Our Intuition	190
5.3	PROV	192
5.4	Influence Factor	199
5.4.1	Modelling evident:influenceFactor	200
5.5	Quality Fragment Generation Procedure	202
5.5.1	Assumptions and Restrictions	203

5.5.2	Structure Generation	204
5.5.3	Combination Strategy	206
5.5.4	Influence Factor Normalization	208
5.6	Evaluation	212
5.6.1	The Zeng Network	213
5.6.2	Data Preparation	217
5.6.3	The Evident Generator	220
5.6.4	The Evident Vocabulary for Provenance-aware Quality Fragments.	221
5.6.5	The Generated Quality Fragment	222
5.6.6	Comparison of the Generated Network With the Zeng Network	225
5.7	Comparison with Related Work	232
5.8	Conclusions and Future Work for Quality Fragment Generation .	233
5.9	Chapter Summary	235
6	Conclusions	236
6.1	Summary of Research Contributions	236
6.2	Future Work	241
	Bibliography	245
A	Minimum Information Checklist Analysis	275
B	The MIM Vocabulary	279
C	The chemmim Checklist	285
D	The mimspin MIM Validation Rules	288
E	The Evident Vocabulary	305
F	GAQ Multi-Entity Bayesian Network	308
G	Wikiprov RDF Serialization Example	324

Word Count: 65331

List of Tables

3	Table of Artefacts that Support this Thesis	21
2.1	Information Quality Dimensions Classified using OPS IQ	74
2.2	Provenance Vocabularies	79
2.3	Provenance Vocabulary Usage in the Web of Data	81
2.4	IQ Assessment Frameworks	83
3.1	Summary of Minimum Information Checklist Analysis	112
4.1	Summary of Evaluation Datasets	173
4.2	Results for GAQ Mfrags compared with GA2GAQ.pl	174
5.1	The Core Elements of the PROV Data Model	196
5.2	Summary of PROV Influence Types in ProvBench	198
5.3	t_{A_i} CPT	215
5.4	AuthorTrust CPT	215
5.5	Results from the Zeng Network	216
5.6	Statistics of the Training Dataset	218
5.7	Statistics of the Test Dataset	218
5.8	Results from the <i>article_Q</i> Quality Fragment compared with the Zeng Network	226
5.9	<i>article_Q</i> Results for Featured and Cleanup Articles in Training Set	230
5.10	<i>article_Q</i> Results for Featured and Cleanup Articles in Test Set .	231

List of Figures

1.1	The Minimum Information about and RNAi Experiment (MIARE) Checklist (extract).	33
1.2	Annotation of UniprotKB P06727 with the term “lipoprotein metabolic process” (GO:0042157) in the Gene Ontology Annotations Database.	34
1.3	Position of the term “lipoprotein metabolic process” (GO:0042157) in the Gene Ontology.	35
1.4	The Three Aspects of Quality Knowledge.	37
1.5	A Readers Guide to the Structure and Dependencies in this Thesis	41
2.1	Two-dimensional matrix illustrating a spectrum of applicability and effectiveness of IQ metrics.	51
2.2	Map of Concepts used in this Thesis	54
2.3	The Information Quality Life-cycle [Mis08]	54
2.4	Abstract Representation of the GAQ score Assessment processes using a Reusable Quality Component for the GAQ Score	55
2.5	Abstract Representation of a Quality View reusing the GAQ Score Assessment	57
2.6	The Linking Open Data cloud diagram from [CJ11]	59
2.7	Abstract Representation of a Quality Standard based Assessment	65
2.8	Evidence Code Ranks (ECRs) as Defined by Buza et al. as an Example of Prior Knowledge	67
2.9	Abstract Representation of a Quality Fragment based Assessment	67
2.10	Abstract Representation of a Quality View based Assessment . . .	69
2.11	Using the OPS IQ classification.	72
2.12	Core Elements of the PROV Provenance Model [GM13]	77
2.13	Example PROV Provenance Graph for a GO Annotation.	78
2.14	The WIQA Filtering Processes.	84
2.15	Intrinsic vs. Provenance-based Quality Knowledge in the GAQ score.	89

2.16	Example Bayesian Network Metric from Wang et al. [WV05]	92
3.1	The Minimum Information about and RNAi Experiment (MIARE) Checklist (Assay Requirement Set).	99
3.2	A Minimum Information Checklist Solution in the Web of Data.	101
3.3	The Wikipedia Article and Chembox for Bicarbonate	103
3.4	Interaction Between the IQ Life-cycle and Linked Data Extraction processes.	108
3.5	The 11 Requirements of the Chemmim Checklist.	109
3.6	Comparison of Wikipedia Article for Aluminium Hydroxide Oxide (Stub) and Acetic Acid (A).	111
3.7	The Anatomy of a Minimum Information Checklist.	113
3.8	The Minimum Information Model Vocabulary.	116
3.9	Creating an ObjectReport and Aligning it with a Checklist Requirement.	121
3.10	Creating a DataReport using a blank node and Aligning it with a Checklist Requirement.	121
3.11	Creating a ReportSet and Aligning it with a Checklist Requirement Set.	122
3.12	Associating the InChI Requirement with a SPARQL Query to Automatically Generate Reports.	123
3.13	Requirement Satisfaction.	124
3.14	Requirement Satisfaction.	125
3.15	Implementation of the MIM Web Service	126
3.16	SPIN Rules in the mimspin Ontology for Constructing Triples.	128
3.17	SPIN Rules in the mimspin Ontology for Constraint Checking.	128
3.18	Individual Requirement Satisfaction Across All Chembox Instances.	132
3.19	Requirement Set Satisfaction Across All Chembox Instances.	132
4.1	Bayesian Network for chembox Quality.	147
4.2	Bayesian Network for <i>GAQ</i> score	152
4.3	Normal and Cumulative Distribution of <i>GAQ</i> scores.	154
4.4	Bayesian Network for <i>groupGAQ</i> with two annotations.	156
4.5	The <i>GAQScore</i> , <i>productGAQ</i> , <i>groupGAQ</i> , and <i>meanGAQ</i> MFragments.	159

4.6	Example SSBN generated using the productGAQ MFrag for a gene product <i>bio2rdf_uniprot : Q8IZF5</i> with two annotations <i>goa_resource : GDB_74</i> and <i>goa_resource : GDB_75</i>	161
4.7	The myGAQ MFrag.	163
4.8	The Evident Orchestrator Framework	163
4.9	The UnbBayes Framework Workbench	169
4.10	Part of the productGAQ SSBN for Apolipoprotein A-IV (uniprot:P06272) visualized in UnbBayes	173
4.11	GAQ Score Assessment for annotations with Missing Depth Metadata.	175
4.12	GAQ Score Assessment for annotations with Missing Depth Metadata.	176
4.13	Comparison of each Gene Product productGAQ Scores with Missing Depth Metadata.	177
4.14	GAQ Score Assessment for P06727 with Missing evidence code Metadata.	178
4.15	Comparison of each Gene Product productGAQ Scores with Missing evidence code Metadata.	179
5.1	Quality Fragment Generation for Chembox based on WikiArticle Quality Fragment and Provenance Data.	189
5.2	The Extended PROV Model [LSM ⁺ 13]	193
5.3	Example of a Provenance Graph from ProvBench Wikipedia using wasInfluencedBy	194
5.4	Example of a Provenance Graph from ProvBench Wikipedia using sub-properties of wasInfluencedBy	194
5.5	Example of a Provenance Graph from ProvBench Wikipedia using qualifiedGeneration	195
5.6	Using evident:influenceFactor in Wikipedia Provenance	201
5.7	Overstating evident:normalInfluenceFactor in Wikipedia Provenance	202
5.8	Quality Fragment Generation for Chembox using Quality Fragment Generation Procedure.	206
5.9	Quality Fragment Generation for Chembox after Initialization. . .	207
5.10	Quality Fragment Generation for Chembox after First Pass of Generation.	207

5.11	Provenance graph for Chembox Derived From two Wikipedia Ar- ticles.	209
5.12	SSBN for Chembox derived from two Wikipedia Articles.	209
5.13	Dynamic Bayesian Network from Zeng et al. to Estimate Quality of Wikipedia Revisions	214
5.14	Author Trust Quality Fragment	216
5.15	Wikipedia Provenance Example with Additional Metadata	219
5.16	The Evident Generator Framework	221
5.17	AuthorTrust Quality Fragment Described using evident Vocabulary.	222
5.18	The Automatically Generated Quality Fragment for Wikipedia Provenance.	224
5.19	Assessment Values for Revisions of the United States National Forrest Wikipedia Article using Automatically generated Quality Fragments	226
5.20	Assessment Values for Revisions of the United States National For- rest Wikipedia Article using Bayesian Network from Zeng et al. .	227
5.21	Comparison of Assessment Values for cleanup and featured in Train- ing Set.	228
5.22	Confusion Matrix for Test Set Classification	228
6.1	Structure of the Thesis Presented	237

Listings

2.1	Example WIQA Policy from Bizer et al.	85
3.1	The Simple chembox MediaWiki Template	104
3.2	Extract from DBpedia Resource for the Bicarbonate Article . . .	106
3.3	The chemmim Checklist Encoded Using the MIM Vocabulary (ex- tract).	117
3.4	Declaring a <code>mim:DataRequirement</code>	118
3.5	Declaring a <code>mim:ObjectRequirement</code>	118
3.6	Declaring a Type Restriction on the <code>InChI</code> Requirement.	118
3.7	Declaring a Type Restriction on the <code>Image</code> Requirement.	119
3.8	Declaring the <code>Identifiers</code> Requirement Set.	119
3.9	Declaring the Requirements that are Members of the <code>Identifiers</code> Requirement Set.	119
3.10	Declaring a Cardinality Constraint on the <code>InChI</code> and <code>SMILES</code> Re- quirements in the <code>Identifiers</code> Requirement Set	119
3.11	<code>mimspin:DataRequirementSatisfaction</code>	128
3.12	<code>mimspin:violates DatatypeRestriction</code>	129
3.13	chembox Linked Data Extraction for the Article Ethane (extract)	131
3.14	<code>InChI</code> Requirement Described using OWL	136
3.15	<code>InChI</code> Requirement with Cardinality Constraint Described using OWL	137
3.16	<code>InChI</code> Requirement Translated to SPARQL-based Integrity Con- straint	137
4.1	Bio2RDF Gene Ontology Annotations Data (Extract).	144
4.2	Bio2RDF Gene Ontology Annotations Data with <code>sn_goa:GAQscore</code> property.	158
4.3	Example of a Context Constraint interpreted as an OWL Class Description	166

4.4	The GAQscore MFrag in PR-OWL2	167
4.5	Describing a Continuous Resident Node using <code>pr-owl2:ContinuousResidentNode</code>	167
4.6	The Continuous Function for the meanGAQ Resident Node	167
4.7	Aligning the depth Variable with the RDF property <code>go-depth</code> . .	168
4.8	Aligning the productGAQ variable with the type <code>uniprot_core:Protein</code> using <code>evident:definesMetricFor</code>	168
4.9	SPARQL Query to integrate Bio2RDF data for evaluation	172
5.1	Provenance Describing that the chembox <i>wasDerivedFrom</i> a Wikipedia article.	189
5.2	RDF Representation of WikiProject Chemicals Assessment Data.	190
5.3	The CPD Script Generated for <i>article_Q(article_1)</i>	225

Abstract

The Web is being transformed into an open data commons, and is now the dominant point of access for information seeking scientists. In parallel the scientific community has been required to manage the challenges of “Big Data” - characterized by its large-scale, distributed, and diverse nature. The *Web of Linked Data* has emerged as a platform through which the sciences can meet this challenge, allowing them to publish and reuse data in a machine readable manner. The openness of the Web of Data is however a double-edged sword. On one hand it drives a rapid growth of adoption, but on the other a lack of governance and quality control has led to data of varied quality and trustworthiness. The challenge scientists face then is not that data on the Web is universally poor, but that the quality is *unknown*.

Previous research has established the notion of *Quality Knowledge*, latent domain knowledge possessed by expert scientists to make quality based decisions. The main idea pursued in this thesis is that we can address Information Quality (IQ) issues in the Web of Data by repurposing these existing mechanisms scientists use to evaluate data. We argue that there are three distinct aspects of Quality Knowledge, *objective*, *predictive*, and *subjective*, defined by information required for their assessment, and present two studies focused on the modelling and exploitation of the objective and predictive aspects. We address the objective aspect by developing the Minimum Information Model as a repurposing of Minimum Information Checklists, an increasingly prevalent type of quality knowledge employed in the Life Sciences. A more general approach to modelling the predictive aspect explores the use of Multi-Entity Bayesian Networks to tackle the characteristic uncertainty in predictive quality knowledge, and the inconsistent availability of metadata in the Web of Data.

We show that by following our classification we can develop techniques and infrastructure to successfully evaluate IQ that are tailored to the challenges of the Web of Data, and informed by the needs of the scientific community.

Declaration

No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Copyright

- i. The author of this thesis (including any appendices and/or schedules to this thesis) owns certain copyright or related rights in it (the “Copyright”) and s/he has given The University of Manchester certain rights to use such Copyright, including for administrative purposes.
- ii. Copies of this thesis, either in full or in extracts and whether in hard or electronic copy, may be made **only** in accordance with the Copyright, Designs and Patents Act 1988 (as amended) and regulations issued under it or, where appropriate, in accordance with licensing agreements which the University has from time to time. This page must form part of any such copies made.
- iii. The ownership of certain Copyright, patents, designs, trade marks and other intellectual property (the “Intellectual Property”) and any reproductions of copyright works in the thesis, for example graphs and tables (“Reproductions”), which may be described in this thesis, may not be owned by the author and may be owned by third parties. Such Intellectual Property and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property and/or Reproductions.
- iv. Further information on the conditions under which disclosure, publication and commercialisation of this thesis, the Copyright and any Intellectual Property and/or Reproductions described in it may take place is available in the University IP Policy (see <http://documents.manchester.ac.uk/DocuInfo.aspx?DocID=487>), in any relevant Thesis restriction declarations deposited in the University Library, The University Library’s regulations (see <http://www.manchester.ac.uk/library/aboutus/regulations>) and in The University’s policy on presentation of Theses

Abbreviations

CPD	Conditional Probability Distribution.
CPT	Conditional Probability Table.
CSVF	Chemical Structure Validation Filter.
ECR	Evidence Code Rank.
GAQ	Gene Ontology Annotation Quality Score.
GO	The Gene Ontology Annotations Database.
GO	The Gene Ontology.
IC	Integrity Constraint.
InChI	IUPAC International Chemical Identifier.
IQ	Information Quality.
ISA	Investigation Study Assay.
IUPAC	International Union of Pure and Applied Chemistry.
LOD	Linked Open Data.
MEBN	Multi-Entity Bayesian Network.
MFrag	MEBN Fragment.
MIABE	Minimum Information about a Bioactive Entity.
MIARE	Minimum Information about an RNAi Experiment.
MIBBI	Minimum Information for Biological and Biomedical Investigations.
MIC	Minimum Information Checklist.
MIM	Minimum Information Model.
MIMV	Minimum Information Model Vocabulary.
minim	The minim minimum information checklist vocabulary.

NIH	National Institute of Health.
NPC	NIH Chemical Genomics Center Pharmaceutical Collection.
OPS IQ	Objective, Predictive and Subjective IQ classification.
OWA	Open World Assumption.
OWL	The Web Ontology Language.
PGM	Probabilistic Graphical Model.
PR-OWL2	The Probabilistic Ontology Language.
PROV	The PROV specification for provenance on the Web.
PROV-DM	The PROV Data Model.
PROV-O	The PROV Ontology.
RDF	Resource Description Framework.
RFC 2119	Key words for use in RFCs to Indicate Requirement Levels.
SMILES	Simplified molecular-input line-entry system.
SPARQL	SPARQL Protocol and RDF Query Language.
SPIN	SPARQL Inferencing Notation.
SSBN	Situation Specific Bayesian Network.
UNA	Unique Names Assumption.
URI	Unique Resource Locator.
W3C	World Wide Web Consortium.

Glossary

Quality Knowledge	The latent knowledge scientists use to make quality-based decisions.
Objective Quality Knowledge	Quality Knowledge informed by an objectively defined standard.
Predictive Quality Knowledge	Quality Knowledge informed by prior knowledge that can relate features of data to its likely quality.
Subjective Quality Knowledge	Quality Knowledge informed by the user's subjective needs.
Quality Knowledge Encoding	The mechanism used to encode Quality Knowledge as a reusable quality component e.g. a perl script.
Quality Evidence	The metadata required to make a quality assessment.
Quality Evidence Alignment	A process of identifying and annotating Quality Evidence in some data.
Quality Evidence Encoding	The mechanism used to describe Quality Evidence e.g. RDFS Vocabularies.
Quality Fragment	A reusable quality component that makes use of predictive Quality Knowledge.
Quality Standard	A reusable quality component that makes use of objective Quality Knowledge.
Quality View	A reusable quality component that makes use of subjective Quality Knowledge.
Reusable Quality Component	Quality Knowledge realised as a software component.

The Information Quality Life-Cycle The process of designing and creating reusable quality components.

Table of Artefacts

Name	Location
Minimum Information Model Vocabulary	http://purl.org/net/mim/ns
Minimum Information Model SPIN Semantics	http://purl.org/net/mim/mimspin
Chemmmim Checklist	http://purl.org/net/chembox/chemmmim
Chemmmim Chembox SPIN alignment rules	http://purl.org/net/chembox/chemboxspin
Chembox RDF Extraction Data	http://purl.org/net/chembox/
Mim Validation Web Service Implementation	http://github.com/matthewgamble/mim-ws/
Gene Ontology Annotations RDF Data	http://purl.org/net/goa/
Wikipedia-provenance RDF extraction tool	http://github.com/matthewgamble/wikipedia-provenance
Wikipedia-provenance RDF Data	http://purl.org/net/wikipedia-provenance/
The Evident Vocabulary	http://purl.org/net/evident

Table 3: Table of Artefacts that Support this Thesis

“*The reasonable man adapts himself to the world;
the unreasonable one persists in trying to adapt the world to
himself.
Therefore all progress depends on the unreasonable man.*”

- *George Bernard Shaw*

“ *In music, one doesn't make the end of the composition the point of the composition.
If that were so, the best conductors would be those who played fastest, and there would be composers who wrote only finales.
People would go to concerts just to hear one crashing chord – because thats the end!*

We thought of life by analogy with a journey, with a pilgrimage, which had a serious purpose at the end, and the thing was to get to that end: success, or whatever it is, or maybe heaven after youre dead.

*But we missed the point the whole way along.
It was a musical thing – and you were supposed to sing, or dance, while the music was being played. ”*

- Alan Watts

Acknowledgements

First and foremost, I would like to thank my wife Jennifer; whose unwavering support is the reason this thesis has been written; and who was not my wife before this process began, yet still agreed to be in spite of it. To my parents Alison and George, thank you for your support, and for putting me in a position to even be able to consider taking on such a task. And to Dr. Peter McNerney for going through the journey with me, and not rubbing it in my face when he beat me to the post.

I would especially like to thank my supervisor Prof. Carole Goble CBE for her patience and guidance, and providing every opportunity that a PhD student could wish for. To Prof. David Karger thank you for teaching me to keep my eye on the nail, and not the hammer. And to Lorraine Wood, who was perhaps the first to take seriously my academic endeavours.

I would also like to thank Mr Ian Cottam for his support in using the EPS condor pool for data analysis and preparation, which quite literally took months (if not years) off the time required. Finally thanks go to Antony Williams for his insight and guidance through the jungle that is chemistry data online.

This research has been supported by an *Engineering and Physical Sciences Research Council Doctoral Training Award*, a *University of Manchester School of Computer Science Annual Enhanced ICT Stipend*, and a *University of Manchester School of Computer Science Enhanced ICT Stipend Award for Excellent Academic Achievements*.

This research is also partially supported by: the Wf4Ever project Project 270129 funded under EU FP7 Digital Libraries and Digital Preservation (ICT-2009.4.1), and the Innovative Medicines Initiative Joint Undertaking under grant agreement number 115191, resources of which are composed of financial contribution from the European Unions Seventh Framework Programme (FP7/2007-2013) and EFPIA companies in kind contribution.

Chapter 1

Introduction

“*The Feynman Problem-Solving Algorithm:*

1) write down the problem;

2) think very hard;

3) write down the answer. ”

- proposed by Murray Gell-Mann,

Nobel Laureate and colleague of Richard Feynman at CalTech

1.1 Problem

The way in which scientists interact with data has been significantly impacted in recent years with the proliferation of free- and open-access databases and data resources [HLOD10] [Wil08] [HLVA07]. The Web is facilitating a transformation in scientific practice, impacting dissemination of scholarly literature [HPK08] and increasingly the way we share and consume scientific data [Sci11][Nat08a]. Popular Web-based resources such as GeneWiki, WikiGenes, ChemSpider, Uniprot, the Gene Ontology Annotations Database, DrugBank, PubChem, PDB, ChEMBL, and DailyMed, all provide access to scientific data and are accessed by thousands of research scientists everyday. The Web is being transformed into an open data commons by technology, policy, and community activity [Bou12], and is now the dominant point of access for information seeking scientists [NHL⁺10]. In parallel as a result of advances in computational processing, and increasingly computationally aware scientific practitioners, the volumes of data that scientists are

producing and consuming has dramatically increased [HT03]. The scientific community has been perhaps the first community forced to manage the challenges of “Big Data” [Nat08b] - characterized by the large-scale, distributed, and diverse nature of the data. This “data deluge” has resulted in a new wave of scientific disciplines focused on the re-use and repurposing of experimental data [FSC⁺09] in order to generate new results, and transformed existing disciplines such as Ecology [RJS11], that can benefit from the re-use of data.

More recently the Web of Linked Data [BHBL09] has emerged as a platform through which the sciences can meet this Big Data challenge, allowing them to publish, share, discover and ultimately reuse data in a machine readable manner. The rapid growth and wide adoption of the Linked Data approach is underpinned by the openness of the platform. Using established Web standards such as Uniform Resource Identifiers (URIs) and the Resource Description Framework (RDF), Linked Data provides a Web-scale and open approach to data integration. Everybody can publish their data on this open Web, make replicates, and host them at distributed locations. The goal of the Web of Data is to create a global data-space by lowering the barriers to joining together previously separate data. Since 2007 the Linking Open Data project [LOD] has been tracking the progress of the Web of Data, and by 2012 there was an estimated 52 billion RDF triples.

The open nature of the Web of Data has particularly attracted the attention of the Life Sciences community, because rarely does a single research group have sufficient resources to manage data across the whole spectrum of varied complexities within their domain [FSC⁺09]. Consequently, an increasing volume of biological data has been made available in a Linked Data format. Each of the Web-based data resources mentioned above for example are available on the Web of Data. To enable ‘big’ science these distributed datasets must be gathered and integrated in order to create a big picture about what is known or what can be done. Therefore along side these standalone datasets are datasets targeted at integration. The majority of these efforts have been created by third parties attempting to integrate existing resources, these include Bio2RDF [BNt⁺08], Chem2Bio2RDF [CDJ⁺10], LinkedLifeData [MPPG09] and the Open PHACTS [WHG⁺12] project. More recently the European Bioinformatics Institute (EBI) has embraced the Linked Data approach by providing an RDF-based platform for six of its databases. As one of the Life Sciences’ major service providers, this is a clear signal of the community’s move towards Web-scale open integration.

However, as with the traditional document Web, this openness is a double-edged sword. On one hand it drives a rapid growth of adoption and has made a large volume of data accessible on the Web in a structured format. On the other hand, lack of governance and quality control has led to a Web of Data of varied quality and trustworthiness [SK12] [ZRM⁺12] [HUH⁺12]. Bio2RDF data for example is only published periodically, and quickly becomes out of date with respect to its source data. The fact that it is out of date, and exactly which version of the source data the dataset is based on, is not always obvious.

A challenge then, common to each of these data intensive fields of research, is that their results are almost wholly reliant upon the quality of data that they are gathering from the Web. Errors can and do appear in online datasets [WE11] [WMSE10]. These errors then propagate across other resources on the Web, and impact results [OOO⁺02]. Take for example the recently published National Institute of Health’s (NIH) Chemical Genomics Center Pharmaceutical Collection (NPC) [HSW⁺11]. The collection was published on the Web by the NIH as a “definitive, complete, and non redundant” set of all approved molecular entities. Issues relating to the quality of the data available in the collection were raised by members of the online chemistry community soon after its release [Eki11], with estimates of up to 10% of the entries having some form of error [WE11]. These errors, from a well trusted organization, have the potential for wide impact. The creators of the NPC browser have subsequently included a statement on their Web page addressing the issues, explaining:

“ Despite our best efforts on curation, every structure is suspect until proven otherwise. This sentiment certainly applies equally to any chemical database. ”

The challenge scientists face is not that the quality of data on the Web is universally poor, instead the challenge is that the quality is *unknown*. Furthermore it is often difficult and time consuming to assess the quality of the data. In direct response to the issues discovered in the NIH’s NPC data, a team of cheminformaticians conducted a study to verify the correctness of available information about the 200 best-selling drugs [FMF⁺12]. Four independent groups from leading institutions; The Royal Society of Chemistry, University of North Carolina, AstraZeneca, and the Institut Hospital del Mar, were asked to discover the correct structures of the drugs using Web-based resources, and then compare

results. The study took the teams approximately one week to conduct, and no group achieved 100% accuracy.

As a contrasting example, consider Wikipedia¹. Wikipedia is in many respects the Web in microcosm - anyone can publish and edit content, even anonymously. Data from Wikipedia is also available on the Web of Data in the form of the popular DBpedia [ABK⁺07]. Projects such as GeneWiki [HIOG⁺08] and WikiProjectChemicals [WPC12] provide large volumes of scientific data using Wikipedia as a platform. WikiGenes for example contains over 10,000 pages on gene and protein function that are regularly accessed over 50 million times a year, and edited over 15,000 times a year [Su09].

The wide held perception of Wikipedia is that it is a low quality, and therefore untrustworthy resource [PWS09]. It is similarly distrusted by the scientific community [WE11]. There have however been a number of studies that highlight a disagreement between this perceived low trustworthiness of information on Wikipedia by users, and the empirically validated correctness [Gil05][Che06][WE11].

These studies of the NPC browser and Wikipedia demonstrate that the quality of information held on Web based data resources may not be immediately obvious to the consumers, and can even contradict expectation. The NPC browser is provided by the NIH, a highly trusted source, yet has been shown to contain a significant number of errors. Wikipedia is generally perceived to be an untrustworthy resource, yet has been shown to be of relatively high quality.

It is clear then that there is a need to establish approaches that aid the scientific community in dealing with this data quality issue, making the information about the quality of resources more explicit, and supporting them in making decisions about which resources to trust.

Broadly there are two strategies to dealing with the issue of “messy data” [SK12]:

1. Develop best practices to encourage better quality data at the point of publishing.
2. Develop techniques and infrastructure that help users manage and overcome the problems posed by messy data.

The first of these strategies is a challenging prospect. The diversity of publication pipelines and open nature of the Web mean that high levels of support

¹<http://www.wikipedia.org>

and community engagement are required to gain adoption, which goes beyond the scope of this thesis. Our focus is on the second of these strategies, dealing with data “post publication”. This does not preclude the fact however that techniques proposed in this thesis also have the potential to address the first strategy.

1.1.1 A Motivating Question

The goal of this thesis is therefore to develop techniques to address questions of the type:

- What is the quality of the DBpedia entry for the chemical Ethane?

Information Quality (IQ) is an ambiguous and overloaded term [Cha05], and has long been the subject of systematic study in a diverse set of fields and situations. In order to support the scientific user in discovering useful data our first challenge is to understand the concept of quality from the scientific user’s perspective.

In order to answer this question it is therefore necessary to ask a number of follow up questions, including:

- What do we mean by *quality*?
- Who is using the data and what do *they* mean by quality?
- What *information* do we have to hand to assess the entry?

Practically our problem is impacted further by our data platform - the Web of Linked Data. The data is highly heterogeneous, and even data of the same type can be represented in a multitude of different ways, with varying levels of supporting data, termed “metadata”. As such any solution attempting to answer questions like the one above need to be robust in the face of uneven representation and metadata.

The Web of Linked Data has been characterised as the “pay-as-you-go Web” [BS10], with much of its success coming from incremental and distributed approaches to publishing, integration and improvement [Biz13]. It follows that approaches to IQ assessment on the Web of Data must be equally as agile in order to be successful.

1.1.2 Quality Knowledge and the Information Quality Life-cycle

Previous work investigating IQ in e-Science by Missier [Mis08] has established the notion of *Quality Knowledge* in the sciences. Quality Knowledge is the latent domain knowledge possessed by expert scientists that they apply to make quality based decisions. This knowledge must be elicited from the scientists in order to be encoded for re-use. Take for example the process of automated DNA sequencing which possesses inherent uncertainty. The Phred score [EG98] is an attribute of computed DNA sequences that can be used to characterize their quality, where a higher Phred score indicates a higher likelihood of a correct sequence. The knowledge related to how to calculate the Phred score, and how to apply it to filter low quality sequences are both examples of Quality Knowledge. By developing mechanisms to encode and share this knowledge it is possible then to develop *reusable quality components*; software components that support its automatic application. For the Phred score for example, the command line tool **Phred** [GE02] is a reusable quality component that can be used to calculate the Phred quality scores for a given sequence.

Missier describes the process of creating and exploiting these reusable quality components in the *Information Quality Life-cycle*. This life-cycle is a continuous process in order to develop, exploit, and improve reusable quality components.

The main idea pursued in this thesis is that we can address IQ issues in the Web of Data by grounding our work in this Life-cycle and Quality Knowledge, and that existing mechanisms scientists use to evaluate data can be repurposed and applied to the Web of Data. Furthermore we believe that there are common features that define distinct aspects of Quality Knowledge, that mean we can propose general techniques and infrastructure to build reusable quality components. In the next section, we provide some detailed examples of existing Quality Knowledge used to evaluate scientific data on the Web.

1.2 Examples of Quality Knowledge for Scientific Data on the Web

We have found that Quality Knowledge is apparent in the mechanisms already employed by scientists [GG11a], and readily observable in the literature. To illustrate this knowledge in the context of the scientific data on the Web we introduce three motivating examples below. We continue to refer to these examples through the thesis to ground our work in real challenges faced by the scientific user.

1.2.1 Minimum Information Checklists

In the Life Sciences, the Biosharing initiative [SRSF⁺12] is driving efforts to control the quality of the data published by a diverse range of research groups by gathering together and coordinating reporting standards to which data submitters must comply. Several classes of interoperable reporting standard are combined: reporting frameworks such as the ISA (Investigation, Study, Assay) framework [SRSB⁺08], data formats, controlled vocabularies, and Minimum Information Checklists (MICs)[TFS⁺08].

In order to fully understand an experimental report, scientists require a certain degree of metadata pertaining to the experiment’s context, instruments used, methodology etc. MICs such as the Minimum Information About an RNAi Experiment (MIARE) checklist shown in Figure 1.1, detail the information that should be reported to ensure that an experiment description is of sufficient quality. Each requirement in the checklist describes a piece of information that should be provided when publishing an experimental report. Requirements in checklists such as MIARE vary in granularity from specific information such as MIARE B.2.6.1 requiring the name of an instrument used or MIARE B.1.2 requiring a description of the biological question being addressed. The Quality Knowledge encoded in these checklists is that by providing the information specified, the resulting experiment report will be of higher quality, and therefore more useful to the community.

These checklists cover a wide range of types of biological investigation with some 60+ MICs currently listed for structuring and curating data on the Biosharing resource. The Minimum Information for Biological and Biomedical Investigations (MIBBI) project [TFS⁺08] in particular has highlighted the important role of MICs in the reporting of biological investigations. MIBBI is primarily focused

MIARE – Summary of Required Information

Minimum Information About an RNAi Experiment (MIARE)
(www.miare.org)

Checklist of Required Information*

The purpose of this check-list is to guide and help experimentalists to ensure that the data supporting their results based on RNA interference experiments can be made publicly available, in a format that enables unambiguous interpretation of the data and potential verification of the conclusions.

The following check-list only contains mandatory information, describing the information that SHALL¹ be reported for an RNAi experiment. OPTIONAL information has been omitted and can be found in the full MIARE Reporting Guideline document at www.miare.org

Checklist

A. Assay description:

- A.1. Assay ID
- A.2. Assay name
- A.3. Assay type (primary/confirmatory/other)
- A.4. Target organism (Taxonomy ID)
- A.5. Number of distinct genes targeted for knock-down
- A.6. Experiment publication (PubMed ID)
- A.7. Primary contact information

B. Protocol:

B.1. Experimental description

- B.1.1. Experiment title
- B.1.2. Biological question description - (*including sample description and keywords*)

B.2. Assay

- B.2.1. Assay protocol and design -(*including number and description of replicates (biological/technical)*)
- B.2.2. Pre- and post-treatment (protocol/type/compound)
- B.2.3. Bio-material manipulations (*including growth conditions/cell culture conditions and if applicable cell separation technique*)
- B.2.4. Number of cells per well
- B.2.5. Compound(s) name (if applicable)
 - B.2.5.1. Assay reagent name
 - B.2.5.2. Assay reagent manufacturer
- B.2.6. Instrument (repeat this section for each instrument used)
 - B.2.6.1. Instrument name
 - B.2.6.2. Instrument manufacturer
 - B.2.6.3. Type of readout
 - B.2.6.4. Instrument settings

v0.8.0 / May 2011

Page 2 of 6

Figure 1.1: The Minimum Information about and RNAi Experiment (MIARE) Checklist (extract).

on the ‘Omics, where experiments are characterized by high volumes of output data with a significant potential for reuse. The integration of quality control in the process of making data accessible has led to the creation of a number of respected ‘Omics databases, such as the ArrayExpress database that are regulated by the Minimum Information about a Micro Array Experiment (MIAME) and

Database	Gene Product ID	Symbol	Qualifier	GO Identifier	GO Term Name	Aspect	Evidence	Reference	Taxon	Date	Assigned By
UniProtKB	P06727	APOA4		GO:0042157	lipoprotein metabolic process	P	IEA	InterPro2GO	9606	20131116	InterPro

↑
Gene Product
↑
GO Term
↑
Evidence Code

Figure 1.2: Annotation of UniprotKB P06727 with the term “lipoprotein metabolic process” (GO:0042157) in the Gene Ontology Annotations Database.

Minimum Information about a Sequence (MINSEQE) MICs. In principle, an entry can not be published in the ArrayExpress database if it is not compliant with the relevant MIC.

1.2.2 Gene Ontology Annotations Quality Score

The EBI’s Gene Ontology Annotation (GOA) project is a resource for functional genomics [CMB⁺04], currently providing gene annotations data for well over 100,000 species. The Gene Ontology is a controlled vocabulary that is used to describe the attributes of gene products. Gene annotation is then the process of associating terms in the Gene Ontology with identifiers for gene products from popular resources, such as the Uniprot Knowledge Base (UniprotKB). Figure 1.2 shows an example gene annotation from the GOA database, annotating the gene product P06727 from UniprotKB, with the term “lipoprotein metabolic process” (GO:0042157) in the Gene Ontology. Each annotation is enriched with information about how that annotation was produced using one of a series of *evidence codes*. The annotation shown for example has the evidence code IEA. This means that the annotation was “Inferred from Electronic Annotation”, this is used to indicate when the annotation has been created computationally, and no curator has checked the annotation to verify its accuracy.

With such high volumes of data from varying sources it can be difficult for users to understand the quality of the data in the GOA database [SAD12]. The Gene Ontology Annotation Quality Score (GAQ) [BMW⁺08] is an example Quality Knowledge used to address this problem and assess the quality of gene annotations. The GAQ score defines a numeric measure of quality against which an annotation a in the GOA database can be measured as:

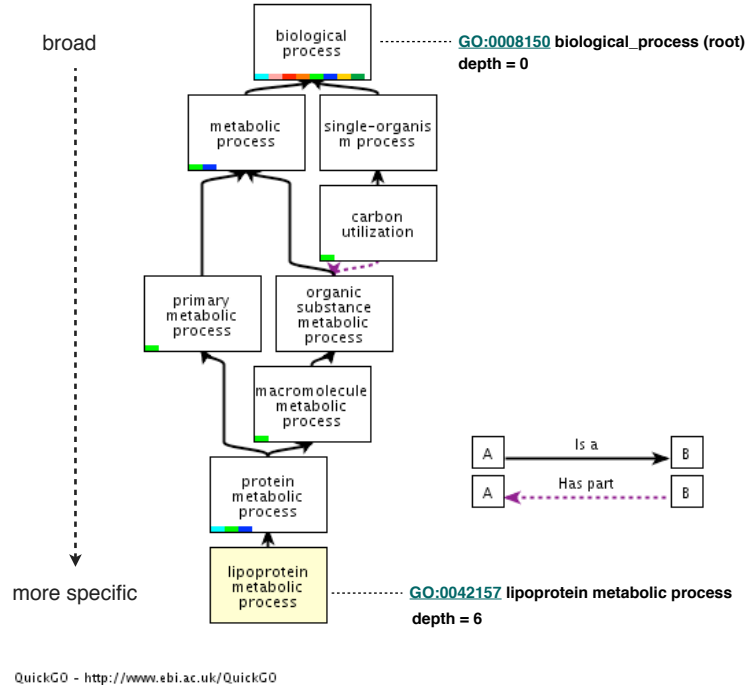


Figure 1.3: Position of the term “lipoprotein metabolic process” (GO:0042157) in the Gene Ontology.

$$GAQ(a) = ECR_a \times depth_a$$

The GAQ score for an individual annotation is defined as the product of its term’s depth in the ontology $depth$, and the *evidence code rank* (ECR) of the annotation. The structure of the Gene Ontology is a directed acyclic graph (DAG) with three root terms. The depth of a term from its root in the Gene Ontology is an indicator of the level of detail captured by that term. This is illustrated in Figure 1.3 showing the the term “lipoprotein metabolic process” (GO:0042157) with a depth of 6².

Evidence code rank (ECR) is a quantitative ranking of each type of evidence code according to its perceived reliability. Each evidence code has a corresponding ECR on a scale of 0 to 5. For example the evidence code Inferred from Genetic Interaction (IGI) is an annotation created by manual experimentation and is given an ECR of 5. In contrast the evidence code IEA has an ECR of 2.

²Because there is often more than one path from the root to a term, the depth d is taken as the *longest* path.

The GAQ score therefore captures two distinct elements of Quality Knowledge. Firstly the intuitive knowledge that an annotation using an ontological term from a deeper part of the ontology indicates a more specific term. Secondly, that the method by which the annotation was produced will impact its likely quality.

1.2.3 Online Chemical Structure Repositories

There has been a significant recent increase in the volume of data available in Web-based chemical structure repositories, freely available to the chemistry community [VN12]. Data from these repositories is used to build computational models and integrated into systems to support drug discovery [WET12]. As previously discussed, there are quality issues surrounding online chemical structure repositories. These issues go beyond just the NPC dataset and are a feature common across repositories. The aim of the previously discussed [FMF⁺12] study was to create a definitive list of the top 200 best-selling drugs. The list acts as a gold standard that public structure repositories can be benchmarked against. It provides chemists with some prior knowledge about how well each repository reports each of the verified structures, and with it a level of trust in the repository based on its prior performance. The result is then a quantified ranking of data accuracy in a series of well-known public data resources [WET12]. Both the gold standard and the quantified ranking are examples of Quality Knowledge used to evaluate online structure repositories.

A further strategy employed to address quality issues in online chemical databases is crowd sourced curation [WHG⁺12]. A specific issue addressed is the misalignment of chemical synonyms with chemical compounds. Chemical compounds typically have a number of possible synonyms, for example Acetic acid is also referred to as Ethanoic acid [IUP04] or Methanecarboxylic acid [EB161]. ChemSpider allows users to edit and curate the list of synonyms for each chemical to improve the discoverability of compounds. Crowd supplied synonyms can often be inaccurate and incorrectly attributed, and are accompanied by metadata about whether the curator is a registered expert or non-expert. The Quality Knowledge employed is that registered experts are more likely to supply high quality synonyms than non-experts. The metadata describing *where* the curation came from therefore plays a crucial role in deciding whether it should be trusted.

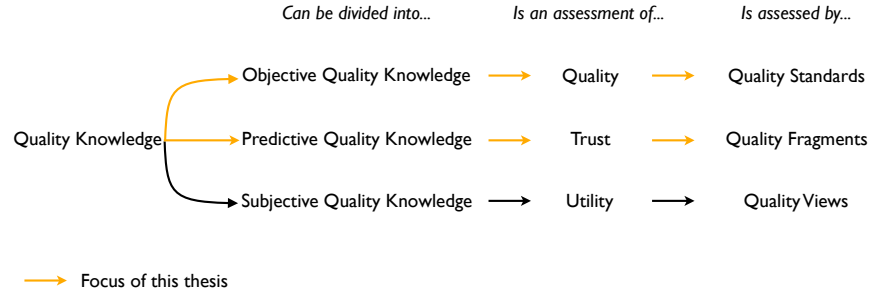


Figure 1.4: The Three Aspects of Quality Knowledge.

1.3 Research Aims and Objectives

The aim of this research is to investigate and develop approaches that successfully support the assessment of the quality of scientific data on the Web of Data. To do this we aim to allow the encoding of the practices already employed by scientists, using techniques and infrastructure that are robust to the heterogeneity of the Web of Data.

Specifically, our objectives are:

- A1** To investigate and review the scientists approach to IQ.
- A2** To review existing approaches to IQ assessment in light of the findings in A1 and establish a framework for future investigation.
- A3** To design solutions for creating reusable quality components that reflect the findings from A2, and are robust to the heterogenous nature of the Web of Data.
- A4** To realise the components in A3 in the Web of Data using Semantic Web and Linked Data technologies.
- A5** To evaluate the solutions in A4 against *real* data.
- A6** To understand how our solutions compare to existing and related work.

1.4 Objective, Predictive, and Subjective

We have surveyed the literature [GG11a], in an effort to understand the problem of IQ from the scientific user's perspective. As a first step towards our goals we

have systematically examined Quality Knowledge and divided the issue into three related aspects of *objective*, *predictive* and *subjective*.

Figure 1.4 summarizes how we have divided Quality Knowledge into these three related aspects and describes the type of reusable quality component that each aspect *is assessed by*. Our goal is to draw upon the features of these distinct aspects of Quality Knowledge and develop techniques and infrastructure for the Web of Data.

This separation of concerns is achieved by observing three, prevalent properties of scientific data: (1) that regular use of standards means that data quality is commonly defined *objectively*; (2) in the absence of a standard many of the metrics and mechanisms are *predictors* for the *likely* quality based on where the data came from (its provenance) and other metadata ; and (3) the quality and trustworthiness of the data and the entities that produced that data inform the scientific user’s *subjective* belief as to whether it is *fit-for-purpose*.

We introduce these concepts here, and expand on their discussion in Chapter 2.

Objective Quality Knowledge. Community defined norms and standards play a crucial role in the successful exchange of scientific data online [Edw04]. Standards such as MICs act as a community defined touchstone against which data can be benchmarked. MICs exemplify this objective aspect, they are often the result of experts in the field collectively defining the required set of metadata to improve the quality of reported data. We use the term *Quality Standards* to refer to reusable quality components that use this objectively defined Quality Knowledge.

Predictive Quality Knowledge. Often the complexity of data precludes the ability to establish mechanisms that directly assess the correctness of that data [Mis08]. Instead metrics such as the GAQ score are developed as a predictor of the likely quality given some evidence. This evidence is typically in the form of provenance data and metadata. In the case of the GAQ score the provenance data are the evidence codes, a description of the process that was used to create the data.

We refer to the quality components that assess this predictive aspect as *Quality Fragments*³.

³The term “Fragments” is borrowed from a type of Bayesian Network that we make use of in this thesis to model predictive Quality Knowledge.

Subjective Quality Knowledge is a fitness-for-use interpretation of data quality [Jur74], where quality assessments are combined with the scientists own requirements as the basis of a decision making process [Mis08]. For example many scientists will use a Wikipedia entry as an introduction to a new topic [MBKZ⁺11], it is likely that their requirements will be lower in this instance than if they are looking to directly use data from a GeneWiki page for subsequent experimentation. It is this subjective element of Quality Knowledge that Missier has explored in previous work. Missier refers to quality components that make use of this subjective knowledge as *Quality Views*.

1.5 Research Hypotheses

Our central hypothesis is as follows:

H1 We can use the Information Quality Life-cycle as a foundation to develop IQ solutions for scientific information in the Web of Data that can successfully support quality based decisions.

Quality decisions may rely on the result of an evaluation of a data set against one dimension of IQ, or the ranking or classification of alternative data elements.

From this central hypothesis grounding our work in the Life-cycle we have developed four further hypothesis. The first of these is concerned with better understanding the nature of Quality Knowledge as defined by Missier:

H2 There are distinct aspects of Quality Knowledge, based on the sources of information required for their assessment, that mean we can propose common techniques and infrastructure for those aspects that can be used inform effective IQ solutions for the Web of Data.

In response to this we have elicited three aspects of Quality Knowledge: objective, predictive, and subjective. The work of Missier previously focused on the subjective aspect. In this thesis we have chosen to focus on the objective and predictive. The specific hypotheses we investigate are:

H3 The checklist-based approach of MICs provide an existing example of objective Quality Knowledge that we can exploit to develop reusable quality

components in the Web of Data that can be used to successfully evaluate the quality of Linked Data.

H4 There is a predictive aspect of Quality Knowledge that we can model using Bayesian Networks to create reusable quality components in the Web of Data that can support the replication of existing metrics, and support the approximation of those metrics in the face of incomplete metadata.

H5 We can exploit the prevalent use of provenance in predictive Quality Knowledge to support the automatic generation of Bayesian Network-based quality components that can be reused to successfully assess the quality of data with similar provenance.

1.6 Methodology and Approach

The research method used for this thesis consists of three primary components: A literature review and analysis of related IQ frameworks and methodologies to support objectives A1 and A2 and hypothesis H1; theoretical work and modelling to support objective A3 and hypothesis H2, H3, H4; and model implementation and experimental evaluation in order to support objectives A4, A5, A6 and hypotheses H2, H3, H4.

Survey based methods including structured interviews and structured questionnaires were considered as a complimentary activity for objective A1. Structured surveys have been conducted previously in IQ investigations in other domains [WS96][KSW02]. We felt instead that there was sufficient evidence in the literature to establish the scientists' view on IQ. Indeed it follows from our previous discussion that we believe Quality Knowledge exists in metrics and behaviours readily observable in existing infrastructure and the literature. Therefore, whilst discussions were had with colleagues in the sciences both informally and formally⁴, the focus of this thesis is to establish techniques based upon existing work.

Figure 1.5 provides the reader with a guide to the structure of this thesis, beginning with this **Introduction**.

⁴A limited number of unstructured interviews were conducted whilst the author was a Visiting Student at the Massachusetts Institute of Technology. The interviews served to confirm aspects of the data quality issue. The interviews also highlighted to the author the prevalent use of Wikis and Wikipedia in the Sciences.

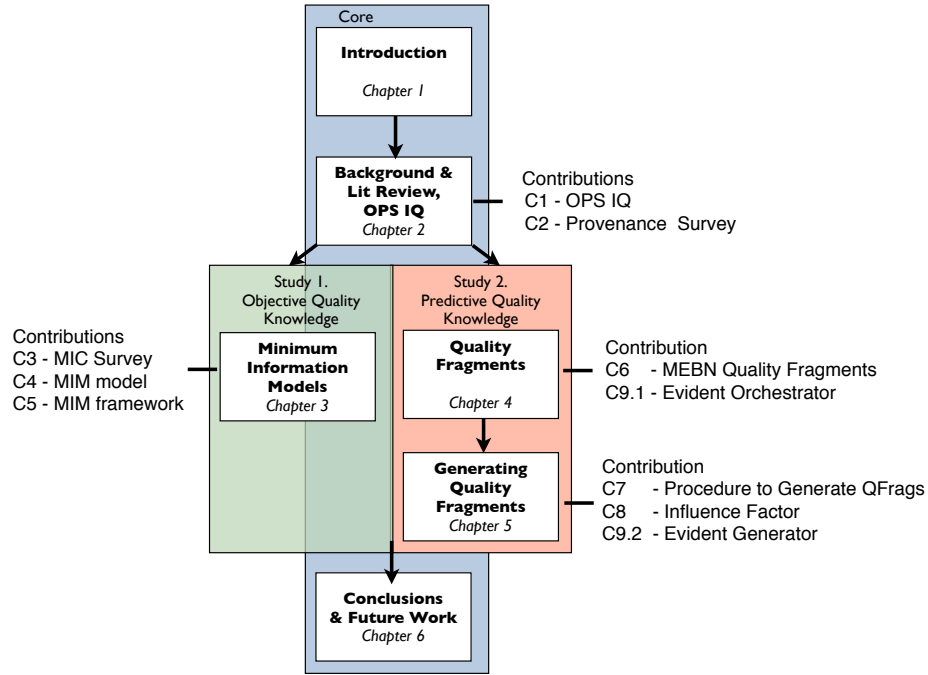


Figure 1.5: A Readers Guide to the Structure and Dependencies in this Thesis

The **Background and Literature Review** allowed us to identify common features of the scientists approach to IQ. Based upon this understanding we have developed our Objective, Predictive and Subjective classification of IQ dimensions (OPS IQ). In line with previous IQ studies [SGTS07] we have used our classification to guide the development of specific IQ assessment solutions. From our classification we established two further areas of investigation in support of objective A3. We have tackled these two areas of investigation with two distinct and parallel studies (shown in Fig 1.5):

- Study 1. Investigating Objective Quality Knowledge with the exploitation of *Quality Standards* in the Web of Data.
- Study 2. Investigating Predictive Quality Knowledge with a probabilistic approach to modelling *Quality Fragments*.

Study 1. Objective Quality Knowledge Using our classification we identified an opportunity to apply MICs, an existing Quality Standard, to the Web of Data. To create reusable quality components (objective A3) we have proposed the Minimum Information Model (MIM), a meta-model suitable for encoding MICs. The design of this meta-model has been informed by an analysis of currently

available checklists. To validate the model and its suitability to the Web of Data (objective A4) we have developed a vocabulary and supporting framework using current Semantic Web and Linked Data technologies. Finally we have performed an evaluation, exercising the framework to assess the chemical compound data available on DBpedia and Wikipedia (objective A5). We compare the merits and limitations of our approach with existing alternative approaches (objective A6).

Study 2. Predictive Quality Knowledge The second of our investigations explores the development of a general approach to modelling predictive Quality Knowledge. To develop reusable quality components (objective A3) we have translated the modelling problem into a the domain of Bayesian Networks. Bayesian Networks are a well understood declarative representation for encoding domain knowledge about uncertainty. To apply this approach to the Web of Data (objective A4) we have extend PR-OWL2, a Bayesian Network encoding developed for the Semantic Web. In order to evaluate this approach we have designed and implemented the *Evident* framework. Evident is an extension to the UnBbaves framework, built to support PR-OWL2.

Our final piece of work explores the role of provenance in predictive Quality Knowledge. We have proposed and implemented a procedure for automatically building Bayesian Network-based quality components. We evaluate this approach by comparing results from a Bayesian Network automatically generated by our approach, against an existing hand crafted Bayesian Network metric (objectives A5 and A6).

1.7 Research Contributions

In this section we detail specifically the novel contributions to the state of the art that have resulted from this research.

C1 OPS IQ: An assessment-oriented Information Quality classification

The first of our contributions is our Objective, Predictive and Subjective classification. As an assessment-oriented classification definitions of objective, predictive and subjective are given in terms of the sources of information required for their assessment. We also propose a series of recommendations and considerations for realising IQ solutions for each of the classes

of assessment. Finally, we use our classification to classify a series of 26 traditional IQ dimensions drawn from 12 existing IQ studies.

C2 An assessment of Provenance Usage in the Web of Data

Provenance plays an important role in our approach to automatically building Quality Fragments. We have repeated a previous study [Har09a] conducted in 2009 to identify the trends in provenance usage in Semantic Web and Linked Data datasets. The assessment demonstrates that there is a continued and increasing usage of provenance metadata.

C3 A Structural analysis of current Minimum Information Checklists

As part of the development of a meta-model for MICs we performed the first large scale structural analysis of MICs. We analysed 41 of the 65 MICs currently available from bio-sharing.org and identified the common structural features of Minimum Information Checklists.

C4 A meta-model for describing Minimum Information Checklists

The Minimum Information Model (MIM) is a meta-model informed by our structural analysis suitable for representing MICs. We provide an implementation of the model, the MIM Vocabulary (MIMv), built using RDFS and OWL2 Semantic Web technologies. Description of the checklists are agnostic to the representation of the data that will be assessed against it, meaning that the same checklist can be used with different datasets.

C5 A prototype framework for assessing existing Linked Data against Minimum Information Checklists

We have produced a prototypical framework in order to evaluate the MIM vocabulary. The MIM framework provides an implementation of the semantics for MIM checklist satisfaction. These semantics are encoded in the SPARQL Inferencing Notation (SPIN) and the framework itself is implemented as a Web service.

We demonstrate the application of the framework with a case-study evaluating a Linked Data extraction of the chemistry data available in Wikipedia.

C6 Modelling Quality Fragments using Multi-Entity Bayesian Networks

We have demonstrated the suitability of Multi-Entity Bayesian Networks to build Quality Fragments. To encode these for use in the Web of Data

we have used the PR-OWL2 vocabulary and extended it with our own *Evident* vocabulary. To the best of our knowledge we are the first to exploit template-based Bayesian Networks for the task of quality assessment in the Web of Data.

C7 An approach to procedurally building Quality Fragments using provenance

We have proposed a procedure to analyse a provenance graph for a piece of data, and automatically generate a Bayesian Network-based Quality Fragment to predict its likely quality. The key contribution of this work is to establish the metadata and core features required to support such a procedure.

C8 Influence Factor: A quantitative measure of influence for the PROV model

A further contribution that emerged from this investigation was an extension to the W3C PROV model representing provenance. We have proposed `evident:influenceFactor` after identifying a gap in the model for describing the *degree* to which one provenance entity influenced another. We demonstrate through our investigation the role this plays in automatically computing quality or trustworthiness from provenance data.

C9 Evident: A prototype framework for supporting Quality Fragments

The Evident is framework designed and implemented as an extension to the existing UnBBayes framework. The framework is in two parts, the Orchestrator and the Generator. The Orchestrator extends the execution stage of UnbBayes to support Quality Fragments. The Generator implements the procedure to automatically build Quality Fragments from provenance data. These automatically generated Quality Fragments can then be executed by the Orchestrator.

1.8 Publications and Research Activity

The following publications and research activity were conducted during the course of this project. The reviews, feedback, and discussions that resulted from these

publications and activity had a valuable influence on the outcomes of this thesis.

1.8.1 Publications

In the following publications the author of this thesis was the primary contributor, or made a significant contribution to work and text presented.

- P1** Matthew Gamble, Jun Zhao, Graham Klyne, Carole Goble. *MIM: A Minimum Information Model Vocabulary and Framework for Scientific Linked Data*. Proceedings of the Eighth IEEE International Conference on eScience, 2012.
- P2** Matthew Gamble, Carole Goble. *Quality, Trust, and Utility of Scientific Data on the Web: Towards a Joint Model*. Proceedings of the International Conference on Web Science 2011 (WebSci11).
- P3** Matthew Gamble, Carole Goble. *Standing on the Shoulders of the Trusted Web: Trust, Scholarship and Linked Data*. Proceedings of the International Conference on Web Science 2010 (WebSci10): Extending the Frontiers of Society.
- P4** Jun Zhao, Graham Klyne, Matthew Gamble and Carole Goble. *A Checklist-Based Approach for Quality Assessment of Scientific Information*. 3rd International Workshop on Linked Science 2013 (LISC2013).

In the following publications the author of this thesis was not the primary contributor, but made a supporting contribution to part of the work or text presented.

- P5** Sean Bechhofer, Iain Buchan, David De Roure, Paolo Missier, John Ainsworth, Jiten Bhagat, Philip Couch, Don Cruickshank, Mark Delderfield, Ian Dunlop, Matthew Gamble, Danus Michaelides, Stuart Owen, David Newman, Shoaib Sufi, Carole Goble, *Why Linked Data is Not Enough for Scientists*. Future Generation Computer Systems, Volume 29, Issue 2, February 2013, Pages 599-611.
- P6** Sean Bechhofer, David De Roure, Matthew Gamble, Carole Goble, Iain Buchan. *Research Objects: Towards Exchange and Reuse of Digital Knowledge*. The Future of the Web for Collaborative Science (FWCS 2010)(WWW 2010).

1.8.2 Research Activity

- Visiting Student with Prof. David Karger at the Computer Science and Artificial Intelligence Laboratory (CSAIL), Massachusetts Institute of Technology. November 2011 to June 2012.
- Invited presentation, “MIM: A Minimum Information Model Vocabulary and Framework” to the W3C Health Care & Life Sciences Group. February 2012.
- Invited poster: “Trust, but Verify: Trusted Data Sharing in Long-tail Collaborative Science on the Semantic Web”, Web Science: A new frontier, the Royal Society. September 2010.
- Invited panel speaker, “Metrics & Measurement”. Online Information Conference 2012.
- Invited panel speaker, “The Future of Digital Scholarship.” The British Library, 3 May 2011.
- Matthew Gamble, “Altered states for the lawmakers of the new measurement frontier” - Article about Altmetrics for Times Higher Education, 28 July 2011.
- Program Committee. ACM Web Science Conference 2011, 2012.

1.9 Thesis Organization

The rest of this thesis is presented as shown in Figure 1.5. Chapter 2 presents our literature review and the subsequent IQ classification, OPS IQ. Chapter 3 then presents the first of our studies investigating objective Quality Knowledge. We describe the Minimum Information Model (MIM) Vocabulary and Framework, the first novel contribution to result from our IQ classification. We also present the results of an evaluation of the MIM framework using data from the chemical articles in Wikipedia. We discuss the potential of our approach and limitations and compare with alternative approaches to validating Linked Data.

Chapter 4 introduces the second of our studies investigating predictive Quality Knowledge. We describe our probabilistic approach to implementing Quality

Fragments using template-based Bayesian Networks. We then evaluate our approach by modelling the GAQ score metric and using it to assess the Bio2RDF Linked Data representation of data from the GOA database.

Chapter 5 continues with our probabilistic approach and details our procedure to automatically build Quality Fragments using provenance data. We evaluate the procedure using provenance data for Wikipedia articles. We compare a Quality Fragment automatically generated by the procedure with an existing Bayesian Network based metric from the literature.

Finally in Chapter 6 we draw together conclusions from our two investigations and propose an agenda for future work.

Chapter 2

Background

“ *Trust, but verify.* ”

- *Ronald Reagan, 40th President of the United States of America.*

2.1 Chapter Introduction

The goal of this chapter is to present the state of the art in Information Quality (IQ) research and present our conceptual classification of Quality Knowledge. We begin the chapter by providing an overview of IQ research beginning with IQ Methodologies that have given rise to much of the future research into IQ, and ground our approach and terminology in the work of Missier and the IQ Life-cycle. We then outline the specific challenges and opportunities to IQ assessment posed by scientific data, and the Web of Data. We establish our conceptual IQ classification by examining the features of scientific Quality Knowledge how to best model and exploit these in the Web of Data. Finally we review the state of the art in IQ assessment on the Web and the Web of Data. In our review we identify the principal components of existing Web-based IQ assessment solutions, and highlight practices that can inform our own IQ assessment solutions.

An earlier version of the IQ classification presented in this chapter was previously published as part of: Matthew Gamble, Carole Goble. *Quality, Trust, and Utility of Scientific Data on the Web: Towards a Joint Model*. Proceedings of the International Conference on Web Science 2011 (WebSci11).

2.2 What is Data Quality?

The issue of Data or Information Quality (IQ) is one that is prevalent not just in the scientific domain, but any domain where critical decisions rely on high quality data. Research into methodologies to detect, assess, and improve data quality has been conducted in fields such as statistics, business management [WS96] and computer & information science [BCFM09]. The scope and overall goal of these methodologies is varied. IQ methodologies draw upon a variety of definitions of quality to inform their modelling. The following three definitions are representative of the range of definitions typically chosen:

- The most frequently adopted is that of Juran, who describes data quality simply as “fitness for use” [Jur74].
- A definition from Redman [Red01] defines high quality data as “data that are free from defects”.
- The ISO 9001 standard [ISO08] defines quality as “the degree to which a set of inherent characteristics fulfils requirements”.

Many early methodologies take a predominantly theoretical approach in an effort to establish a broad conceptual model of IQ [WS96]. Others instead take a more practical approach and look to establish methodologies informed by the practical process of assessing IQ [NR00], considering for example what data is needed, where that data comes from, and at what point in the process. Some methodologies specialize further and are referred to as *situation specific* methodologies. These methodologies typically focus on a particular platform or type of data, for example information on the Web [KB05], or biological databases [MH05]. Despite this diverse scope of study, common to almost all IQ research is the characterization of IQ as a multi-dimensional concept. As a multi-dimensional concept IQ becomes the aggregation of multiple information quality *dimensions* [Nau02][Red01] [Jur74] [Wan98].

Quality dimensions are the abstract definitions of a particular aspect of quality, for example:

Accuracy is “the degree to which information represents the states of the real world” [WS96].

Timeliness is “a measure of whether data is timely enough to be used for a specific task” [PLW02].

Completeness is “the degree to which data are not missing in a schema” [PLW02].

These dimensions and others such as reputation, consistency, relevancy etc. can then be applied to data in isolation to measure one particular aspect of IQ, or combined to gain an aggregate view of the data’s quality. The application of dimensions is achieved through the development of *quality metrics*.

Quality metrics such as the previously discussed GAQ score, can be viewed as instantiations of these IQ dimensions that assess one or more dimensions for a particular type of data.

As an example of this process consider the GAQ metric and the dimensions of specificity and accuracy. Buza et al. [BMW⁺08] identify specificity and accuracy as two abstract dimensions relevant to assessing the quality of gene annotations. The authors then go on to instantiate those dimensions in the GAQ metric, using the GO term depth as a specific instantiation of specificity, and the evidence code rank as an instantiation of accuracy. The metric itself then provides a scoring function for gene annotations. The task of a quality metric is therefore to take a type of data and provide a scoring or classification function based upon one or more IQ dimensions relevant to that data.

Some quality dimensions are specific to a particular domain such as *liveness* as a measure of the ability to access a Web resource [ZKGG13], or *impact* as a measure of the quality of scholarly publication [PPH12]. There is however significant agreement on a core set [BCFM09] (see Table 2.1 later in the chapter which details a series of common dimensions from the literature). The elicitation, definition, and subsequent classification of quality dimensions is central to many of the established IQ methodologies.

Methodologies and solutions to IQ assessment are strongly affected by the definition of quality that they choose to adopt. Figure 2.1 illustrates how we can judge metrics along two axes

- Applicability - a measure of how broadly reusable the metrics are.
- Effectiveness - how discriminating they are in their quality assessment.

Approaches that adopt the Juran definition focus on the *use* element of “fitness for use”. This typically leads to the assertion that the quality of data cannot

	Broadly Applicable	Narrowly Applicable
Effective	Desirable	“Fitness for use”
Ineffective	“Free from defects”	Undesirable

Figure 2.1: Two-dimensional matrix illustrating a spectrum of applicability and effectiveness of IQ metrics.

be assessed independently of the data consumer [WS96] and that information quality is entirely subjective to the user, and the context in which the data is to be used [Biz07] [KB05] [Wan98]. IQ solutions resulting from this assumption are often effective at measuring IQ for the data or user in question, but are narrowly applicable, tailored to a very specific use [Mis08].

Solutions drawing from Redman’s “free from defects” definition often apply broad and general techniques for identifying defects in data [FH10], such as identifying duplicate records, or syntactical errors. This leads to a broadly applicable IQ assessment but due to not being tailored to the context, data, or user these ‘one-size-fits-all’ metrics are often limited in their effectiveness [PME⁺08].

The ISO9001 definition highlights what we view as a subtle but important variation. By defining the measure of quality as *requirements* as opposed to intended *use*, this suggests that requirements can be defined and exist independently of an individual.

Whilst it is clear that there is a subjective element to IQ assessment, in parallel we also observe that there many instances where there is an agreement within a community about quality, in the form of a standard. Standards allow these requirements to be defined objectively (or more accurately inter-subjectively¹). This observation has repercussions for how we develop solutions to assess IQ. Moreover, the scientific community is particularly well placed to benefit from this community defined quality, where there is an established practice of using community defined norms and standards for data. To develop effective solutions

¹Intersubjectivity is a term from the field of philosophy used to describe a shared understanding. It is defined in [Sch06] as “the sharing of subjective states by two or more individuals”

we require a methodology that can support us in identifying these objective and subjective elements. The challenge is to identify the IQ requirements that can be defined objectively and exist independently of an individual, but in turn identify when and how to support the subjective requirements that are required.

Existing information quality methodologies can be broadly viewed in two classes, organisational methodologies such as the Total Data Quality Management (TDQM) methodology [Wan98] or Web-based such as Naumann's Subject-Process-Object methodology [Nau02]. For organisational methodologies the focus is typically on intra-organisational data quality issues where the goal of the assessment is to inform quality-control procedures for large-scale monolithic information systems. TDQM provides an early, influential approach to modelling the data consumer's view of information quality and partitions data quality dimensions into four classes:

1. **Intrinsic data quality** - accuracy, objectivity, believability, reputation;
2. **Contextual data quality** - relevancy, value-added, timeliness, completeness, amount of data;
3. **Representational data quality** - interpretability, ease of understanding, concise representation, consistent representation; and
4. **Accessibility data quality** - accessibility, access security.

This classification is termed a *semantic-oriented* classification [Nau02], grouping together dimensions based upon some perceived semantic similarity. The classification demonstrates no clear separation of objective and subjective dimensions. Dimensions such as believability, relevance, and ease-of-understanding require some subjective interpretation by user, and appear across each of the defined classes.

In the context of Web-based information systems the goal is to support the data consumer in assessing the quality of data, and ultimately decide if the data are of sufficient quality to meet their needs. In Web-based methodologies, data producer and consumer are typically unknown to each other, which introduces some uncertainty. This introduction of uncertainty dictates that *trust* often plays a role in information quality assessment [GA07] for Web-based IQ assessment. Prior work in Web-based assessment often adopts a primary concern of either quality [Biz07] [KB05] or trustworthiness [GA07] [KFW08].

In contrast to semantic-oriented classifications, the Subject-Object-Process (SOP) methodology instead presents an *assessment-oriented* classification. In the SOP classification quality dimensions are classified not by semantic similarity, but instead by considering the source of information that is required to assess the dimension.

The SOP methodology identifies three sources of information for quality assessment; *subject* - the user who is making the quality assessment, *process* - the query mechanism that is orchestrating the quality assessment, or *object* - the data that is being assessed. IQ dimensions are then classified with respect to the source of the metadata and criteria needed to make a quality assessment. SOP classifies dimensions as follows:

- **Subject** - Believability, Concise representation, Interpretability, Relevancy, Reputation, Understandability, Value-added.
- **Object** - Completeness, Customer Support, Documentation, Objectivity, Price, Reliability, Security, Verifiability, Timeliness.
- **Process** - Accuracy, Amount of data, Availability, Consistent representation, Latency, Response time.

By considering the actors and entities involved in IQ assessment, this approach goes some way to separating objective and subjective dimensions. A further advantage of an assessment-oriented IQ classification is that, given its practical focus on the active process of IQ assessment, it can more readily guide the modelling of future quality assessment solutions. For this reason we have chosen an assessment-oriented approach to guide our own IQ classification described later in this chapter (section 2.4).

2.2.1 The IQ Life-Cycle

The conceptual framework for IQ most closely related to our objectives was developed in the thesis of Missier [Mis08]. Missier has studied the assessment of IQ in eScience, with a focus on Workflow based *in silico* experimentation. The scientific context of the study means that Missier provides a useful and appropriate separation of concerns, and with it a conceptual framework that informs the work in this thesis. This section defines much of the terminology used throughout this thesis to describe to concepts involved in IQ assessment. Figure 2.2 provides a

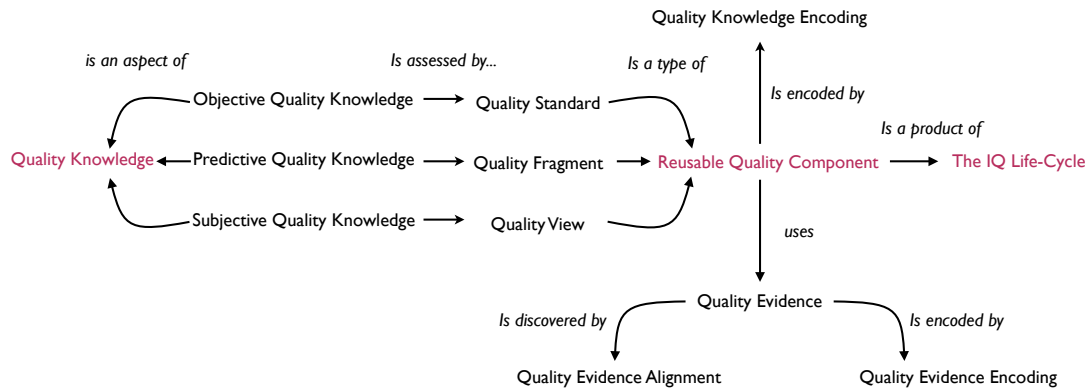


Figure 2.2: Map of Concepts used in this Thesis

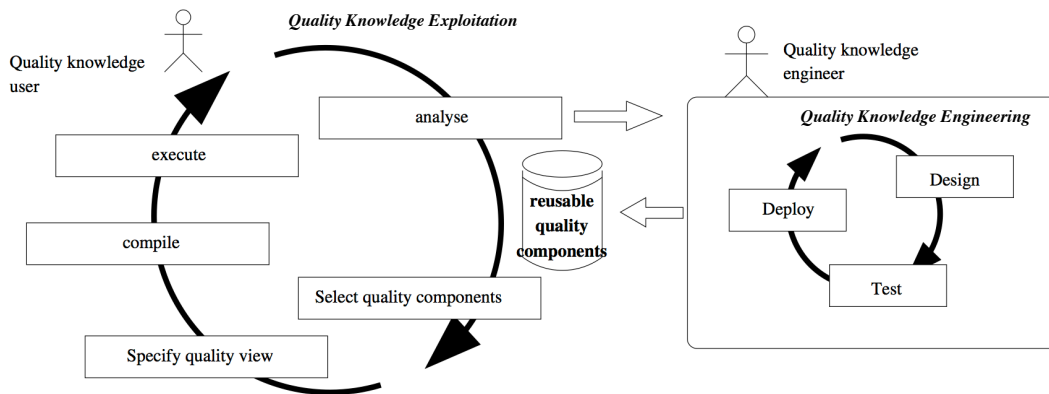


Figure 2.3: The Information Quality Life-cycle [Mis08]

concept map to illustrate relationships between the concepts defined. We have also included a glossary of quality related terms at the start of this thesis.

At the core of Missier's approach are three concepts, *Quality Knowledge*, *reusable quality components*, and *the Information Quality Life-Cycle*.

- **Quality Knowledge** is the latent knowledge that scientists make use of when critically evaluating their data, and making decisions about its quality. e.g. The knowledge that term depth in the GO ontology relates to specificity.
- A **reusable quality component** is the Quality Knowledge realized as a software component that can be used to assess the quality of data. e.g. A Perl script that encodes the GAQ metric that can be reused for any set of

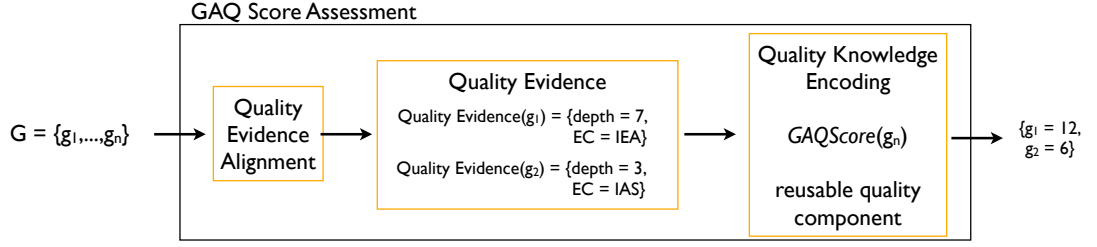


Figure 2.4: Abstract Representation of the GAQ score Assessment processes using a Reusable Quality Component for the GAQ Score

gene annotations data.

- **The IQ Life-cycle** is the process of designing and creating quality components, and making use of those components to assess the quality of data.

The IQ life-cycle (shown in Fig. 2.3) has two distinct and interacting components - *Quality Knowledge Engineering* and *Quality Knowledge Exploitation*. In this thesis we focus primarily on the process of engineering and the development of reusable quality components for the Web of Data.

Quality Knowledge Engineering is the concern of an engineer who is engaged in a cycle of modelling activity. The role of the engineer is to encode domain expert knowledge by *designing*, *testing* and *deploying* software components that can be reused to assess the quality of data. In the case of [Mis08] these reusable quality components are Web Services that provide some information about quality. An example would be a Web Service that provides an implementation of the GAQ score. To develop reusable components the task of the engineer is threefold. Given some class of data such as Gene Annotations $G = \{g_1, \dots, g_n\}$:

1. Decide which metadata attributes of G are to be used as *evidence* for the quality assessment. For our Gene Annotation example this is the depth and evidence code metadata.
2. Design and *encode* a scoring or classification of the data based upon the evidence, and Quality Knowledge.
3. Evaluate the performance of the quality component

Figure 2.4 shows an abstract representation of an assessment that makes use of a reusable quality component for the GAQ score. Given a set of gene annotations

G , the first task is to discover the metadata attributes that are to be used in the assessment. We refer to this process as ***Quality Evidence Alignment***. A more general definition is that it is the process of annotating input data so that the required metadata can be discovered by the quality component to be used in assessment. The annotated data is referred to as ***Quality Evidence***. Missier does not directly address the issue of alignment and treats the process as a “black-box”. The Web of Data is however characterised by its heterogeneous representation and as a result we see it necessary to directly address the issue in this thesis.

We choose to extend Missier’s terminology relating to quality evidence by introducing the term ***Quality Evidence Encoding***. This refers to the mechanism used to describe the evidence, such as the specific Semantic Web vocabularies. Alignment is then the active process using an encoding to describe evidence.

The next stage is to take the evidence and pass it to the quality component to perform a quality assessment. The assessment can be a scoring on a continuous scale, such as the GAQ metric, or a classification of the data, dividing it into discrete classes such as *accept* and *reject*.

We introduce ***Quality Knowledge Encoding*** to refer to the specific mechanism that is used to encode the quality component. This encoding can be either a declarative or procedural mechanism. A Web service for example can be encoded in a procedural mechanism such as the Java programming language. Once encoded it can be made available to the user to make quality assessments.

Quality Knowledge Exploitation is the task of the user, who uses quality components to assess, and ultimately make decisions about the quality of data. This exploitation is achieved through the specification of **Quality Views**. The terminology is borrowed by Missier from the domain of relational databases to highlight that they are somewhat analogous to database views, a pre-established component that can provide a custom interpretation over the data. To define Quality Views, users describe a policy that combines reusable quality components with their own subjective requirements. Figure 2.5 illustrates an abstract example of an assessment using a Quality View which makes use of the GAQ score Assessment and introduces a filter to remove gene associations with a score less than 10.

The two assessments in Figures 2.4 and 2.5 are composed of several stages of

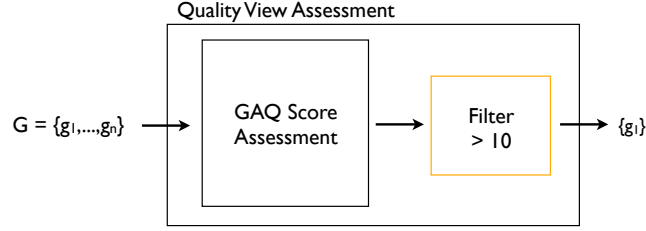


Figure 2.5: Abstract Representation of a Quality View reusing the GAQ Score Assessment

computation that require some co-ordination. We introduce the term **Orchestration** to refer to the mechanism used to manage this co-ordination. Missier for example makes use of a scientific Workflow engine as an Orchestration mechanism.

Missier’s previous work focused primarily on Quality Knowledge Exploitation, defining a semantic model of IQ in the form of an Ontology, and an XML based policy language to encode Views. The ontology provides a structure to describe IQ concepts and facilitates the description and advertising of existing quality components to users. The policy language allows the user to encode their requirements in a machine readable manner. They can be compiled into workflows to make a quality assessment, and shared as additional quality components. The XML policy language is an example of a declarative encoding, in contrast to the procedural Java Web-services.

The work in this thesis complements the work of Missier by exploring the use of the IQ Life-cycle to bring reusable quality components to the Web of Data. We identify two desirable characteristics of reusability currently present in the quality components proposed by Missier that we aim to maintain with our solutions:

- **Direct Reuse:** Given data of the same type, and represented in a consistent manner, the quality component can be reused to assess the quality of that data.
- **Modularity:** Components can be used as part of more complex assessments, for example the use of the GAQ score as part of a Quality View.

In the next section, we examine the Web of Data and identify a number of opportunities and challenges to building reusable quality components. We examine Quality Knowledge in a *assessment-oriented* manner to identify the distinct

information sources required, and understand how to exploit these in the Web of Data.

2.3 Quality Knowledge and the Web of Data

2.3.1 The Linked Data Approach

The immediacy and scale of the Web coupled with the exponential increase in the volume of scientific data [HTT09] has affected the community’s ability to effectively judge the quality and trustworthiness of that data. Traditional mechanisms of quality control have been disrupted as data sharing platforms both large scale [Dat; PSS⁺05; BAW⁺05] and “long-tail” [DGS09; Ope12] have enabled scientists to share their data. These platforms are increasingly making use of the Web of Linked Data for sharing and discovery. Linked Data is of particular utility to the Life Sciences community who have a frequent need for data integration. The goal of the Linked Data approach is to create a global information space through the use of Web and Semantic Web technologies. In 2006, Tim Berners-Lee described a set of four ‘rules’ for publishing data on the Web, known now as the Linked Data principles [BL06]:

1. Use URIs as names for things.
2. Use HTTP URIs so that people can look up those names.
3. When someone looks up a URI, provide useful information, using the standards. (RDF, SPARQL)
4. Include links to other URIs, so that they can discover more things.

Like the traditional Web, the Web of Data is made from documents and links between those documents. In the case of the Web of Data these documents are machine-readable documents encoded in RDF and the links are HTTP URIs.

The Listing below shows an example of simple RDF document that contains a representation of a gene annotation:

```
1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
2 @prefix : <http://sierra-nevada.cs.manchester.ac.uk/goa/GDB> .
4
4 <http://sierra-nevada.cs.manchester.ac.uk/goa/GDB#GDB_121> rdfs:type <http://
  bio2rdf.org/goa_vocabulary#GO-Annotation> ;
```

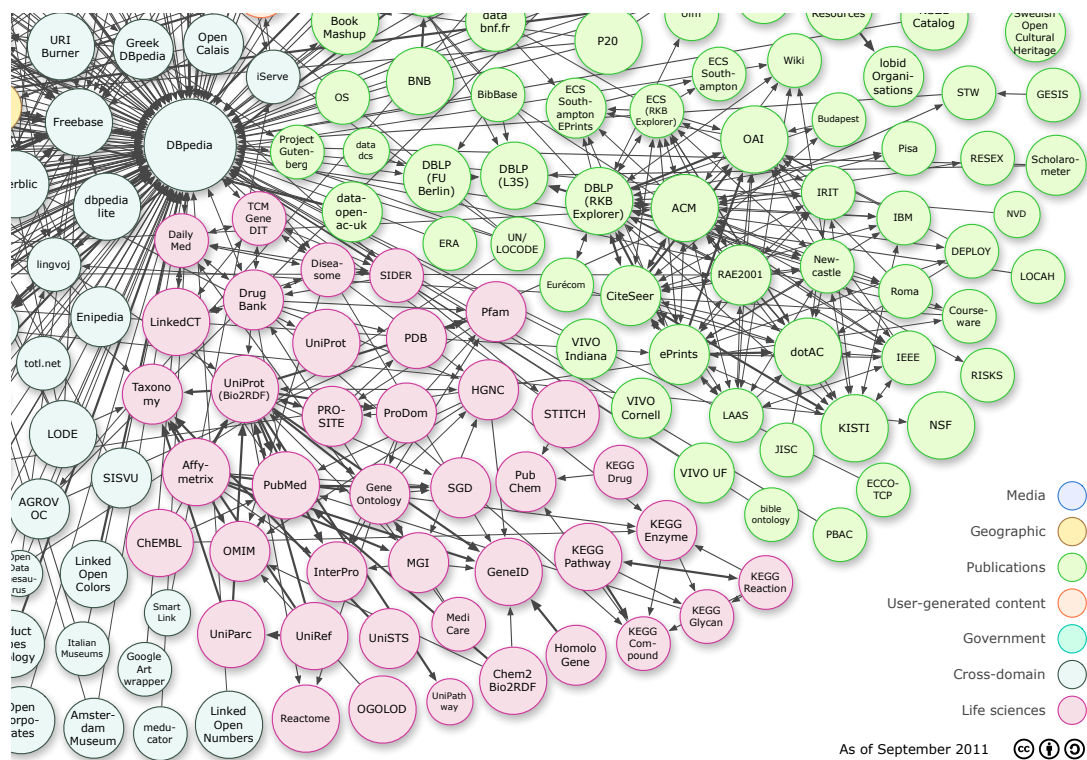


Figure 2.6: The Linking Open Data cloud diagram from [CJ11]

```

5 <http://bio2rdf.org/goa_vocabulary#evidence> <http://bio2rdf.org/eco
  #0000304> ;
6 <http://bio2rdf.org/goa_vocabulary#go-term> <http://bio2rdf.org/go
  #0005634> .

```

The URI `<http://sierra-nevada.cs.manchester.ac.uk/goa/GDB>` is used to represent the RDF document itself and `<http://sierra-nevada.cs.manchester.ac.uk/goa/GDB#GDB_121>` identifies the particular gene annotation that is being described in the document. The utility of Linked Data comes from using resolvable URIs to represent data, for example the GO term that is used by this annotation is represented as `<http://bio2rdf.org/go#0005634>`. This means that a human or machine can follow the URI and access another RDF document that provides further information about that GO term.

Life Sciences datasets make up a significant part of the Linked Open Data (LOD) cloud [CJ11], a series of 295 Linked Data datasets made available in the Web of Data using open licenses. Figure 2.6 is part of an illustration of the LOD cloud made available periodically by the maintainers of the collection, and highlights the Life Sciences datasets available as of September 2011.

For the scientific community there are a number of motivations to publish data into the Web of Data [MBD⁺12]:

Shareability - A *data provider* or *data producer* wants to make their data openly available in a machine readable manner so that a *data consumer* can reuse that data.

Integration - Many datasets in the Life Sciences contain complementary or overlapping data that can be merged and linked at the point of related concepts or IDs.

Discoverability - A scientific user would like to align their data with information in the Web of Data to discover related resources.

For our IQ assessment we are particularly interested in the shareability scenario, and supporting the role of the data consumer. Scientific data exchange in the web is often *asynchronous* [LMS⁺05] where a *data provider* lies between the *producer* and *consumer*. The producer deposits data with the provider, which may oversee its curation or modification, and the consumer accesses the data some time later. We believe that all three stand to gain from improved understanding and improved mechanisms for IQ in scientific Web data.

The W3C Healthcare and Life Sciences (HCLS) Interest Group has recently published a series of recommendations for publishing Life Sciences data into the Web of Data [MBD⁺12] in response to the broad adoption of Linked Data. Amongst them are recommendations on:

- How to create RDF representations of the data and link it to existing datasets.
- The use of ontologies and vocabularies to provide rich descriptions and metadata about the data being published.
- The inclusion of *provenance* metadata, information about where the data came from, who produced it, under what conditions etc.

Specific approaches are also being developed to better support the publication of a unit of scholarly information [BBR⁺13]. The Object Reuse and Exchange (ORE) specification [LVdSJ⁺07], Research Objects [BDRG⁺10] and Nano Publications [GGV10] act as containers to encapsulate and enrich scientific information. Scientific Linked Data is not then just the raw data itself replicated in the

Web of Data, but also a rich set of metadata, provenance data, and links to other data resources where further information may be available.

The result of this open and Web-scale approach to data sharing is a data landscape with a wide spectrum of moderation and control. Despite guidelines such as those from the HCLS, often there is no agreement on how a particular type of data should be represented, and which vocabularies to use. Indeed due to the dynamic and rapidly changing nature of the Life Sciences, there may not yet be standard ontologies or vocabularies available to describe the data. Publishers often therefore need to define their own.

The method by which a Linked Data dataset is produced can also affect its quality. In some cases the data are produced directly as RDF documents. More often though for scientific data the data is extracted from another existing representation or relational database, into RDF [MBD⁺12]. Due to this varied data production pipeline the metadata and provenance available data are not always consistent across the Web of Data.

The Pedantic Web Group² have studied empirically the impact of the publication pipeline on the quality of RDF data. Studies of 50,000 RDF documents in 2010 [HHP⁺10], and 4 million RDF documents more recently in 2012 [HUH⁺12], have highlighted a catalogue of common errors introduced into RDF documents due to poor serialization and publication practices. These poor practices lead to data that is *incomplete* - with supporting metadata data missing, *inconsistent* - where for example the same data is represented with different datatypes, and *incoherent* - due to the incorrect use of RDF and other Semantic Web publishing standards. These errors go on to impact consumers of those data sets and their ability to effectively exploit the data.

The Open PHACTS project [WHG⁺12] for example is bringing a number of pharmacological datasets into the Web of Data and in turn developing novel approaches to integration [GGL⁺12]. The project has faced challenges relating to poor quality data, particularly with data that has been converted into an RDF representation, and has unknowingly been out of date with respect to its source. The project has therefore found it necessary to improve the provenance metadata that describes how and when data have been extracted [BEG⁺13].

The Web of Data therefore brings both opportunities and challenges to the task of IQ assessment. In particular we identify two key opportunities and two

²<http://pedantic-web.org>

key challenges.

Opportunities The most important opportunity is that the data and metadata are consistently available (with regards to serialisation) in a machine readable fashion. This makes it possible to consider mechanised, reusable approaches to IQ assessment. The prevalent use of metadata and provenance data is also a clear benefit where Quality Knowledge often depends upon not just the data, but also the metadata.

Challenges The two key challenges to creating reusable quality components posed by linked data are:

- The *inconsistent representation* of data and metadata, where different vocabularies and structures are used to represent the same types of data. This has an impact on the process of Quality Evidence Alignment. In order to use a quality component, the data and metadata must be in a format such that it can be discovered.
- The *inconsistent availability* of metadata. Whilst metadata use is prevalent, the amount of metadata included both within the same dataset, and across different datasets can be inconsistent. Even if we assume a perfect alignment, i.e. one that can discover all Quality Evidence if it is available, the inconsistent availability of metadata means that we might still be missing metadata required for a quality assessment.

In response to these challenges, we propose two further desirable characteristics of reusability for quality components:

- **Robust to metadata representation:** Solutions to IQ assessment will need to include robust techniques to identify Quality Evidence.
- **Robust to metadata availability:** We see the need for techniques that can provide a *best-effort* assessment given partial metadata.

In this context best-effort means we aim to develop solutions that can provide an *approximation* of the same quality assessment with complete metadata, such that it might still support the user in making quality based decisions.

2.3.2 Examining Quality Knowledge

A data quality issue caused by autonomous and distributed data publication is not new [Biz07], nor is it unique to Linked Data. We believe though that the scientific community is particularly well placed to manage this quality issue through the exploitation of Quality Knowledge. This domain knowledge can be seen explicitly through the use of IQ metrics such as the previously discussed GAQ score, and through the general practices of research scientists, such as peer-review and curation.

Our analysis of Quality Knowledge is based upon three observations related to a scientist's decision about whether to use data:

1. Is it good given how it compares against norms and *standards*?
2. Is it *likely* to be good given our *prior knowledge* of how its metadata and *provenance* relate to quality i.e. where it came from, who produced it, under what conditions?
3. Is it a good fit to current needs?

These three concerns underpin our approach to Quality Knowledge and IQ assessment. Rather than a single concern, these three interrelated aspects which we term 1) *objective*, 2) *predictive*, and 3) *subjective* form the basis of our quality assessment for Web-based scientific data. In the rest of this section we address each of these aspects and highlight their defining characteristics. In turn we propose three corresponding classes of quality component, *Quality Standards*, *Quality Fragments*, and *Quality Views*.

Objective Quality Knowledge

Is it good when compared to norms and standards?

Central to the *objective* aspect is the role of standards. Standards are a critical component of scientific collaboration, particularly as scholarly activity moves to the Web. They act as a 'social technology' [Edw04], negotiated by the community in an effort to support the asynchronous nature of science on the Web [Zim08]. Communities and moderators define, promote, and adhere to standard data formats, vocabularies and quality standards such as Minimum Information Checklists [FSC⁺09] and domain specific quality requirements [F⁺98].

Consider the situation of online chemical structure repositories such as the ChemSpider repository [PW10], hosted and managed by the Royal Society of Chemistry. It is now community consensus in chemistry that a good quality description of a chemical compound requires that it provide an IUPAC International Chemical Identifier (InChI). An InChI is a textual representation of a chemical compound that provides a unique human and machine readable encoding for all possible compounds. For example the InChI identifier for the chemical compound Ethane is encoded as “1S/C2H6/c1-2/h1-2H3”. This reporting requirement has been subsequently captured in the MIABE (Minimum Information About a Bioactive Entity) checklist [OALB⁺11]. ChemSpider also enforces this requirement and will not allow entries in its repository that do not provide an InChI identifier. MICs therefore provide an objective instantiation of the IQ dimension of completeness, that has been agreed upon by members of the community.

As a further example ChemSpider also evaluates new entries to its database using a set of quality measures called Chemical Structure Validation Filters (CSVFs) [WET12]. These filters detect common inaccuracies in chemical structure data, for example a filter for *Hypervalency*. Hypervalency occurs where a compound representation, such as an InChI, contains an atom that defines more bonds than is theoretically possible. A common example of this is pentavalent carbon, where carbon is described with 5 bonds. This is an issue that is easily defined and detected, but if not checked can lead to errors being introduced into repositories [WET12]. CSVFs therefore provide another objective definition of quality for the dimension of correctness.

Both MICs and the CSVFs define a standard which can be encoded and used to measure the quality of data. Moreover these standards are defined objectively and can be assessed without the need for any additional information or subjective interpretation. As an extension to Missier’s conceptual framework we propose the term ***Quality Standards*** to refer to quality components that make use of objective elements of Quality Knowledge. Figure 2.7 illustrates the process of using a Quality Standard-based assessment. The distinctive feature is that the Quality Knowledge encoding is based entirely upon an objectively defined standard.



Figure 2.7: Abstract Representation of a Quality Standard based Assessment

Predictive Quality Knowledge

Is it likely to be good given our prior knowledge of how its metadata and provenance relate to quality?

The complexity of data can often prevent the creation of an objective standard. For scholarly artifacts such as publications, mechanisms such as peer review and curation have long been ingrained in the scholarly process in an effort to ensure quality and increase trustworthiness. It is difficult though to establish an objective measure of what makes a good paper. It is more difficult still to define such an objective measure that can be mechanised and automated.

In the absence of an objective standard, metrics are often used instead as predictors for the likely quality of data. Journal impact factor, citation counts, and the H-index for example are long standing metrics that have been employed by the community as predictors for the likely quality of research output. Common to these metrics is some prior knowledge about how metadata relates to the quality of that data. For example a higher citation count is generally regarded to correlate with a higher quality paper.

Specifically we make two observations about the predictive aspect of Quality Knowledge: 1) The use of some prior knowledge related to metadata to make predictions about quality, 2) that provenance data often plays an important role in these predictions.

A feature of the predictive aspect that distinguishes it from objective quality knowledge is the presence of uncertainty. We can say for certain for example that a chemical structure description that contains hypervalent atoms is of poor quality. We cannot however say with the same certainty that a publication with a low number of citations of poor quality. Instead we are using prior knowledge to estimate a likelihood of quality.

This prior knowledge can either be established empirically, or through some intuitive knowledge of the data. The Altmetrics movement [PGT12] for example is developing a diverse series of metrics for scholarly artifacts, many of which exemplify predictive Quality Knowledge. These metrics use metadata generated by activity on the Web surrounding scholarly artifacts, to predict its quality, impact, relevance etc. Activity such as PDF downloads, Twitter mentions, bookmarking activity in online paper repositories such as Mendeley [HR08] and Citeulike [cit13], Wikipedia citations, blog mentions, and others are measured to gain a richer landscape of metadata related to an artifacts use.

A core activity of the altmetrics research agenda is to examine existing artifacts to establish empirically the prior knowledge about how this metadata correlates to dimensions such as impact, relevance, importance and reproducibility [PPH12]. This prior knowledge can then be used to make predictions for new artifacts based on the same type of metadata.

More informally, mechanisms such as popularity and prestige have long informed decisions about the quality of research [DC11]. Scientists are likely to rely on prior knowledge and select data from a source known to them or widely regarded as trustworthy [Gio07].

Provenance data plays a central role in making these decisions. Predicting quality based upon Journal Impact factor means that we need to know *where* the paper was published. Using citation count we not only need the number of citations, but information about *when* it was published to provide context for that number of citations. Using authorship we need to know *who* published the paper. Indeed much of the scientific process relies on provenance, knowing who produced what, when, and how [Gol08].

The creators of GeneWiki have applied a metric called WikiTrust [AdAP10] to monitor the quality of the content of their pages on Wikipedia [GCdAS11]. WikiTrust uses provenance metadata about the authorship of previous revisions of a page, to predict the likely quality of the current version of that page.

The GAQ score also exemplifies this predictive behaviour, using provenance metadata about gene annotations, and some prior knowledge about that metadata, to make predictions about their likely quality. Each gene annotation is accompanied by its evidence code, provenance metadata that details the method by which it was generated. The GAQ metric captures prior knowledge about how the evidence code relates to quality using the evidence code rank (ECR). Figure

Code	Code definition	Evidence code rank
IDA	Inferred from Direct Assay	5
IGI	Inferred from Genetic Interaction	5
IMP	Inferred from Mutant Phenotype	5
IPI	Inferred from Physical Interaction	5
IC	Inferred by Curator	4
TAS	Traceable Author Statement	4
IEP	Inferred from Expression Pattern	3
RCA	Inferred from Reviewed Computational Analysis	3
IGC	Inferred from Genomic Context	3
ISS	Inferred from Sequence or Structural Similarity	2
IEA	Inferred from Electronic Annotation	2
NAS	Non-traceable Author Statement	2
NR	Not Recorded	1
ND	No Biological data available	0

Figure 2.8: Evidence Code Ranks (ECRs) as Defined by Buza et al. as an Example of Prior Knowledge

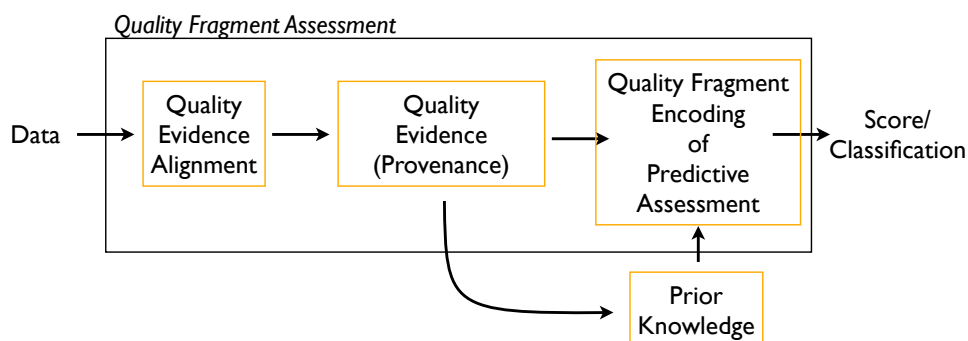


Figure 2.9: Abstract Representation of a Quality Fragment based Assessment

2.8 shows the ECRs developed by Buza et al. for the GAQ metric. This prior knowledge is maintained separately and is used to inform the GAQ score quality prediction.

We observe that prior knowledge used to inform predictive assessments can change over time as we get new information. The ECRs for the GAQ metric for example have been updated as new types of evidence have emerged, and current process have been better understood. This change in prior knowledge is independent of the mechanism used to make the predictive assessment.

We use the term *Quality Fragments* to refer to reusable quality components that make a predictive assessment of quality. Figure 2.6 illustrates the process of an assessment using a Fragment.

Subjective Quality Knowledge

Is it a good fit to current needs?

Our final aspect of quality knowledge takes in to account the user’s subjective needs. A subjective interpretation of quality is aligned with the concept of utility. Utility is a well researched and understood concept in economic theory [Sti50], and game-theory [VNM07], where an individual’s set of preferences are defined as a scoring or classification of some expected outcome. The evidence used in a subjective assessment is often therefore the result of some previous quality assessment to which the user applies their own interpretation.

This subjective view can be seen for example in the varied interpretation of evidence codes by the life sciences community. Evidence codes used for gene annotations broadly fit into two categories - assigned by human curators, or inferred electronically. Scientists using data from GOA database will use the method by which annotations were generated to filter or rank their data [BMW⁺08] - some choose to omit electronically derived annotations completely [SAD12], believing them to be of lower quality.

This subjective interpretation is often the view taken by IQ approaches that adopt Juran’s “fitness-for-use” definition of quality. Figure 2.5 for example demonstrates this type of fitness-for-use scoring, and illustrates that Missier’s conceptual model accounts for the subjective view of IQ. The distinguishing feature is the need to take into account the users requirements at some point during the assessment. We continue to use the term *Quality View* to refer to quality components that make use of subjective quality requirements.

In light of our extension of the concept of Quality Knowledge we note that Missier’s previous work has been primarily concerned with the subjective aspect, developing a general solution to modelling the user’s subjective view of quality using the Quality View XML policy language, and compiling those views into reusable quality components, in the form of scientific workflows. The work in this thesis therefore compliments Missier’s work by instead focusing on the objective and predictive aspects of Quality Knowledge. We investigate approaches to modelling the objective and predictive aspects of the scientific approach to IQ assessment, and develop techniques so that we can realize them as reusable quality components that can be exploited in the Web of Data.

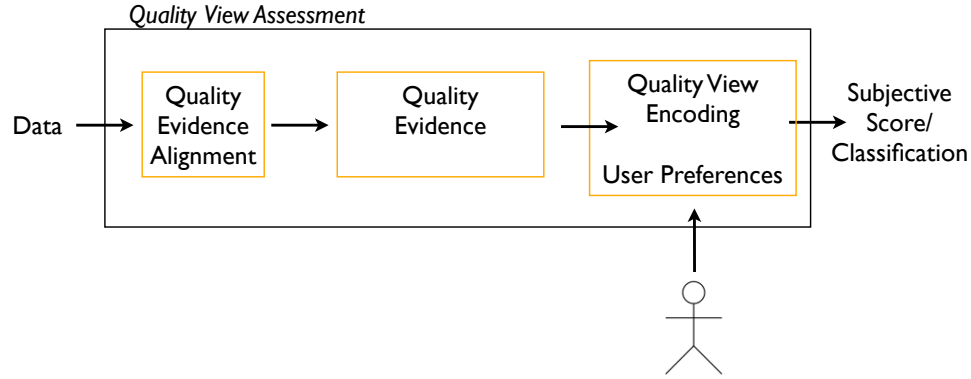


Figure 2.10: Abstract Representation of a Quality View based Assessment

2.4 OPS IQ: an Objective, Predictive and Subjective Classification

In this section we present the OPS IQ classification, our assessment-oriented IQ classification tailored to scientific Quality Knowledge. In the previous section we identified three, prevalent aspects each with a corresponding type of quality component that serves to define the process of its assessment: (1) objective assessed by Quality Standards, (2) predictive assessed by Quality Fragments; and (3) subjective assessed by Quality Views.

Our classification therefore defines three classes of IQ dimensions, one for each aspect: Objective Assessment, Predictive Assessment, and Subjective Assessment. As an assessment-oriented classification these classes are defined in terms of the sources of information required during their assessment.

In total we identify 6 unique information sources: Data, Quality Standards, Quality Fragments, Quality Views, User Preferences and Prior Knowledge. We define three classes of assessment as follows:

Objective Assessment is a function of the *Data* assessed against a *Quality Standard* to provide an objective measure of quality e.g. completeness.

Predictive Assessment is a function of the *Data* against a *Quality Fragment* using *Prior Knowledge* to provide to estimate likely quality e.g. reputation

Subjective Assessment is a function of the *Data* against a *Quality View* using the *User's Preferences* to provide a quality assessment tailored to the user's specific quality needs e.g. relevance.

Using these definitions we propose a series of components and considerations for each type of IQ assessment. We identify three components common to **All** assessments: *Quality Evidence Description*, *Quality Evidence Alignment* and *Orchestration*. We also propose a number of considerations specific to each assessment.

An **Objective** assessment requires an encoding of the Quality Standard that can be subsequently shared, for example a mechanism to encode a MIC as RDF. Unique to an objective assessment is the *stability* of the assessment, because it does not rely on any external factors. If the data and quality standard are not changed, neither will the result of the assessment. Objective assessments can be computed ahead of time and stored along-side the data, or computed and cached by the user. Assessments will only need to be updated when the data or standard change.

A **Predictive** assessment requires an encoding of the Quality Fragment. We propose two further considerations for predictive assessments:

- A mechanism for establishing and encoding the prior knowledge that will inform the assessment, for example the correlation between evidence codes and their impact of the likely quality of a gene annotation.
- A mechanism to incorporate uncertainty into the assessment.

A consideration that needs to be made is *when* to establish the prior knowledge. We identify three possible strategies:

1. Compute the prior knowledge once. For example calculating the GAQ's ECRs once. With this strategy we are better able to compare results over time, but newer results may not continue to reflect the current state of the data.
2. Compute the prior knowledge initially and then update periodically based upon new information. This is the strategy taken to update the ECRs for the GAQ metric.
3. Compute the prior knowledge at query time. An up-to-date reference set of data with known quality could be used as a ground truth to compute the parameters of prior knowledge. This would however increase the computational requirements of the assessment.

Finally a **Subjective** assessment requires an encoding of the Quality View, and a mechanism for eliciting and encoding the users subjective requirements for quality. Missier achieves the elicitation through a software tool, the Qurator Workbench. The encoding is achieved through the use of the XML policy language. This means that the user’s requirements are captured ahead of time, and reused at query time. An alternative approach would be to provide a parameterized query, that can be populated at query time by the user. This means that the user’s requirements would be up-to-date, but introduces the need for a mechanism of user interaction.

As a further component of our OPS IQ approach and to demonstrate the classification Table 2.1 details an analysis of twelve information quality studies [JQJ98; Wan98; LSKW02; SVM⁺04; Nau02; BS06; PLW02; EM02; MH05; KFW08; GA07; AT99]. From these twelve IQ studies we have elicited a set of 26 common³ information quality dimensions. For each of the dimensions we have established which OPS classification they fall within.

The goals of our classification framework are similar to those presented by a previous IQ framework discussed in Stvilia et al. [SGTS07]. Like Stvilia we have developed a framework for abstract IQ concepts to guide the development of context specific IQ solutions. Stvilia like many current approaches takes a “top down” approach. Such approaches use a standard set of pre-determined dimensions as a starting point. Specific IQ metrics are then derived from these dimensions for the data in question. Stvilia develops solutions from a set of 41 “generic IQ metrics”. Whilst there has been some success in developing metrics using a top-down approach [Wu13], we have observed that many of the metrics employed by scientists emerge bottom up from domain expert knowledge [BMW⁺08] [SAD12] [WET12] [MH05] or through empirical study of the data [PPH12] [GCdAS11] in response to a particular IQ demand. In contrast to a top-down approach, our aim is to take these existing mechanisms for quality assessment and through our classification, establish processes and techniques suitable for realising these in the Web of Data. Moreover, having a consistent classification enables us to be systematic in our guidance of subsequent development.

An advantage of top-down approaches such as Stvilia possess over our “bottom-up” approach is the ability to guide the creation of metrics in a domain where

³[NR00] was consulted as a guide to synonymous concepts in information quality.

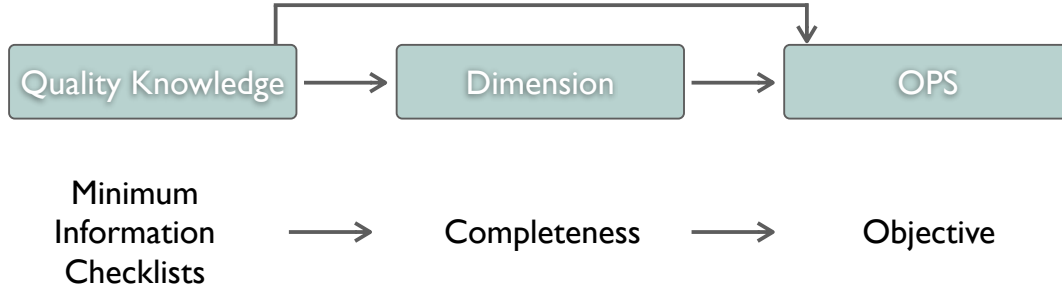


Figure 2.11: Using the OPS IQ classification.

none are currently known. However by pre-determining the abstract set of IQ metrics the authors have made a series of assumptions that restrict the scope of their IQ assessment solutions at the outset.

2.4.1 Using the Classification

The work that follows in this thesis serves to evaluate our classification and proposed recommendations. To do this, we have used the OPS classification in two ways:

1. To inform the development of an IQ solution based upon an Minimum Information Checklists, an existing example of objective Quality Knowledge.
2. To inform a general approach to modelling predictive Quality Knowledge in the Web of Data.

Figure 2.11 illustrates the process of using the OPS classification to inform an IQ solution. Consider Minimum Information Checklists. MICs are a *completeness* assessment that require the data and the MIC (a Quality Standard). MIC assessment is therefore classified as an objective assessment under the OPS classification. With this classification we consult the recommendations to inform our objective IQ assessment solution.

The recommendations can also be consulted to develop more general IQ assessment infrastructure. The recommendations for a predictive assessment for example inform our general approach to modelling Quality Fragments.

Our analysis has highlighted the prevalent role that provenance information plays in the assessment of predictive Quality Knowledge. Later in this thesis, we

investigate the ability to automatically build Quality Fragments using provenance information. In the next section, we provide a more detailed introduction to provenance of electronic information, as well as its representation and usage in the Web of Data.

Quality Dimension	OPS	IQ Studies
Completeness	Objective	[JQJ98; Wan98; LSKW02; SVM ⁺ 04; Nau02; BS06; PLW02; EM02; MH05]
Accuracy	Objective	[Wan98; AT99; EM02; SVM ⁺ 04; JQJ98; Nau02; BS06; KFW08; GA07]
Timeliness	Subjective	[Wan98; LSKW02; EM02; JQJ98; Nau02; BS06; PLW02; MH05]
Consistency	Objective	[BS06; PLW02; LSKW02; EM02; SVM ⁺ 04; Wan98; Nau02; GA07]
Accessibility	Subjective	[Wan98; LSKW02; EM02; JQJ98; BS06; PLW02; Nau02]
Reputation	Predictive	[Wan98; LSKW02; Nau02; BS06; PLW02; GA07; JQJ98]
Objectivity	Predictive	[Wan98; LSKW02; AT99; Nau02; PLW02; KFW08; GA07]
Conciseness	Subjective	[LSKW02; EM02; JQJ98; Wan98; PLW02; Nau02]
Relevance	Subjective	[Wan98; LSKW02; Nau02; PLW02; KFW08; BS06]
Understandability	Subjective	[LSKW02; Nau02; PLW02; Wan98; EM02; BS06]
Believability	Predictive	[Wan98; LSKW02; Nau02; PLW02; GA07]
Interpretability	Subjective	[Wan98; LSKW02; JQJ98; Nau02; PLW02]
Currency	Objective	[AT99; EM02; SVM ⁺ 04; BS06; GA07]
Security	Predictive	[Wan98; LSKW02; Nau02; PLW02]
Amount of Data	Subjective	[Wan98; Nau02; PLW02; GA07]
Correctness	Objective	[JQJ98; MH05; BS06; KFW08]
Value-Add	Subjective	[Wan98; Nau02; PLW02]
Stability	Objective	[KFW08; MH05; BS06]
Applicability	Subjective	[EM02; LSKW02]
Authority	Predictive	[GA07; AT99]
Freedom from Errors	Objective	[LSKW02; PLW02]
Recommendation	Predictive	[GA07; KFW08]
Trustworthiness	Predictive	[KFW08; SVM ⁺ 04]
Usefulness	Subjective	[JQJ98; MH05]
Cost	Subjective	[BS06; Nau02]
Usability	Subjective	[PLW02; LSKW02]

Table 2.1: Information Quality Dimensions Classified using OPS IQ

2.5 The Role of Provenance

2.5.1 What is Provenance?

In the fields of Computer and Information Science *provenance* is metadata that describes the history and production of a piece of data or information. The production and maintenance of provenance metadata has been identified as a beneficial activity to support the critical assessment of data variety of domains. In the databases domain, Buneman et al. [BKWC01] describe provenance as:

“the description of the origins of data, and the process by which that data arrived in the database”

In eScience provenance is used to capture the orchestration of scientific workflows [ZWG⁺04] [MGBM07], and is described by Simmhan et al. [SPG05a] as:

“information that helps determine the derivation history of a data product, starting from its original sources”.

More recently, provenance has been used to address the challenge of capturing the origins and production of information on the Web [Mor10] [MCF⁺11] [GG11b]. This challenge is characterized by the fact that, in contrast to previous closed systems such as databases, the Web is a distributed and open system [GGCM12].

There is an overhead in the capturing and storing of this provenance information [SPG05b]. Provenance metadata is therefore captured with the view that it can be used to subsequently interrogate some aspect of the data. Goble [Gob02] identifies a number of such uses when considering provenance information for scientific data, including:

- **Quality Assessment** - To estimate data quality based on the sources and processes that lead to the data.
- **Credit and Citation** - To determine who should be credited for data, based upon other resources that informed its production.
- **Justification and Auditing** - Where provenance can be used as an historical record of the source and method by which data was produced.

- **Repeatability** - To repeat and validate a process or experiment.

The need for provenance to understand and assess the quality of data on the Web is recognized [MPdS04] [Gol05] [CBHS05] [Biz07] [GA07] [HZ09] [CGVH10]. Early in the development of the Semantic Web, Tim Berners-Lee proposed a user interface feature for Web browsers called the “Oh Yeah?” button [BL97]. The intended purpose of this button is to allow users to instruct the browser to evaluate a piece of information and suggest whether that information should be trusted or not. The implication of this button is that, in order to make an assessment, the browser must have access to that information’s provenance, where it came from, how it was produced.

The provenance literature provides a series of characterizations of types of provenance metadata depending on the intended usage and domain. These characterizations include provenance as a process [Mor10], provenance as annotation [Mor10], Why provenance [BKWC01], Where provenance [BKWC01], and How provenance [CCT09]. For the purposes of this thesis we adopt a “process-documentation” view of provenance [GM09] which the W3C Provenance Incubator Group [GCG⁺10] described as

“a *record* that describes entities and processes involved in producing, delivering and otherwise influencing a resource.”

From this record we are interested in two types of provenance:

- *Lineage* or *Why* provenance, which we view as defining the collection of entities that justify the existence of a resource, and had some influence in its production.
- *How* provenance. Where lineage provenance provides us with the entities that were involved in a resources production, how provenance provides information about how those entities were involved in its production.

For open systems such as the Web, a provenance record is typically represented as a Directed Acyclic Graph (DAG) that describes the derivation of a resource, and the stages of its production [MCF⁺11]. A recent activity by the W3C provenance working group [MG11] has formalised PROV, a series of specifications for modelling, describing, and publishing provenance of information on the Web. The PROV specification overview [GM13] defines provenance as:

“Information about entities, activities, and people involved in producing a piece of data or thing, which can be used to form assessments about its quality, reliability or trustworthiness.”

PROV is an example of a graph-based representation of provenance and uses three core concepts to represent the nodes of the graph: *entities*, *agents*, and *activities*. Entities represent the data or information for which we are describing the production history. Agents are the individuals or organisations that influence those entities during their production. Activities capture the processes that generate and modify these entities. Figure 2.12 illustrates how these core concepts are joined through a series of well-defined relationships to create a graph-based provenance description.

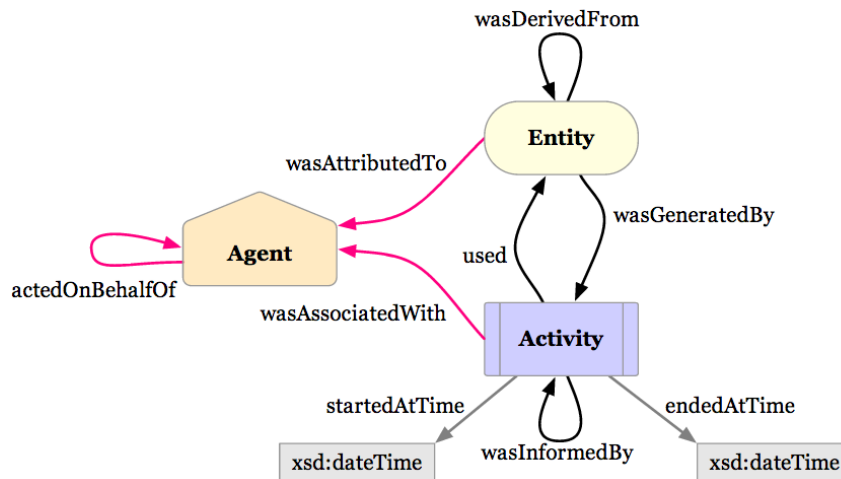


Figure 2.12: Core Elements of the PROV Provenance Model [GM13]

With these concepts and relationships PROV can be used to encode information about the production of data such as Figure 2.13 which describes that the GO annotation *P06727* is an *entity* and *was Generated By* the process *Electronic_Annotation_Process_1*, which is a type of *activity*. It is this structured provenance information that can be exploited for quality and trustworthiness assessments.

For the recording and exchange of provenance information, the PROV specifications provide the PROV Ontology (PROV-O) [LSM⁺13] as the primary implementation of the PROV model. PROV-O is implemented using OWL2/RDF,

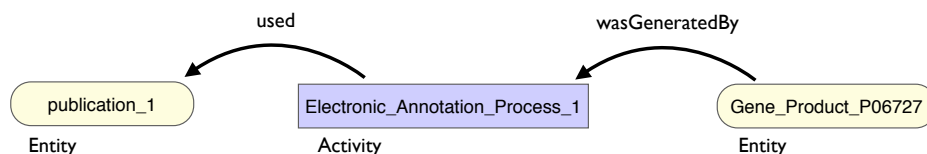


Figure 2.13: Example PROV Provenance Graph for a GO Annotation.

a Semantic Web standard, allowing PROV metadata to be serialised and published in the Web of Data. There has been a significant amount of research into provenance representation on the Web and as a result, there have been a number of models and vocabularies proposed prior to the development of PROV specification. The PROV model is therefore intended to be a high-level interchange language that existing models of provenance can map to, and future models can use as a foundation.

An important question to address given the use of provenance metadata in the assessment of quality is: how do we evaluate the quality of the provenance metadata itself? This has presented itself as an interesting topic of study of its own [CP12] [GMM06] [GM09], and one that we consider out of scope for this thesis. To make the problem tractable we make the assumption that we have ‘good’ provenance i.e. the provenance metadata we have available for a piece of information accurately reflects its true provenance. We envision that similar techniques that are used to address the issues of data quality, may also be applied to provenance quality. Indeed, we can view provenance metadata as a special class of data (the PROV model caters for this through the use of “Provenance Bundles” as a container for provenance) and then reason about its quality.

We consider however the issue of incomplete or missing provenance as related to the inconsistent availability of metadata in the Web of Data, and therefore within the scope of our study.

2.5.2 State of Provenance in the Web of Data

There are a number of vocabularies that have been proposed to represent provenance metadata in the Semantic Web. These vocabularies are being used to varying degrees by many of the datasets available on the Web of Data. Many of the vocabularies complement each other in the scope of their potential usage, as such datasets do not necessarily use just one provenance vocabulary. The Bio2RDF

dataset for example makes use of three vocabularies, VoID [ACHZ09], PROV, and Dublin Core [WKLW98] to describe its data [CCTAD13]. Whilst PROV has only recently been established as a recommendation by the W3C (April 2013), we can look at the usage of existing provenance vocabularies to gauge the usage of provenance metadata.

To understand the current state of provenance usage, and whether there is an increasing usage in Semantic Web datasets, we have performed an update of an analysis originally conducted by Hartig [Har09a]. The original analysis was performed in February 2009 by searching the Sindice⁴ Web data index for occurrences properties from a number of provenance vocabularies. The result of each search is the number of documents in the Sindice Web data index in which the term occurs at least once. Sindice is not an exhaustive index of all of the data available on the Web of Data, but none-the-less provides a useful insight into the trends of Semantic Web data.

We have chosen to replicate this study to discover the relative change since 2009. Table 2.2 lists each of the provenance vocabularies that were considered in the original analysis in 2009. Table 2.3 presents the results of our updated

Prefix	Name	Namespace URI
dc-terms	Dublin Core Terms	http://purl.org/dc/terms/
dc11	Dublin Core Elements	http://purl.org/dc/elements/1.1/
foaf	Friend of a Friend	http://xmlns.com/foaf/0.1/
sioc	Semantically Linked Online Communities	http://rdfs.org/sioc/ns
swp	Semantic Web Publishing Vocabulary	http://www.w3.org/2004/03/trix/swp-2/
wot	Web of Trust	http://xmlns.com/wot/0.1/
cs	Change Set Vocabulary	http://purl.org/vocab/changeset/schema#
iw- Prov	Proof Markup Language	http://inferenceweb.stanford.edu/2006/06/pml-provenance.owl#

Table 2.2: Provenance Vocabularies

analysis performed in February 2013. Each entry in the table gives the number of occurrences from the original 2009 study and the number of occurrences in our 2013 study. To account for the general increase in volume of the data that Sindice indexes we have also measured the relative change in usage. This is done by calculating the coverage of each of the terms as the number of documents the

⁴<http://www.sindice.com>

term appears in relative to the total number of documents indexed by Sindice at the time. In February 2009 Sindice indexed ~ 48.99 million documents [sin09]. In February 2013 Sindice indexed ~ 708.18 million documents.

From the results in the table we can see that the usage of provenance metadata has increased in the Web of Data in two ways.

1. In literal number of triples the usage of some terms has increased significantly since 2009. For example there are approximately an additional 9 million occurrences of the term `dc:creator` from the Dublin Core vocabulary which is used to denote authorship.
2. Provenance metadata has also increased relative to coverage with some core provenance elements seeing whole percentage point increases in their coverage. For example the term `dcterms:modified`.

The increase in provenance usage is not equally distributed across each of the vocabularies considered. With some provenance vocabularies that have not achieved any adoption. However, in general we can see that the terms related to creation and attribution have seen a significant increase in usage, which is promising for potential IQ assessments. With the recent standardization of the PROV suite of specifications by the W3C we can expect this trend to continue. Indeed DBpedia has recently adopted some of the PROV terms to describe provenance for its data.

In the next section we review some existing approaches to IQ assessment in the Web and Web of Data and look at how those solutions have addressed the required infrastructure components.

Property	Occur-	Change		Cover-		Change	
	rences Feb 2009	Feb 2013	#	%	age 2009	2013	percentage points
dcterms:creator	134	16,536,628	+16,536,494	12340667%	0.00027%	2.33507%	2.33480%
dc11:creator	24,150	9,422,998	+9,398,848	38919%	0.04930%	1.32718%	1.27789%
dcterms:contributor	11	3,534,904	+3,534,893	32135391%	0.00002%	0.49915%	0.49913%
dc11:contributor	465	4,062,784	+4,062,319	873617%	0.00095%	0.57363%	0.57268%
dcterms:source	1	886,177	+886,176	88617600%	0.00000%	0.12513%	0.12513%
dc11:source	3630	3,526,919	+3,523,289	97060%	0.00741%	0.49751%	0.49010%
dcterms:created	73010	10,151,557	+10,078,547	13804%	0.14903%	1.42316%	1.27413%
dc11:created	9710	1,386,861	+1,377,151	14183%	0.01982%	0.19446%	0.17464%
dcterms:modified	2320	8,591,147	+8,588,827	370208%	0.00474%	1.21280%	1.20807%
dc11:modified	9700	519,242	+509,542	5253%	0.01980%	0.07195%	0.05215%
dcterms:publisher	87	1,327,254	+1,327,167	1525479%	0.00018%	0.18741%	0.18723 %
dc11:publisher	808	12,245,351	+12,244,543	1515414%	0.00165%	1.72902%	1.72737 %
dcterms:provenance	7	131,603	+131,596	1879943%	0.00001%	0.01858%	0.01857%
foaf:made	5420	54,611	+49,191	908%	0.01106%	0.00695%	-0.00412%
foaf:maker	29370	3,326,047	+3,296,677	11225%	0.05995%	0.46551%	0.40556%
sioc:creator of	1370	24,649	+23,279	1699%	0.00280%	0.00329%	0.00049%
sioc:has creator	4520	825,738	+821,218	18169%	0.00923%	0.11596%	0.10674%
sioc:modifier of	3	22	+19	633%	0.00001%	0.00000%	0.00000%
sioc:has modifier	4	36	+32	800%	0.00001%	0.00000%	0.00000%
sioc:owner of	3020	1,770	-1,250	-41%	0.00616%	0.00018%	-0.00599%
sioc:has owner	553	49,103	+48,550	8779%	0.00113%	0.00686%	0.00573%
sioc:earlier version	0	96	+96	-	0.00000%	0.00001%	0.00001%
sioc:later version	0	23	+23	-	0.00000%	0.00000%	0.00000%
sioc:next version	3	26	+23	767%	0.00001%	0.00000%	0.00000%
sioc:previous version	3	200	+197	6567%	0.00001%	0.00003%	0.00002%
swp:assertedBy	0	3	+3	-	0.00000%	0.00000%	0.00000%
swp:authority	0	8	+8	-	0.00000%	0.00000%	0.00000%
swp:quotedBy	0	3	+3	-	0.00000%	0.00000%	0.00000%
swp:validUntil	0	3	+3	-	0.00000%	0.00000%	0.00000%
wot:assurance	135	305	+170	126%	0.00028%	0.00002%	-0.00025%
wot:fingerprint	52	101	+49	94%	0.00011%	0.00001%	-0.00010%
wot:hasKey	23	57	+34	148%	0.00005%	0.00000%	-0.00004%
wot:hex id	48	88	+40	83%	0.00010%	0.00001%	-0.00009%
wot:identity	36	55	+19	53%	0.00007%	0.00000%	-0.00007%
wot:length	43	82	+39	91%	0.00009%	0.00001%	-0.00008%
wot:pubkeyAddress	54	96	+42	78%	0.00011%	0.00001%	-0.00010%
wot:sigdate	8	10	+2	25%	0.00002%	0.00000%	-0.00002%
wot:signed	3	4	+1	33%	0.00001%	0.00000%	-0.00001%
wot:signer	2	4	+2	100%	0.00000%	0.00000%	0.00000%
iwProv:hasMember	1	0	-1	-100%	0.00000%	0.00000%	0.00000%
iwProv:isMemberOf	1	0	-1	-100%	0.00000%	0.00000%	0.00000%
iwProv:hasPublisher	1	0	-1	-100%	0.00000%	0.00000%	0.00000%
iw-	1	0	-1	-100%	0.00000%	0.00000%	0.00000%
Prov:hasPublicationDateTime							
iw-	1	0	-1	-100%	0.00000%	0.00000%	0.00000%
Prov:hasUsageDateTime							
iwProv:hasSource	1	0	-1	-100%	0.00000%	0.00000%	0.00000%
iw-	1	0	-1	-100%	0.00000%	0.00000%	0.00000%
Prov:hasInferenceEngineRule							
iw-	1	0	-1	-100%	0.00000%	0.00000%	0.00000%
Prov:usesInferenceEngine							
ouzo:belongsTo	2	0	-2	-100%	0.00000%	0.00000%	0.00000%
ouzo:dataDerivedFrom	2	0	-2	-100%	0.00000%	0.00000%	0.00000%
ouzo:launchedBy	2	0	-2	-100%	0.00000%	0.00000%	0.00000%
ouzo:processInput	2	0	-2	-100%	0.00000%	0.00000%	0.00000%
ouzo:runsWorkflow	2	0	-2	-100%	0.00000%	0.00000%	0.00000%
cs:createdDate	3	6,231	+6,228	207600%	0.00001%	0.00088%	0.00087%
cs:creatorName	3	5	+2	67%	0.00001%	0.00000%	-0.00001%

Table 2.3: Provenance Vocabulary Usage in the Web of Data

2.6 Related Work in Information Quality Assessment

In the final part of this chapter we review related work in the area of IQ assessment solutions. In the preceding sections of this chapter we have identified the infrastructure components required for Web-based IQ assessment: Quality Evidence Description, Quality Evidence Alignment, Quality Knowledge Encoding, and Orchestration. We use these components as a framework to compare existing IQ solutions. Table 2.4 summarises “complete” IQ assessment solutions discussed in this section. These complete solutions address each of the components we have outlined above. Where the authors have not described clearly the mechanism used for part of their approach we assume that it was achieved using a bespoke implementation specifically for that study. We do not however limit our discussion to complete solutions and also discuss solutions to part, or parts, of the IQ assessment process.

Provider-Centric IQ Assessment

Mihaila et al. [MRV00] present early work in addressing IQ based information source selection on the Web. Preceding many of the current Semantic Web standards, the authors propose an XML based language called *source content quality data* descriptions (scqd) for describing IQ metadata about a data source, such as its completeness, or when it was last updated. These descriptions are published by data providers along side their data resource and each piece of IQ metadata acts as Quality Evidence. Quality Knowledge is then encoded in queries written in a bespoke query language. These queries use the evidence to rank information sources and select the most appropriate resource given the requirements expressed in the query. Naumann et al. [NR00] [Nau02] also present a similar approach to query planning using IQ metadata. IQ meta data is provided as part of the data model in a distributed relational information system. The authors then propose a framework for query-planning that dynamically evaluates the best information source to satisfy parts of the query.

There are two drawbacks to these *provider-centric* solutions:

- There is a reliance on the data provider to provide detailed quality metadata. This means that only resources that have adopted the approach can be assessed.

Framework	Quality Evidence Encoding	Quality Knowledge Encoding	Alignment	Orchestration	Goal	Provenance Use	OPS
Mihaila et al. [MRV00]	sqcd descriptions	sqcd queries	Manual	Bespoke Implementation	Source Selection	No	Sub-jective
WIQA [Biz07]	Semantic Web Publishing Vocabulary	WIQA-PL	Human	WIQA-Framework	Filtering	Yes	Sub-jective
Quarator [Mis08]	Web Service Outputs	Semantic Web Services, Quality Views	Annotation Functions	Quality Workflows	Classification	No	Sub-jective
Trellis [GR02]	Dublin Core	Manual	Human	TRELLIS	Credibility, Reliability	Yes	Sub-jective
Sieve [MMB12]	LDIF Provenance	SQASL, Scoring Functions	R2R mapping	LDIF	Data Fusion	Yes	Objective
Furber et al. [FH10]	DQM ontology	SPARQL Queries	SPARQL	SPIN Engine	Data Cleaning	No	Objective
Hartig [HZ09]	Provenance Vocab	Bespoke Implementation	Assumed	Bespoke Implementation	Scoring	Yes	Pre-dictive
IWTrust [ZDSM05]	Proof Markup Language	PML Queries	Automatic	IWTrust Framework	Explanation	Yes	Pre-dictive
Ceolin [CGVH10]	FOAF, Dublin Core, annotationTrust	Prolog	Assumed	Prolog Module	Likelihood Assessment	Yes	Pre-dictive
tSPARQL [Har09b]	Trust-weighted RDF graph	tSPARQL Queries	Assumed	tSPARQL Engine	Ranking	No	Pre-dictive
Dai et al. [DLBK08]	Relational Schema	Bespoke Implementation	Assumed	Bespoke Implementation	Scoring	Yes	Pre-dictive

Table 2.4: IQ Assessment Frameworks

- There are also issues related to whether the data provider is correctly incentivised to provide the correct IQ information [Mis08].

User-centric IQ Assessment

Gil et al. [GR02] present a more user-centred approach to addressing the quality of information on the Web. The authors describe the TRELLIS system, an information browser which is used to support data analysts in making decisions about what information to accept. This approach is realised in two stages. This first stage is a process of Quality Evidence Alignment. The browser allows analysts to mark-up Web pages as structured information in RDF, attributing statements in Web pages to sources, and associating those sources with a reliability and credibility scoring. In the second stage subsequent users can then apply a filtering policy that uses these scorings to determine which information to use for their analyses.

Bizer [Biz07] proposes a similar browser-based framework and describes the Web Information Quality Assessment (WIQA) framework for filtering Web content based upon user requirements.

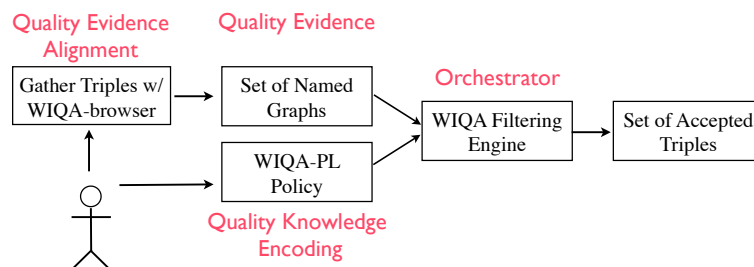


Figure 2.14: The WIQA Filtering Processes.

Figure 2.14 outlines the processes involved in the WIQA framework. Quality Evidence is gathered through the WIQA Browser. To mark up Quality Evidence, Bizer defines an RDFS vocabulary, the Semantic Web Publishing Vocabulary (SWPV). SWPV provides a mechanism for describing provenance information such as the source of information using `swp:sourceURL`, and who made a statement via `swp:assertedBy`. Like TRELLIS, the alignment process is therefore performed by the user as they are browsing Web pages.

For the assessment phase Bizer improves on TRELLIS, supporting more than

one assessment policy by allowing users to define their own. To encode these policies Bizer describes the WIQA-PL policy language, an extension of the authors' previously proposed TriQL.P policy language [BCGM05]. WIQA-PL policies capture a user's subjective preferences about IQ as a pattern.

```

1 NAME "Information from highly rated analysts"
2 DESCRIPTION "Accept only information that has been asserted by analysts who
  achieved a StarMine score above 89."
3 PATTERN {
4     GRAPH fd:GraphFromAggregator {
5         ?GRAPH swp:assertedBy ?warrant .
6         ?warrant swp:authority ?authority .}
7     GRAPH fd:BackgroundInformation {
8         ?authority rdf:type fin:Analyst .
9         ?authority fin:benchmark ?benchmark .
10        FILTER (?benchmark > 80) . }
11 }

```

Listing 2.1: Example WIQA Policy from Bizer et al.

WIQA-PL policies can then be executed over the Quality Evidence gathered by the browser to filter information. Listing 2.1 illustrates an example WIQA policy from [BC09] that filters for information asserted by people who are highly rated.

A strength of the WIQA framework lies in how the filtering results are presented to the user. Bizer provides “explanations” - human readable strings that describe why a piece of information appears in a result. Bizer views human readable explanations as key for users to understand and accept IQ assessments, as apposed to numerical values [personal comms w/ Bizer].

Both WIQA and TRELLIS make use of RDF as a mechanism for representing Quality Evidence in a machine readable and structured manner. This allows them to mechanise approaches to assessing IQ. Another feature common to both is that they expect the user to perform the alignment manually by browsing content. In contrast to provider-centric approaches this moves the task away from the provider, but it in turn raises questions about the scalability of the approach. These approaches were however developed prior to much of the recent increase in availability of Web Data, and there is scope to re-apply these techniques to existing RDF in the Web of Linked Data.

IQ assessment for Linked Data

There has been recent interest from the Linked Data community in achieving scaleable approaches to “RDF Validation”. A workshop held by the W3C in September 2013 [Pru13a] sought to establish requirements for RDF validation and evaluate the state of the art for defining constraints over RDF data. Proposals presented ranged from grammar-based [Pru13b] [RLHS13] to query-based [GCL13] [SB13] [JSC13] approaches. A number of common requirements emerged from the workshop including:

1. The need for a declarative definition of the requirements for an RDF graph that can be used for validation.
2. The need to be extensible for specialized use-cases.

The requirements of RDF validation are similar to an objective assessment in our OPS classification, where the declarative definitions are a Quality Standard against which RDF data is to be validated. Amongst the solutions presented there appears to be broad support for the SPARQL Protocol and RDF Query Language (SPARQL) query language for the task of RDF validation.

Furber et al. [FH10] [FH11] present another SPARQL-based approach to validation of RDF data on the Web of Data. The authors use SPARQL to encode a series of general queries to identify common IQ issues for Linked Data, such as missing literals, invalid literal values etc. Specifically the authors use the SPARQL Inferencing Notation (SPIN), which provides a serialisation of SPARQL as RDF that allows the authors to share SPARQL rules as quality components. Alignment is achieved automatically by the SPARQL queries.

The use of SPARQL can be seen as a natural progression from the use of the bespoke WIQA-policy language proposed by Bizer et al. A limitation to using a query language such as SPARQL to describe quality requirements is that the queries are performing both the encoding and alignment tasks and therefore make assumptions about the data’s structure. Bizer controlled both the evidence encoding through SWPV and the knowledge encoding with WIQA-policies. The policies were therefore safe in making assumptions about the structure of the data. However, for general RDF data “in the wild”, making assumptions about a specific representation of the data limits reusability for alternative representations. This suggests that a separation of concerns is required between the encoding and alignment to improve reusability.

An alternative use of SPARQL proposed by Hartig [Har09b] extends the query language to take into account trust annotations on RDF data. Hartig proposes that RDF triples and graphs can be annotated with subjective trustworthiness scores on a scale $[-1...1]$, and proposes tSPARQL as an extension to SPARQL that takes these annotations into account. In contrast to other approaches using SPARQL, tSPARQL does not describe filtering policies, but instead allows users to rank query solutions based upon their likely trustworthiness. Hartig does not prescribe a specific mechanism for calculating trust annotations, and instead suggests that existing approaches such as TRELLIS can be used. The approach is limited however to mechanisms that provide a trust score on a scale $[-1...1]$.

A number of Linked Data specific IQ assessment tools focus on the task of data integration. These tools tend to adopt Redman’s technical “free from defects” definition of IQ and support the user in identifying common, objective quality issues such as duplicate data, inconsistencies and missing values. LODRefine [LOD13] is one such integration tool that builds upon the popular Google refine data management software, a spreadsheet-like tool where users can manually interrogate and integrate large datasets. LODRefine does not provide an automated IQ assessment and instead re-applies techniques present in the Google Refine application to support the user in manually applying their own Quality Knowledge to improve and integrate Linked Data. LODRefine is not the only tool to support the manual application of Quality Knowledge for Web Data integration. TripleCheckMate [KZAL13] is a tool for crowdsourcing Linked Data quality assessment. The tool presents crowd workers with a part of a dataset, and asks them to verify the data along a number of IQ dimensions, such as correctness.

The Linked Data Integration Framework (LDIF) [SMI⁺11] is an integration tool more in line with our objectives in two respects 1) it automates the integration process and 2) it uses quality metrics as part of its integration. Sieve [MMB12] is the component of the LDIF responsible for the quality assessment phase of the integration. Sieve uses a bespoke XML policy language to encode quality filters that can be reused in integration activities. In order to discover evidence metadata the Sieve assessment module requires the data to be annotated with the *LDIF provenance vocabulary*. This annotation is generated automatically by the LDIF when importing data by using the RDF schema mapping tool R2R [BS10].

This approach continues to demonstrate the value in controlling the vocabulary for evidence description, allowing the XML policies to make assumptions about the representation. The approach improves on WIQA’s use of a custom vocabulary by automatically annotating the data upon import using R2R. This however introduces the potential for the mapping approach itself to affect the quality of the data.

Away from integration Hogan et al. [HUH⁺12] present a large-scale standards-based assessment of Linked Data. The authors have evaluated approximately 4 million RDF documents against the recommendations described by the Linked Data principles. The authors demonstrate that by automating such an objective assessment they can provide an immediate and broadly relevant insight into the conformance of datasets, and highlight where lack of conformance is likely to impact data consumers trying to use that data.

Provenance-based IQ Assessment

The predictive approach to evaluating quality has been the subject of modelling [Mar94] and computation [Gol06] in the computational trust literature. More commonly investigated is *agent-based* trust, where trust assessments are considered between two agents [Gol05] [Mar94] [ZL04]. An alternative *data-centric* or *content-based* notion of trust is closely related to our investigation [GA07] [Har10] [MZdS⁺06]. In this data-centric approach, features of data such as its source, author, and its age are used to make judgements about its likely quality and trustworthiness. The user-centric frameworks discussed above demonstrated a data-centric approach to evaluating IQ. Gil et al. [GA07] were amongst the first to establish the notion of content-trust, extending their TRELLIS approach to incorporate additional metadata to be used to make quality and trustworthiness predictions. Surveys of the trust literature [Gol06] [AG07] [OH08] have shown that provenance information plays an important role in the evaluation of data-centric trust.

Provenance-based quality assessment is often based upon the assumption that entities, agents and activities that had some influence on the production of data, will have affected its likely quality. As a result, if we have some mechanism for measuring the quality of those influencers, then we can use it to inform us of the likely quality of the data in question.

When making this kind of assessment in practice, provenance information is

often used in combination with existing mechanisms for assessment to make IQ based decisions. To distinguish between these we propose an alternative and complimentary view of Quality Knowledge for provenance-based approaches that make use of two types *intrinsic* and *provenance-based*:

intrinsic - quality that is related directly to the entities, agents and activities, such as reputation, completeness, accuracy etc.

provenance-based - quality that takes into account how the intrinsic quality of other elements relates to the resource in question, by using provenance information.

To exemplify this consider the gene annotation provenance in Figure 2.15. By highlighting the resources considered we can see that the ECR is intrinsic to the electronic annotation process. In contrast the GAQ score for the gene product *Gene_Product_P06727* takes into consideration the intrinsic ECR score for the process due to the provenance information describing it as generated by that process.

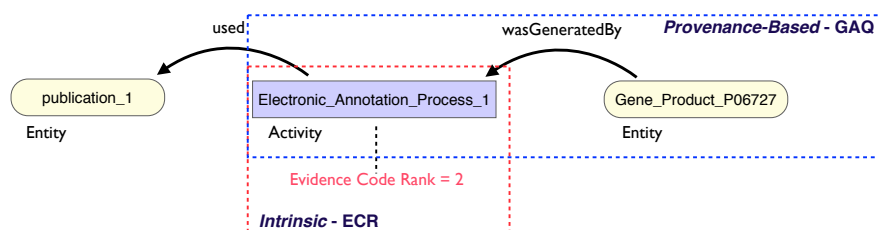


Figure 2.15: Intrinsic vs. Provenance-based Quality Knowledge in the GAQ score.

In the literature Zaihrayeu et al. [ZDSM05] provide an early example of the use of provenance information to evaluate the trustworthiness of information on the Web. The authors present IWTrust which enables the computation of trust values for answers from a query answering engine. The approach uses the Proof Markup Language (PML), a provenance vocabulary where the nodes in the graph are inference steps, and represent intermediate justifications and information sources that lead to a conclusion. Trust values are associated with the intermediate justifications and information sources and act as the intrinsic Quality Knowledge. The provenance-based knowledge is then a mechanism for combining these trust values, taking in to account provenance-based information

such as the length of the path from the conclusion to the intermediate representation.

Dai et al [DLBK08] propose a more general provenance-based approach to assess data in a distributed data system that takes into account the trustworthiness of data sources and intermediate agents. The authors propose a series of measures; two intrinsic: data similarity and data conflicts, and two provenance-based: path similarity and data deduction (taking into account the trustworthiness of intermediate agents). These measures are combined to provide an overall trust value for some data, based upon its provenance. The data deduction measure also takes into account an element of *how* provenance in its assessment. The authors define two types of interaction that an intermediate agent can have with data, PASS or INFER. PASS means that an agent simply passed the data on, INFER means that the agent inferred new knowledge from some input data. These actions impact trust worthiness score differently.

Golbeck et al. [GH06] have demonstrated the use of provenance and intrinsic knowledge about agent trustworthiness in social networks. The authors apply their previously proposed TidalTrust [Gol05] trust metric that uses transitive relations in a social network between two agents to estimate likely trustworthiness. In the FilmTrust system described, the authors evaluate the quality of film reviews by combining provenance-based information about who a review is attributed to, with the intrinsic trustworthiness assessment.

Groth et al. [GMM⁺09] propose the use of provenance to support the prediction of the likely success of newly proposed electronic contracts. This prediction is a combination of two assessments, (1) the success of other previous contract executions (2) similarity of the new contract compared with those previous contracts. In an example use-case of contracts for engine manufacture, the authors use a provenance graph to represent the activities and assets used during a contract execution. A number of intrinsic elements of quality are defined as quality classes for features of a contract such as, time to repair [*short, long*], penalty payment [*high, low*], part supplier [*permitted, prohibited*]. Contract success is assessed using provenance-based quality. The assessment is a scoring based upon the co-occurrence of these intrinsic features in the contract execution provenance graph.

In the Web of Data, Hartig and Zhao [HZ09] demonstrate an approach to

assessing the quality using provenance described using Hartig’s previously proposed Provenance Vocabulary [HZ10]. The the authors focus on the calculation of a score for timeliness and propose three components required for quality assessment:

1. A Provenance Graph for a data item.
2. The annotation of that provenance graph with *impact values*.
3. A mechanism for computing an IQ score given the provenance graph and impact values.

Impact values are defined by Hartig and Zhao as any metadata that might impact a resulting IQ assessment, such as a data creator’s credibility, or a the creation time for their timeliness assessment. 1. is therefore the intrinsic knowledge for elements of the provenance graph, and 2. is the provenance-based knowledge that combines these with provenance information to make an IQ assessment.

Provenance information has also been used in the Web of Data to improve the quality and scalability of reasoning. Bonatti et al. [BHPS11] describe an approach to combine a series of intrinsic quality measures with provenance information to mitigate noise and inconsistencies during reasoning. The authors propose three intrinsic measures for vocabularies and data sources:

- blacklist - Boolean annotations about untrusted sources.
- authority - Boolean annotations to describe the authoritative source of a vocabulary term.
- ranking - A numerical ranking for RDF documents based on the PageRank algorithm.

These measures are then combined with provenance information during reasoning where the authors have shown that they can successfully identify and repair inconsistencies.

Uncertainty based IQ assessment.

We have proposed in this chapter that predictive Quality Knowledge often requires the consideration of uncertainty. Research has previously explored the use of uncertainty modelling in IQ assessment on the Web. A number of studies

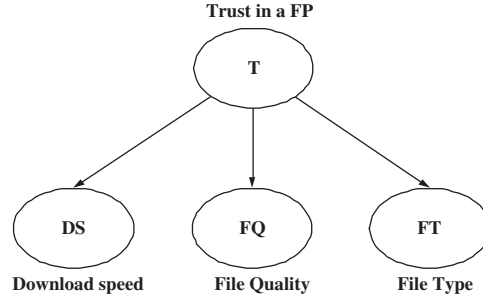


Figure 2.16: Example Bayesian Network Metric from Wang et al. [WV05]

have exploited the uncertainty modelling potential of Bayesian Networks to represent IQ. We introduce Bayesian Networks in detail in Chapter 4, so limit our description of them here to *a mechanism for representing and reasoning about probabilistic relationships between variables* in a given domain.

Bayesian Networks have been used to encode Quality Knowledge in a variety of domains including peer-to-peer (P2P) networks [WV05] [VA07], e-commerce [WCG⁺06], Web-based recommender systems [YGL11] [AEK00], and Web portals [CCSP07]. Figure 2.16 for example shows an example Bayesian Network from [WV05] to predict the trustworthiness (T) of a file provider in a P2P network using three features as Quality Evidence, download speed (DS), file quality (FQ), and file type (FT).

Several of these approaches propose specific Bayesian Networks to tackle an IQ assessment problem such as the network from Wang in Figure 2.16 for nodes in a P2P Network. Caro et al. [CCSP07] also propose a specific Bayesian Network structure by combining dimensions from the TDQM IQ methodology to assess the quality of a Web portal.

Other approaches recognize the modular nature of Bayesian Networks, and the ability to add and remove variables and structure to tailor a network to a specific situation. [WCG⁺06] and [YGL11] propose techniques to procedurally build Bayesian Networks for a specific situation, or user requirements.

Previous IQ work applying Bayesian Networks highlights two strengths that relate to our desired reusability characteristics:

1. Due to their grounding in Bayesian probability, the independence assumptions inherent in Bayesian Networks mean that they can be easily extended and combined to create complex assessments. This makes them a good fit

to the multi-dimensional nature of IQ assessment, and to our reusability requirement for modularity.

2. Bayesian Networks also demonstrate the ability to work well with incomplete evidence. This is achieved by encapsulating the prior likelihood of each piece of evidence in their modelling. This means that in the face of missing evidence, the most likely value can be inferred and used in its absence.

Bayesian Networks have also been used by Zeng et al [ZAFM06] to combine provenance-based Quality Knowledge with uncertainty modelling. The authors have successfully applied a Bayesian Network to the task of computing the likely quality of Wikipedia articles, by examining the provenance of the articles revision history. Three pieces of evidence are used

- Provenance data describing the lineage provenance of each revision of a Wikipedia article.
- The author of each revision.
- A quantitative measure of the contribution that author made to the revision.

The intrinsic quality used by Zeng is a measure of trustworthiness in the author on a scale of [0...1] based upon their status: Administrator, Registered, Unregistered or Blocked. Provenance-based knowledge is encoded in the Bayesian Network and used to predict the likely quality of the latest revision of an article. This a combination of the lineage information about the revisions and the author trustworthiness, weighted by their contribution.

The literature demonstrates that Bayesian Networks are an intuitive approach to modelling Quality Knowledge. None of the approaches reviewed however, propose a mechanism for sharing the Bayesian Network itself as a mechanism for sharing Quality Knowledge. There are a number of recent developments in the field of Uncertainty Reasoning in the Semantic Web (URSW) that present the opportunity to model Bayesian Networks in the Web of Data, including PR-OWL2 [CLC13], BayesOWL [DPP06], and OntoBayes [YC05].

Beyond Bayesian Networks others have applied evidence-based approaches to evaluating uncertainty in the Semantic Web, such as Dempster-Shafer theory [BG11] or subjective logic [Jøs97]. Ceolin et al. [CVHF10] [CNF12a] [CNF12b] apply subjective logic to the quality assessment Web of Data based annotations.

Subjective logic is a probabilistic logic that allows the representation of *opinions*, that are statements coupled with a degree of belief in that statement. Formally an opinion is a quad $\omega(b, d, u, a)$ and $b, d, u, a \in [0...1]$, where b represents belief, d disbelief and u uncertainty such that $b + d + u = 1$. Each piece of evidence, such as who authored a statement, goes towards increasing or decreasing the degree of belief in the opinion for that statement. Any evidence in support of the statement will increase b and any evidence against will increase d . In the absence of evidence the degree of belief in a statement is captured by the prior belief that the statement is true, captured by a .

In [CVHF10] Ceolin uses subjective logics to combine multiple, often conflicting opinions about a statement in order to determine a trust value that indicates the likelihood of its correctness. Quality Evidence comes in the form of provenance vocabularies such as Dublin Core, the Friend of a Friend Ontology (FOAF) [BM10] and a bespoke annotationTrust vocabulary [Ceol]. The Quality Knowledge Encoding to combine evidence and opinions is encoded in a series of Prolog modules. Ceolin demonstrates that subjective logic, like Bayesian Networks also manage well with partial or missing evidence due to the modelling of a prior belief.

In later work Ceolin extends the use of subjective logics to a provenance-based assessment [CGVH10] [CGvH⁺12]. Ceolin addresses the problem of predicting the quality of tags applied to videos in an online, crowdsourced video tagging game called Waisda[Wai]. Ceolin combines the intrinsic reputation of the users, with provenance based features about a tag, such as the reliability of tags at a given time of day, to evaluate an overall trust rating.

Away from the Web, uncertainty reasoning and provenance have been combined to evaluate IQ in relational database systems. Uncertainty and Lineage Database (ULDB) proposed in [BSHW06] and implemented in the Trio system [Wid04] support the ranking of possible solutions to queries. Tuples in the database can have alternatives where each is declared with a confidence value, which acts as the intrinsic quality. The authors define a method of ranking query solutions by inspecting the tuples in the lineage of a solution, and combining their associated confidence scores.

Our review of related work highlights that there is no single IQ assessment task that is being addressed on the Web and Web of Data, but instead a range

of tasks for which different quality components are required. There are therefore a broad spectrum of approaches currently proposed to tackle IQ assessment. For many tasks such as orchestration or evidence encoding there is no clear emerging standard. For the task of alignment and Quality Knowledge encoding there is though a prevalent and flexible use of SPARQL or SPARQL-based techniques.

In summary we make the following observations from our review that inform our investigations:

- SPARQL is emerging as a clear candidate for the validation of RDF data in the Web of Data, and objective IQ assessments.
- There is a need for a rule-based mechanism to manage uneven representation in the Web of Data, and support alignment.
- There is a prevalent use of provenance information to inform and broaden the scope of predictive IQ assessment.
- Uncertainty modelling can mitigate the inconsistent availability of evidence.
- Bayesian Networks provide are a clear candidate for an intuitive representation of Quality Knowledge.

2.7 Summary and Conclusions

In this chapter we have presented the state of the art in approaches to modelling IQ. We have grounded our work in the IQ Life-cycle, a conceptual model tailored to the scientific domain developed by Missier. We have extended the work of Missier by examining Quality Knowledge and have established three prevalent aspects: *objective*, *predictive*, and *subjective*.

From these aspects we have proposed OPS IQ, a process-centric classification tailored to Quality Knowledge. This classification supports the engineer by proposing a series of considerations for IQ assessment solutions.

Finally we have reviewed existing approaches to IQ assessment in the Web and examined how these solutions have addressed each of the components.

In the rest of this thesis we apply our classification and explore two approaches for modelling and computing IQ. We take Minimum Information Checklists as an example of a Quality Standard and implement an IQ solution for evaluating Linked Data against a MIC. Following this we address the predictive aspects of

Quality Knowledge by developing an approach to modelling Quality Fragments in the Web of Data using Bayesian Networks.

Chapter 3

MIM: a Minimum Information Model

“Not everything that can be measured matters, and not everything that matters can be measured.”

- William Bruce Cameron, Sociologist

3.1 Chapter Introduction

This chapter describes the first of the two studies to be drawn from our OPS IQ classification. We have specifically identified Minimum Information Checklists (MICs) as an example of objective Quality Knowledge suitable for exploitation. Our goal is to develop a solution that allows us encode MICs as reusable quality components in the Web of Data. To do this we have developed the Minimum Information Model (MIM) Vocabulary and Framework.

The MIM framework aims to support three core activities: (1) Supporting authorities, data providers and the community, in publishing well structured MICs describing the minimum set of information required when publishing a particular class of data; (2) Supporting individuals such as data creators, and scientists in publishing Linked Data against a MIC; and (3) Supporting users in assessing existing data ‘in the wild’ against a MIC.

To evaluate our framework, we have performed a case-study to assess a Linked Data extraction of the chemical compound data available in Wikipedia. Scientific Linked Data is commonly extracted from an original source and converted into its

Linked Data representation. A MIC assessment is not only assessing the quality of the original resource, but also the quality of the Linked Data extraction. We therefore investigate the interactions between the IQ Life-cycle and the Linked Data extraction process. We highlight the role that the Quality Knowledge Engineering aspect of the IQ Life-cycle can play in informing and improving the extraction process, and how the extraction process can equally inform engineering.

We begin the chapter by revisiting MICs and discussing the types of Quality Knowledge they commonly express. To inform the design process of the MIM vocabulary we present an analysis of existing MICs in order to determine common features. We then discuss the MIM vocabulary, a meta-modelling vocabulary suitable for encoding checklists in RDF, and aligning existing RDF data with those checklists. We go on to demonstrate the framework with a checklist designed for the publishing of chemical compound Linked Data, using the data extracted from Wikipedia. Finally, in light of a number of emerging approaches to RDF validation we compare our approach with existing work.

Parts of the work presented in this chapter have previously been published in the following:

- Matthew Gamble, Jun Zhao, Graham Klyne, Carole Goble. *MIM: A Minimum Information Model Vocabulary and Framework for Scientific Linked Data*. Proceedings of the Eighth IEEE International Conference on eScience, 2012.
- Jun Zhao, Graham Klyne, Matthew Gamble and Carole Goble. *A Checklist-Based Approach for Quality Assessment of Scientific Information*. 3rd International Workshop on Linked Science 2013 (LISC2013).

3.2 Minimum Information Checklists

Minimum Information Checklists have emerged within the Life Sciences as a means of standardising the reporting of experiments in an effort to increase the quality and reusability of the reported data and meta-data. To do this, MICs define a minimum list of information and attributes that must be included in the submitted data and, in some cases, the format in which the data should be reported. Figure 3.1 shows an extract from the previously discussed MIARE

MIARE - Summary of Required Information

Minimum Information About an RNAi Experiment (MIARE)
(www.miare.org)

Checklist of Required Information*

The purpose of this check-list is to guide and help experimentalists to ensure that the data supporting their results can be made publicly available and that the interpretation of the data can be made by others.

The following check-list of information that SHALL be included in the information has been on the Guideline document at www.miare.org

Checklist

A. Assay description:

A.1. Assay ID
A.2. Assay name
A.3. Assay type (primary/confirmatory/other)
A.4. Target organism (Taxonomy ID)
A.5. Number of distinct genes targeted for knock-down
A.6. Experiment publication (PubMed ID)
A.7. Primary contact information

B. Protocol:

B.1. Experimental design
B.1.1. Experiment title
B.1.2. Biological question description - (including sample description and keywords)
B.2. Assay
B.2.1. Assay protocol and design - (including number and description of replicates (biological/technical))
B.2.2. Pre- and post-treatment (protocol/type/compound)
B.2.3. Bio-material manipulations (including growth conditions/cell culture conditions and if applicable cell separation technique)
B.2.4. Number of cells per well
B.2.5. Compound(s) name (if applicable)

Figure 3.1: The Minimum Information about and RNAi Experiment (MIARE) Checklist (Assay Requirement Set).

checklist for reporting RNAi experiments. The MIARE checklist details 63 reporting requirements in total and is attributed to 61 authors who are members of the RNAi community. Requirements are grouped into conceptual sets, the first set of requirements in the checklist is highlighted in the figure. This requirement set describes the 7 pieces of metadata to describe an Assay¹. Some of the requirements in the checklist guide the author with the type of expected report. For example “A.6 Experiment Publication” specifies that an article ID from the PubMed database is required as the report for this requirement.

The MIARE checklist, like others, is not just available as a PDF but also as a tab-delimited spreadsheet to support the author in composing compliant experiment descriptions. The Biosharing project currently lists 60+ MICs available to guide the reporting of a range of biological experiment types. These checklists currently exist in a number of formats including PDFs, Excel spreadsheets, and XML Schema definitions, but crucially for the Web of Data not RDF.

There has been some previous effort to harmonize and structure the representation of MICs beyond their traditional textual representation. The MIBBI

¹An assay is an experimental procedure for testing the properties of a biochemical substance.

project for example currently hosts a Web based tool MICheckout [KFS⁺10]. This tool allows the user to download existing checklists (such as MIARE) or compile their own custom checklists by selecting from a number of common sets of requirements. These MICs can then be exported as HTML, XML Schema definitions, tab-delimited files, and MediaWiki templates. The RightField tool [WOH⁺11] provides the ability to specify MICs as Excel spreadsheets. A feature unique to the RightField tool is that the checklist creator can restrict elements of a spreadsheet to particular terms defined in a controlled vocabulary. This ensures that experimental reports subsequently created using the spreadsheet are compliant to a particular data format.

In the broader effort to improve the quality and interoperability of Life Sciences data, the ISA (Investigation, Study, Assay) framework and supporting tooling [RSBM⁺10] is gaining significant adoption. The ISA framework relies upon data producers conforming to common metadata categories “Investigation, Study and Assay”, with the goal of moving towards an “ISA commons”. Central to the ISA ecosystem is the ISA-Tab format. ISA-Tab is a hierarchical tab-delimited template that details minimum reporting requirements whilst ensuring data is captured in the ISA format.

Figure 3.2 illustrates the current usage of checklist-based solutions and how we aim to extend it by publishing checklists as RDF. Existing tooling built around checklist-based approaches is aimed at supporting experimental scientists in the production of experiment reports that are compliant to a checklist (1). These compliant reports can then be published to the Web. It remains however a challenge to quickly and easily assess an arbitrary set of data against these checklists (2). It is our belief that we can exploit the Quality Knowledge contained in MICs to create reusable quality components to assess Linked Data, by making MICs available in RDF (3).

Given a well-structured representation of our checklist we are no longer restricted to data that was explicitly published against a particular MIC. Instead we can attempt to *align* existing data to the checklist and make an assessment of its compliance. Data consumers will be able to make a better interpretation about the quality of the data, automatically check it, and integrate it with greater confidence. We can express in a checklist for example that the reporting of a unique PubMed ID is a minimal requirement for the description of an Assay to be complete. With a machine readable checklist and data, if a PubMed ID is

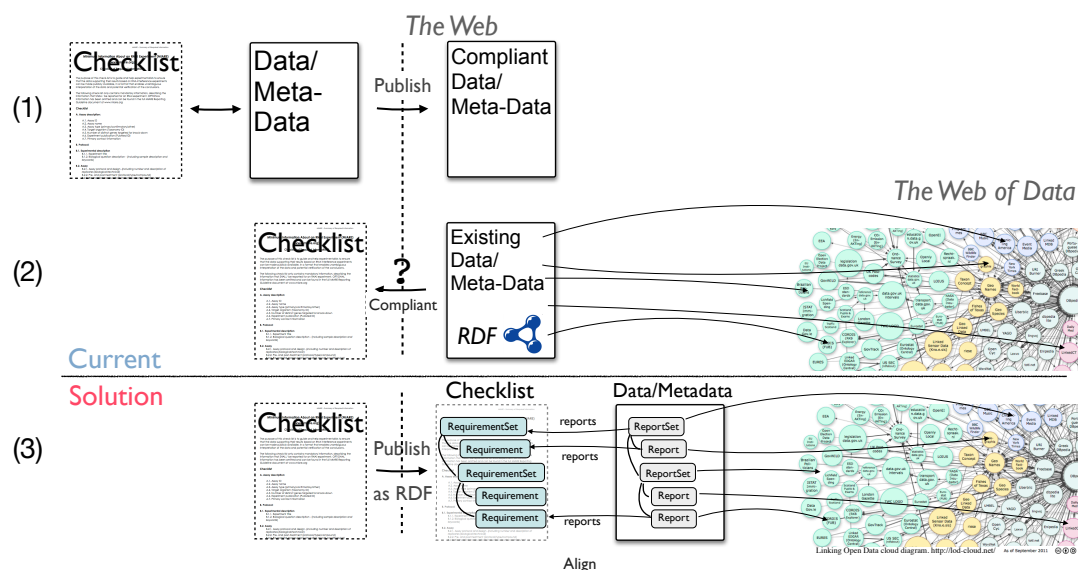


Figure 3.2: A Minimum Information Checklist Solution in the Web of Data.

omitted this flaw in the data can be more readily detected. Crucially we must do this in a way that copes with the wide diversity of MICs, and heterogenous nature of the Web of Data.

We use our OPS classification to guide our solution and identify three components for a MIC-based assessment of published Linked Data: an encoding that enumerates the requirements to be satisfied; an alignment to align parts of the published data that claim to report requirements; and an orchestration process to determine a level of conformance given a set of reports aligned with a MIC.

Quality Knowledge Encoding and **Quality Evidence Alignment** are realized by the Minimum Information Model (MIM) Vocabulary, a meta-modelling vocabulary designed to be used in three ways: (1) To describe a MIC that is specific to some class of data (e.g. MIARE) with the aim of having a library of RDF encoded checklists as community resources for data of various types that can be referred to and used by anybody; (2) To annotate RDF data (e.g. data reporting an RNAi experiment) as reports of requirements; and (3) to express a level of conformance of a group of reports with the requirements of a MIC.

Orchestration is realized by a prototype implementation of a framework that examines requirements reported using the MIM vocabulary to calculate an assessment of conformance to a MIC.

We have developed a case study to evaluate the completeness of the chemical

compound data that is maintained by the WikiProject Chemicals task force in Wikipedia. The goals of our case-study are two fold:

- 1 To demonstrate the ability of the MIM Vocabulary and Framework to support the assessment of Linked Data “in the wild” against a well-structured checklist.
- 2 To investigate the the interactions between the IQ Life-cycle and the Linked Data extraction process.

In the next section, we introduce our WikiProject Chemicals case-study.

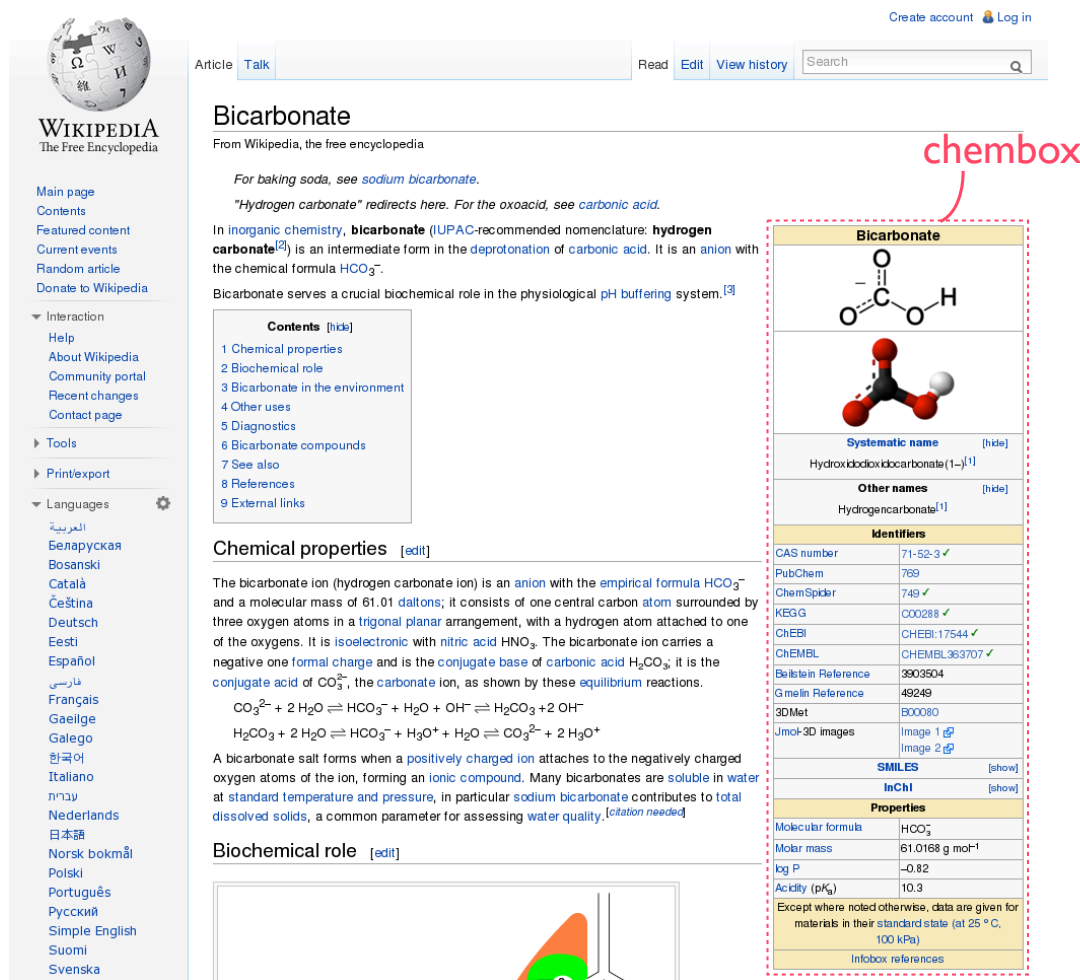
A Note on Scope

The scope of our approach, and MICs in general, is to validate the *reporting* of requirements, and not the *correctness* of the reported information. However, the validation of data completeness is an important first step to increasing the quality of scientific information on the Web of Data.

3.3 A WikiProject Chemicals Case-Study

A collaboration with the Royal Society of Chemistry (RSC) as part of the OpenPHACTS project has presented us with the challenge of addressing the quality of published chemical compound data. In the introduction of this thesis we discussed the current state of online chemical structure repositories. In particular we highlighted issues related the recently released NPC browser and errors found by members of the chemistry community. A root cause of some of the issues found in the NPC browser, and online chemical data in general is that crucial parts of the data and metadata are being thrown-away before publishing [Cla11]. This view is echoed by subsequent calls for the community to adhere to some form of Minimum Information Checklist standard [WE11][FMF⁺12].

It is consensus in the Chemistry community that a good quality description about a chemical compound provides an InChI identifier. The ChemSpider database enforces this requirement and will not allow entries that do not provide an InChI identifier. However, given the open nature of the Web of Data we lose the ability to enforce this requirement. When publishing compound data an individual may discard this InChI. As an example of the varying completeness of chemical data online consider the chemical structure data provided by



The image shows a screenshot of the Wikipedia article for "Bicarbonate" and its associated Chembox. The Chembox is highlighted with a red dashed border and a red arrow labeled "chembox".

Wikipedia Article Content:

Bicarbonate
From Wikipedia, the free encyclopedia

For baking soda, see *sodium bicarbonate*.
 "Hydrogen carbonate" redirects here. For the oxoacid, see *carbonic acid*.

In *inorganic chemistry*, **bicarbonate** (IUPAC-recommended nomenclature: **hydrogen carbonate**^[2]) is an intermediate form in the *deprotonation* of *carbonic acid*. It is an anion with the chemical formula HCO_3^- .

Bicarbonate serves a crucial biochemical role in the physiological *pH buffering* system.^[3]

Contents [hide]

- 1 Chemical properties
- 2 Biochemical role
- 3 Bicarbonate in the environment
- 4 Other uses
- 5 Diagnostics
- 6 Bicarbonate compounds
- 7 See also
- 8 References
- 9 External links

Chemical properties [edit]

The bicarbonate ion (hydrogen carbonate ion) is an *anion* with the *empirical formula* HCO_3^- and a molecular mass of 61.01 *daltons*; it consists of one central carbon atom surrounded by three oxygen atoms in a *trigonal planar* arrangement, with a hydrogen atom attached to one of the oxygens. It is *isoelectronic* with *nitric acid* HNO_3 . The bicarbonate ion carries a negative one *formal charge* and is the *conjugate base* of *carbonic acid* H_2CO_3 ; it is the *conjugate acid* of CO_3^{2-} , the *carbonate ion*, as shown by these *equilibrium reactions*.

$$\text{CO}_3^{2-} + 2 \text{H}_2\text{O} \rightleftharpoons \text{HCO}_3^- + \text{H}_2\text{O} + \text{OH}^- \rightleftharpoons \text{H}_2\text{CO}_3 + 2 \text{OH}^-$$

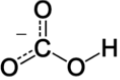
$$\text{H}_2\text{CO}_3 + 2 \text{H}_2\text{O} \rightleftharpoons \text{HCO}_3^- + \text{H}_3\text{O}^+ + \text{H}_2\text{O} \rightleftharpoons \text{CO}_3^{2-} + 2 \text{H}_3\text{O}^+$$

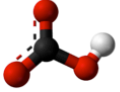
A bicarbonate salt forms when a *positively charged ion* attaches to the negatively charged oxygen atoms of the ion, forming an *ionic compound*. Many bicarbonates are *soluble in water* at *standard temperature and pressure*, in particular *sodium bicarbonate* contributes to *total dissolved solids*, a common parameter for assessing *water quality*.^[*citation needed*]

Biochemical role [edit]

Chembox Content:

Bicarbonate







Systematic name [hide]
Hydroxidodioxidocarbonate(1-)^[1]

Other names [hide]
Hydrogencarbonate^[1]

Identifiers

CAS number	71-52-3 ✓
PubChem	769
ChemSpider	749 ✓
KEGG	C00288 ✓
ChEBI	ChEBI:17544 ✓
ChEMBL	ChEMBL363707 ✓
BioRxiv Reference	3903504
Pubmed Reference	49249
3D Mol	800080
Jmol-3D images	Image 1  Image 2 
SMILES	[show]
InChI	[show]
Properties	
Molecular formula	HCO_3^-
Molar mass	61.0168 g mol ⁻¹
log P	−0.82
Acidity (pK _a)	10.3

Except where noted otherwise, data are given for materials in their standard state (at 25 °C, 100 kPa)

Infobox references

Figure 3.3: The Wikipedia Article and Chembox for Bicarbonate

Wikipedia, and the subsequent Linked Data extraction provided by the DBpedia project. Wikipedia is steadily becoming a valuable reference resource for chemical compound data [MR08]. As an open community data resource the data available is of varying quality and completeness.

In Wikipedia an *Infobox* is a set of semi-structured data that is commonly included in an article as a table on the right-hand side (see for example Figure 3.3). It is the semi-structured data available in Infoboxes that is used by the DBpedia project to generate their Linked Data representation of Wikipedia. To ensure that articles of the same type provide consistent content, authors can reuse Infobox *templates* that provide blank versions of the Infobox to be populated with data. The chemistry community on Wikipedia currently makes use of an Infobox template called *chembox*. The chembox template acts like a MIC

```
1 {{Chembox
2   ImageFile =
3   ImageSize =
4   ImageAlt =
5   IUPACName =
6   OtherNames =
7   Section1 = {{Chembox Identifiers
8     CASNo =
9     PubChem =
10    SMILES = }}
11  Section2 = {{Chembox Properties
12    Formula =
13    MolarMass =
14    Appearance =
15    Density =
16    MeltingPt =
17    BoilingPt =
18    Solubility = }}
19  Section3 = {{Chembox Hazards
20    MainHazards =
21    FlashPt =
22    Autoignition = }}
23 }}
```

Listing 3.1: The Simple chembox MediaWiki Template

for chemical compound articles by providing guideline for which information to include. Figure 3.3 shows an example chembox from the Wikipedia article for the Bicarbonate compound. The ‘simple’ chembox template details a set of 15 reporting requirements that the community recommends for inclusion when creating a chemical compound article. Listing 3.1 details the 15 requirements of the simple chembox in MediaWiki markup format.

Beyond the simple version the community has also defined two further extended versions of the chembox template. The chembox templates are a clear example of objective Quality Knowledge that has been defined and agreed upon by a community². To create a new article an author copies and populates the template in order to create a new chembox. The open nature of Wikipedia means that the author is free to populate as many, or as few properties of the chembox as they wish.

Since 2008, members of the Wikipedia task-force WikiProject Chemicals [WPC12] have made attempts to assess the quality of the chemical compound articles through a process of manual inspection. The goal of the task force is to improve the general quality of chemistry articles on Wikipedia. The members are concerned with the completeness and accuracy of the article text and chembox data.

²It is even possible to observe from the Wikipedia talk pages (discussion pages for articles) related to chembox http://en.wikipedia.org/wiki/Template_talk:Chembox the reasoning that has gone into forming a consensus for the requirements

Articles are rated a 5 point scale of Stub, Start, C, B, and A. Figure 3.6 illustrates for example the significant difference in the amount of content between a Stub article for the compound Aluminium Hydroxide oxide, and an A rated article for Acetic Acid. Out of 11,409 relevant chemical articles identified by the task force there are currently 5030 articles rated as Stub, 2,355 as Start, 52 as C, 478 as B and only 16 rated as A. As of writing (December 2013) there still remain almost 3000 articles un-assessed.

As a result of the scale of the chemistry content in Wikipedia the manual assessment of its quality is a time consuming process. Article quality ratings can also quickly become out of date due to the dynamic nature of the content. Many of the article ratings for example date from 2009 or earlier. We have therefore chosen to demonstrate the MIM framework by assessing the completeness of the data available on these pages with a MIC developed for chemical compound reporting (described in section 3.3.2). To do so we use a structured Linked Data extraction of the data available in the chemboxes. We can not as easily assess the unstructured article text, but by using a Linked Data extraction of the Wikipedia chembox data our MIC assessment can serve as a first level of quality assessment, checking the chemboxes for metadata completeness.

The DBpedia project provides a popular and widely used Linked Data extraction of the information provided in the Infoboxes of Wikipedia pages. This extraction is performed using the DBpedia Knowledge Extraction framework [BLK⁺09]. The framework makes use of mapping files that describe mappings from properties in an Infobox template, to properties in the DBpedia vocabulary. The Listing below for example illustrates an extract from the Chembox mapping for melting point data:

```
1 {{PropertyMapping | templateProperty = MeltingPtK | ontologyProperty =  
   meltingPoint | unit = Temperature }}
```

This mapping states that the chembox property **MeltingPt** will map to the DBpedia vocabulary property <http://dbpedia.org/property/meltingPoint>. These mapping files are available online at mappings.dbpedia.org and can be edited by members of the community, demonstrating an example of the “pay-as-you-go” nature of the Web of Data.

Listing 3.2 shows an extract from the DBpedia resource <http://http://live.dbpedia.org/resource/Bicarbonate> which is the Linked Data extraction of the Bicarbonate page from Figure 3.3. An inspection of the listing reveals that

```

1 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2 @prefix dbpedia-owl: <http://dbpedia.org/ontology/> .
3 @prefix dbpedia: <http://dbpedia.org/resource/> .
4 @prefix foaf: <http://xmlns.com/foaf/0.1/> .
5 @prefix wiki-en: <http://en.wikipedia.org/wiki/> .
6 @prefix owl: <http://www.w3.org/2002/07/owl#> .
7 @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
8 @prefix dcterms: <http://purl.org/dc/terms/> .
9 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
10 @prefix ns12: <http://www.wikidata.org/entity/> .
11 @prefix dbpprop: <http://dbpedia.org/property/> .
12 @prefix ns22: <http://wifo5-03.informatik.uni-mannheim.de/flickrwrappr/photos/> .

13 @prefix template-en: <http://dbpedia.org/resource/Template:> .
14 @prefix category-en: <http://dbpedia.org/resource/Category:> .

15
16 dbpedia:Bicarb
17   dbpedia-owl:wikiPageRedirects dbpedia:Bicarbonate .
18
19 dbpedia:Bicarbonate
20   dbpedia-owl:casNumber "71-52-3"@en ;
21   dbpedia-owl:wikiPageID 3982 ;
22   dbpedia-owl:wikiPageRevisionID 572739319 ;
23   dbpprop:hasPhotoCollection ns22:Bicarbonate ;
24   dbpprop:imagefile "Bicarbonate-ion-3D-balls.png"@en, "Bicarbonate-resonance.
    png"@en ;
25   dbpprop:imagename "Ball and stick model of bicarbonate"@en,
26   "Skeletal formula of bicarbonate with the explicit hydrogen added"@en ;
27   dbpprop:imagesize 121 ;
28   dbpprop:othernames "Hydrogencarbonate"@en ;
29   dbpprop:systematicname "Hydroxidodioxidocarbonate"@en ;
30   dbpprop:wikiPageUsesTemplate template-en:Chembox, template-en:
    Citation_needed,
31   template-en:Clear-left, template-en:Clinical_biochemistry_blood_tests,
32   template-en:Convert, template-en:Eqm, template-en:For, template-en:Ionbox,
    template-en:MeshName, template-en:Oxides_of_carbon, template-en:Redirect
    , template-en:Reflist, template-en:Wiktionary ;
33   dcterms:modified "2013-11-04T12:01:16Z"^^xsd:dateTime ;
34   dcterms:subject category-en:Salts ;
35   a dbpedia-owl:ChemicalCompound, dbpedia-owl:ChemicalSubstance, owl:Thing ;
36   rdfs:label "Bicarbonate"@en ;
37   foaf:isPrimaryTopicOf wiki-en:Bicarbonate .
38
39 wiki-en:Bicarbonate
40   foaf:primaryTopic dbpedia:Bicarbonate .

```

Listing 3.2: Extract from DBpedia Resource for the Bicarbonate Article

only 5 of 19 properties listed in the original Bicarbonate articles chembox are included in its DBpedia representation.

The reasons for the incomplete data can be traced to the mapping used by the DBpedia extraction framework. Firstly the mapping for the chembox template is currently incomplete and there are properties defined in the chembox templates that do not have corresponding mappings in the mapping file. A second reason is related to the usage of the chembox template by article authors. The chembox template has changed over time, and has been inconsistently applied by the

authors of articles. As a result there are properties used in many articles that do not appear in the chembox template. Consider for example the melting point data described in the articles for Ethane³ and Acetic Acid⁴.

The melting point data for Ethane is included in the chembox as follows:

```
1 | MeltingPtK = 90.4
```

In contrast the melting point data for Acetic Acid uses two properties to indicate a range of melting point values from high `MeltingPtKH` to low `MeltingPtKL`:

```
1 | MeltingPtKL = 289
2 | MeltingPtKH = 290
```

This subtle difference in representation is not captured in the DBpedia mapping, and causes the data to be omitted from the resulting Linked Data extraction. The extraction process therefore directly affects the quality of the Linked Data representation and as a result, will affect the results of a MIC assessment. This is likely to be an issue across many scientific resources where a Linked Data representation is typically generated from a primary existing source. Whilst a MIC assessment provides an indication of the completeness of the underlying data, it is clear that we are in fact assessing the quality of the Linked Data extraction. Quality issues related to the Linked Data extraction process will therefore also have an impact when developing a MIC.

3.3.1 Extending the IQ Life-cycle

Figure 3.4 illustrates the interaction between the engineering process and the Linked Data extraction process. Engineering in this case is the process of designing a MIC-based solution that can evaluate the completeness of a Linked Data extraction. Consider the following case: When designing a MIC we perform a test assessment of the extraction of Acetic Acid. One of the requirements of the MIC is for melting point data. The result of the assessment is that the Melting Point requirement is not satisfied. This result may be due to one of three reasons:

1. A fault with our engineering of the MIC solution.
2. A fault with the Linked Data extraction.
3. The information is actually missing from the original Acetic Acid data source.

³<http://en.wikipedia.org/w/index.php?title=Ethane&oldid=579889733>

⁴http://en.wikipedia.org/w/index.php?title=Acetic_acid&oldid=585316574

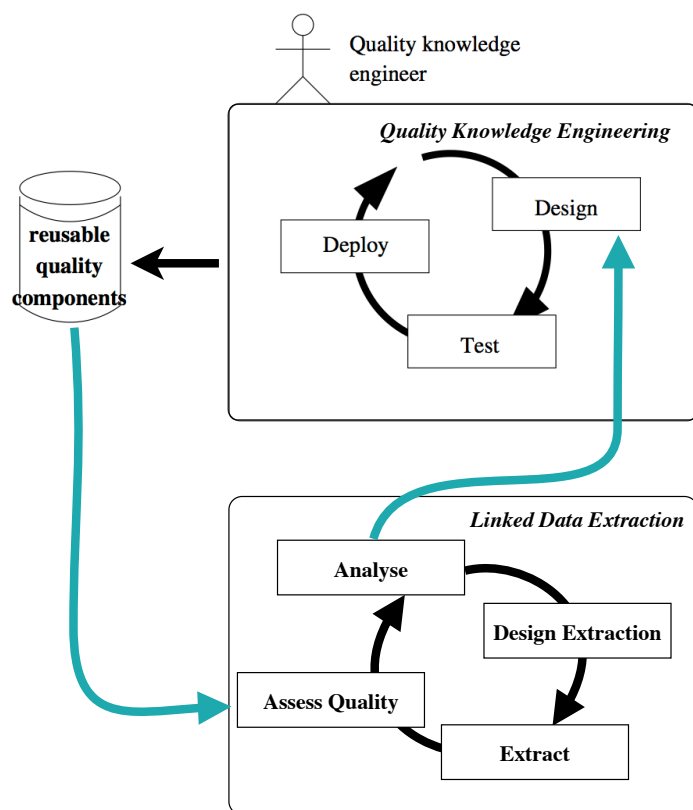


Figure 3.4: Interaction Between the IQ Life-cycle and Linked Data Extraction processes.

We can eliminate the last possibility by establishing a ground truth for a subset of data. We know for example in this case that the original article for Acetic Acid *does* contain melting point data. Any discrepancies between the assessment result and the ground truth will trigger a process of fault-finding in either the Linked Data extraction or engineering. We demonstrate this process further in the results of our case study in section 3.7.1.

3.3.2 chemmim: A Checklist for Chemical Compound Data

We illustrate the rest of this chapter with examples drawn from chemmim, a checklist we have developed for our case study. The chemmim checklist details 11 meta-data reporting requirements for publishing chemical compound data on the Web of Data. Our chemmim checklist is designed to be representative of the meta-data typically required when reporting chemical compound data. Taking

Chemmin
1. Identifiers
1.1 InChI
1.2 SMILES
1.3 PubChem ID
1.4 Chemspider ID
2. Properties
2.1 Melting Point
2.2 Molar Mass
2.3 Solubility
2.4 Formula
3. IUPAC Name
4. Image
5. Synonyms

Figure 3.5: The 11 Requirements of the Chemmin Checklist.

a number of shared elements from the simple chembox template, and an existing checklist for reporting the Minimum Information about a Bioactive Entity (MIABE) (specifically the molecule properties section of the checklist) the full resulting chemmin checklist is shown in Figure 3.5 The checklist groups some of these requirements into sets: *Identifiers* InChI, SMILES, ChemSpider ID, PubChem ID; and *Properties* - Molecular Formula, Molar Mass, Melting Point and Solubility. There are also three further requirements, IUPAC Name, Image, and Synonyms.

For identifiers a compound description must provide an International Chemical Identifier (InChI). The InChI is a textual identifier which encodes chemical information in a standard human readable manner to facilitate both storing and searching for chemical data. The InChI ID for the chemical Ethane for example is “1S/C2H6/c1-2/h1-2H3”. The Simplified Molecular-Input Line-Entry System (SMILES) ID is an older alternative textual identifier similar to the InChI which maintains significant support. An example SMILES id for Ethane is “CC”. The checklist also requires two IDs for popular online chemical structure data bases ChemSpider and PubChem.

The checklist requires four chemical properties. Melting point, molar mass, and solubility are all numerical values. The molecular formula is the traditional representation of a chemical structure e.g. C₂H₆ for the chemical Ethane.

The IUPAC name is the designated name for the compound according to the International Union of Pure and Applied Chemistry (IUPAC). Synonyms are then

additional textual representations that are used by the community to refer to the same chemical compound.

Finally the expected report for the Image requirement is a URL that provides a visual representation of the chemical compound.

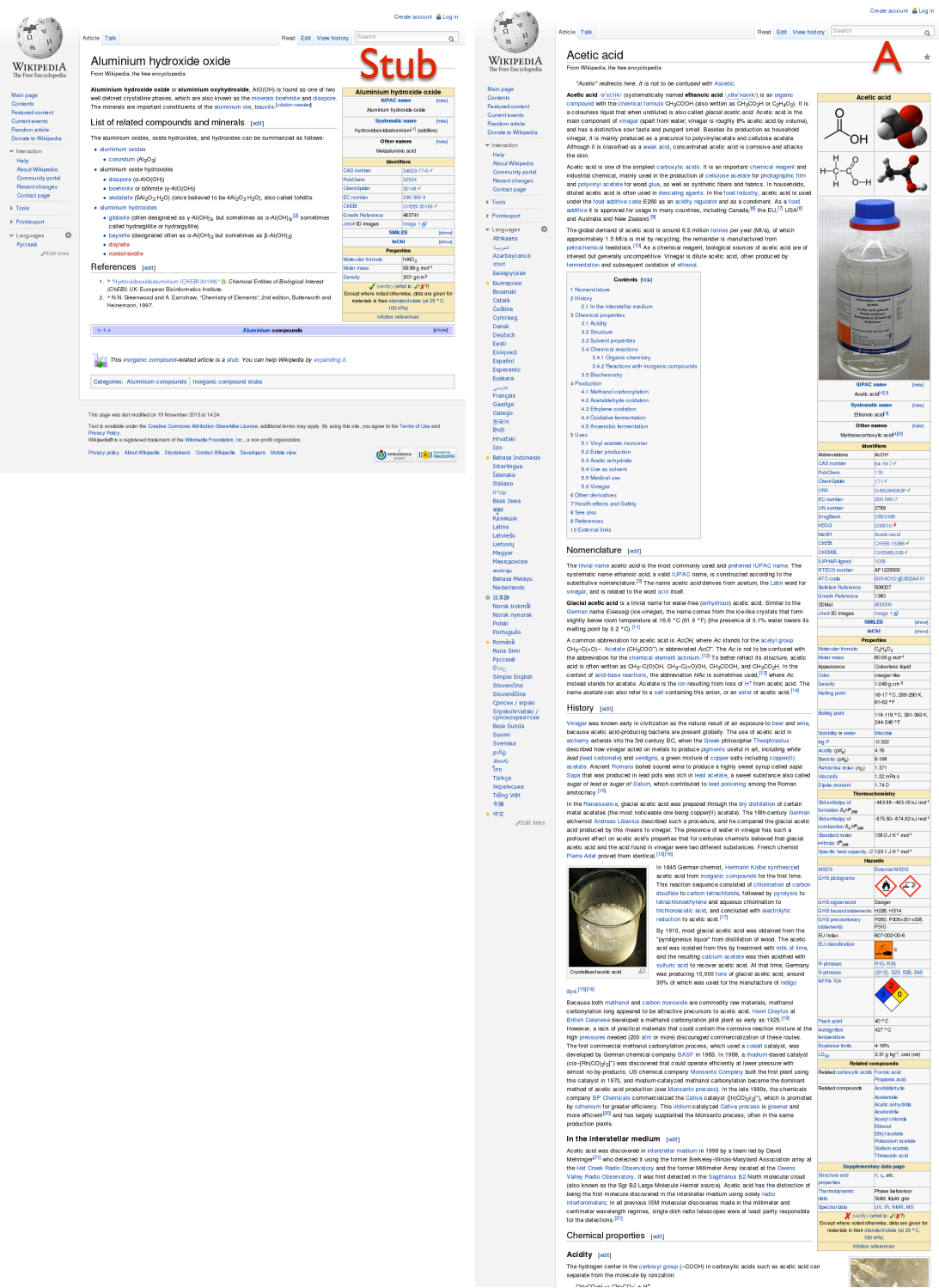


Figure 3.6: Comparison of Wikipedia Article for Aluminium Hydroxide Oxide (Stub) and Acetic Acid (A).

3.4 Minimum Information Checklist Structural Meta-Study

The MIBBI group was the first to perform a meta-study of MICs [TFS⁺08]. The focus of the MIBBI study was on the conceptual content of the checklists, for example 17 of the MICs assessed by MIBBI contained requirements related to experimental study design. The authors highlighted a significant conceptual overlap between the checklists and the potential for cross-community reuse of sub-sections of checklists. From the 50 checklists listed on biosharing.org at the time of our analysis (January 2012) we have successfully analysed 41 checklists. Of the 9 checklists we did not assess 6 were in draft or unreleased, two were unavailable for download, and 1 was an implementation guideline, rather than a reporting guideline.

	Requirements	Requirement Sets	Fine	Medium	Coarse	Authors		Requirement Levels	Vocab/Data Restrictions	Cardinality Restrictions
Average	60	13	38	10	10	18	Coverage	15	31	10
Max	242	65	233	60	30	98				
Min	5	1	0	0	0	2				

Table 3.1: Summary of Minimum Information Checklist Analysis

In this study we have focused on the structural overlap of MICs in order to establish the components required to design a meta-modelling vocabulary sufficient to model MICs. We began our study by analysing a small number of the checklists in detail, identifying common features. We then reviewed the rest of the checklists for their use and specific implementation of the identified features. As we reviewed checklists we continued to revise the list of common features until we established a core set. Table 3.1 provides a summary of our checklist analysis.

Our analysis has elicited 5 core features that are common across MICs: *requirements*, *requirement sets*, *requirement levels*, *type restrictions* and *cardinality restrictions*. In Figure 3.7 we have annotated the MIARE checklist to illustrate the use of each of these core features.

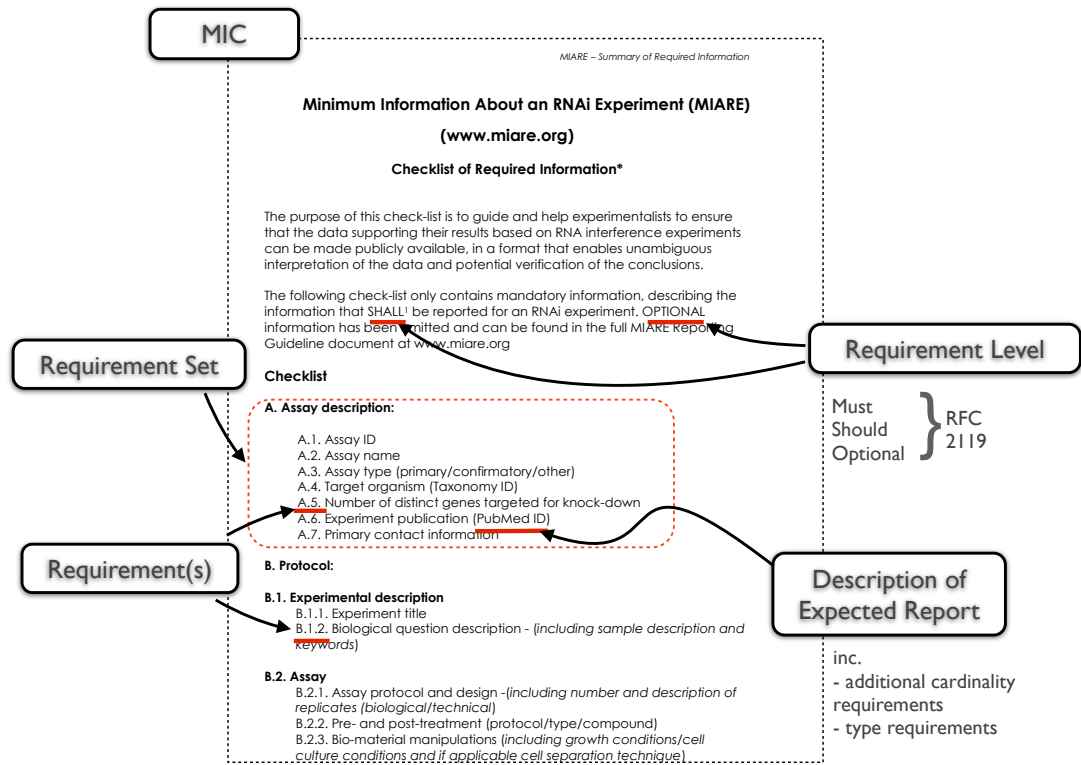


Figure 3.7: The Anatomy of a Minimum Information Checklist.

Requirements A requirement is the atomic unit of a checklist. A requirement such as A.5 or B.1.2 in MIARE indicates the expectation of an experiment description to report a specific piece of data.

In our study we chose to analyse the granularity of the requirements based on the description of the expected report. We did this to gain an insight into the potential effectiveness of our approach. We defined three categories of granularity:

- Fine - Expects a short form answer - e.g. PubMed ID, Title, Assay ID.
- Medium - Requirement could be satisfied by a short form or long form answer e.g. “Were all individuals grown on the same day?” from the MIAME/-Plant checklist for Plant Genomics. This could be answered by a short yes or no response. The checklist does not however restrict a more detailed answer, for example describing each day that individuals were grown.
- Coarse - The requirements can only be satisfied with a long form report - e.g. experiment description.

We believe that our assessment is likely to be more effective for fine grained requirements. This is based upon the assumption that there is a higher likelihood that the report given covers the requirement. From our checklist analysis we observed a higher proportion of fine requirements on average, when compared to medium and coarse requirements.

Requirement Sets. A requirement set is a collection of requirements that relate to a common concept. For example “A. Assay Description” in the MIARE checklist is a requirement set that contains 7 requirements that are related to the description of an Assay. Requirement sets are often hierarchical in a tree-like structure where a requirement set can contain other requirement sets. In the MIARE checklist this can be seen where the requirement set “B. Protocol” contains the requirement sets “B.1 Assay” and “B.2 Experimental Description”. This conceptual grouping of requirements is a finding that is common to both our study and the MIBBI study.

Requirement Levels A prevalent feature of MICs is the enriching of requirements to express a level of importance. MIARE for example states that the PDF version of the checklist (shown in Figure 3.7) contains mandatory requirements that SHALL be reported. There are additional OPTIONAL requirements contained in a further guideline at <http://www.miare.org>. A number of checklists make use of the terms Must, Should and Optional from RFC 2119 [Bra97] (e.g. MIARE [MIA06], and MIFlowCyt [LSB⁺08]). In some cases for example a requirement of Should is used where there may be some commercial or legal restriction preventing that data from always being reported.

RFC 2119 describes how to use the terms MUST, SHOULD, OPTIONAL and their synonyms in specifications. Given the objective of MICs to improve the understanding and reuse of data, we introduce the following interpretation of the terms for MICs:

- **MUST** - The reporting of the requirement is critical for the subsequent reuse and understanding of the data.
- **SHOULD** - If available the reporting of the requirement will improve the understanding and reuse of the data, but its omission will not preclude its reuse.
- **OPTIONAL** The reporting of the requirement is not *required* for the understanding reuse of the data, but may be useful if available.

Type Restrictions. Type restrictions are commonly used to guide the reporter as to the expected information. For example requirement A.6 in MIARE explicitly states that a PubMed ID be used to report the experiment publication. In addition to this checklists often state type restrictions in terms of controlled vocabularies such as the Gene Ontology. The Life Sciences community have a long and significant investment in the use of controlled vocabularies and ontologies to aid interoperability [SCC97][SAR⁺07]. A number of checklist developers are also developing vocabularies in parallel with their checklists to describe the data. The RightField tool also captures this behaviour, enabling the integration of vocabulary restrictions into Excel spreadsheet representations of MICs.

Cardinality Restrictions. Cardinality restrictions indicate the number of reports that are expected for a requirement. For example though not explicitly stated, we might expect that the requirement “A.1 Assay ID” (Figure 3.7) should have one and only one report if it is indeed a unique ID. Any data reporting more than one Assay ID for the “A. Assay Description” requirement set would therefore not satisfy the requirement.

Our analysis of checklists has shown that these 5 features are commonly used across many of the diverse MICs currently available. In the next section, we present the MIM vocabulary which makes use of these common features to define a mechanism for encoding and publishing MICs as RDF.

3.5 The MIM Vocabulary

The MIM Vocabulary (MIMV) is a meta-modelling vocabulary that allows the encoding of arbitrary MICs in RDF. The vocabulary allows us to publish MICs as reusable quality components in the Web of Data. The MIM vocabulary itself has been developed as an OWL-DL ontology (available at <http://purl.org/net/mim/ns> and in Appendix B). Figure 3.8 presents the core classes of the vocabulary and the relationships between them. The vocabulary is intended to be used for three purposes:

- *Checklist description:* Describe the requirements of a MIC.
- *Report description:* Align RDF data with requirements in a MIC.
- *Satisfaction description:* Describe how well RDF data aligned with a MIC satisfies the requirements. The reason that we can propose to annotate the

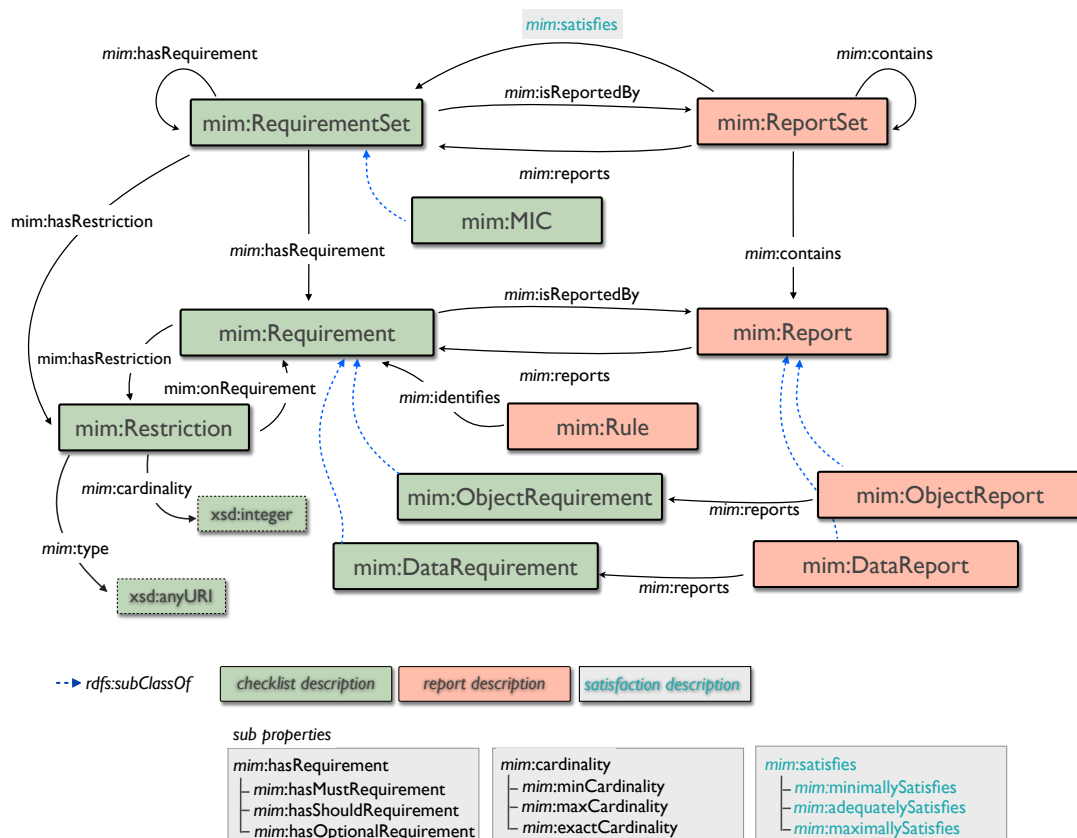


Figure 3.8: The Minimum Information Model Vocabulary.

data with a satisfaction level is because a MIC assessment is an objective assessment and therefore remains *static*.

The figure illustrates how the main features of the vocabulary can be separated into three areas of concern to coincide with these uses. The vocabulary for checklist description has been primarily informed by the structural analysis of existing MICs. The vocabulary for report description has been informed in response to the checklist description and a need to align RDF data with those checklists.

In this section we describe each of the three uses of the vocabulary using the chemmim checklist and RDF data extracted from Wikipedia chemboxes to illustrate throughout (We describe how this data was extracted in section 3.7).

```

1 @prefix mim: <http://purl.org/net/mim/ns#> .
2 @prefix foaf: <http://xmlns.com/foaf/0.1/> .
3 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
4 @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
5 @prefix : <http://purl.org/net/chembox/chemmim#> .
6
7 :MIC rdf:type mim:MIC ;
8   mim:hasMustRequirement
9     :Identifiers , :Properties ;
10  mim:hasOptionalRequirement
11    :Synonym ;
12  mim:hasShouldRequirement
13    :IUPACName , :Image ;
14  mim:hasRestriction
15    [ mim:exactCardinality
16      1 ;
17      mim:onRequirement :Identifiers ,
18        :Properties
19    ] .
20
21 :Identifiers rdf:type mim:RequirementSet ;
22   mim:hasMustRequirement
23     :InChI , :SMILES ;
24   mim:hasShouldRequirement
25     :ChemSpider , :PubChem ;
26   mim:hasRestriction
27     [ mim:exactCardinality
28       1 ;
29       mim:onRequirement :InChI , :SMILES
30     ] .
31
32 :SMILES rdf:type mim:DataRequirement .
33
34 :InChI rdf:type mim:DataRequirement ;
35   mim:hasRestriction
36     [ mim:onSelf "true"^^xsd:boolean ;
37       mim:type xsd:string
38     ] .
39
40 :Image rdf:type mim:ObjectRequirement .
41   mim:hasRestriction
42     [ mim:onSelf "true"^^xsd:boolean ;
43       mim:instanceOf foaf:Image
44     ] .

```

Listing 3.3: The chemmim Checklist Encoded Using the MIM Vocabulary (extract).

3.5.1 Describing a Checklist

The construction of a MIC serves to specify the minimal set of data to provide when reporting a particular class of data. Listing 3.3 details an extract from our chemmim MIC encoded in the MIM Vocabulary. The full encoding of the checklist is available in Appendix C. In the rest of this section we show how the vocabulary can be used to encode each of the core features of a MIC.

Requirements The class `mim:Requirement` is used to describe individual requirements that we expect to be reported by RDF data. In RDF data there

are two possible types of value that can represent data, an RDF URI or an RDF Literal Value. In the vocabulary we have therefore created two sub classes of `mim:Requirement`, `mim:ObjectRequirement` and `mim:DataRequirement`. The naming is borrowed from OWL vocabulary terminology (Object Properties and Data Properties) and reflects the type of value we expect for the requirement in reporting data. For example the InChI requirement in chemmim is defined as a `DataRequirement`:

```
34 :InChI rdf:type mim:DataRequirement ;
```

Listing 3.4: Declaring a `mim:DataRequirement`

This is where we expect the report to be an RDF literal value, for example in our chembox extraction the data is encoded as follows:

```
1 chembox:Ethane chembox:StdInChI "1S/C2H6/c1-2/h1-2H3"^^xsd:string ;
```

In contrast the requirement for an Image is defined as an `ObjectRequirement`:

```
40 :Image rdf:type mim:ObjectRequirement .
```

Listing 3.5: Declaring a `mim:ObjectRequirement`

In this case we expect a report to be an RDF URI for example:

```
1 chembox:Ethane chembox:ImageFileL1 <http://en.m.wikipedia.org/wiki/File:Ethane-A-3D-balls.png>;
```

Type Restrictions Restrictions in the vocabulary (type and cardinality) are described as an attribute of a `mim:Requirement` using an N-ary relation with the property `mim:hasRestriction` and the class `mim:Restriction`. We have followed use-case 1 in the Semantic Web Best Practice Note [NRHW06] for the definition of N-ary relationships. The full InChI requirement description for example restricts the expected data type to `xsd:string` from the XML data type specification [BMC⁺04]:

```
34 :InChI rdf:type mim:DataRequirement ;
35     mim:hasRestriction
36         [ mim:onSelf "true"^^xsd:boolean ;
37           mim:type xsd:string
38         ] .
```

Listing 3.6: Declaring a Type Restriction on the InChI Requirement.

This means that to satisfy the requirement any report of an InChI must be declared as type `xsd:string`. The Image requirement similarly describes a type restriction but as an `ObjectRequirement` restricts reports to a class in a controlled vocabulary:

```
40 :Image rdf:type mim:ObjectRequirement .
41   mim:hasRestriction
42     [ mim:onSelf "true"^^xsd:boolean ;
43       mim:instanceOf foaf:Image
44     ] .
```

Listing 3.7: Declaring a Type Restriction on the `Image` Requirement.

As a result a report of an Image must be declared as the type `foaf:Image`.

Requirement Sets and Requirement Levels. Requirement sets are declared using the class `mim:RequirementSet`. For example the `chemmim` defines a requirement set for chemical Identifiers:

```
21 :Identifiers rdf:type mim:RequirementSet ;
```

Listing 3.8: Declaring the `Identifiers` Requirement Set.

To describe a requirement as part of a requirement set we use one of the sub-properties of `mim:hasRequirement`: `hasMustRequirement`, `hasShouldRequirement`, and `hasOptionalRequirement`. For example the `Identifiers` requirement set contains four requirements:

```
21 :Identifiers rdf:type mim:RequirementSet ;
22   mim:hasMustRequirement
23     :InChI , :SMILES ;
24   mim:hasShouldRequirement
25     :ChemSpider , :PubChem ;
```

Listing 3.9: Declaring the Requirements that are Members of the `Identifiers` Requirement Set.

The use of these properties also defines the requirement level of the corresponding requirements. Therefore the `Identifiers` requirement set declares `InChI` and `SMILES` as *Must* requirements and `ChemSpider` and `PubChem` as *Should* requirements. In order to satisfy the `Identifiers` set, any data reporting it *must* therefore report an `InChI` value and a `SMILES` value, and *should*, if they are available, report `ChemSpider` and `PubChem` IDs.

Cardinality Restrictions. Cardinality restrictions are also declared as an N-ary relation using `mim:hasRestriction` and `mim:Restriction`. In contrast to vocabulary restrictions, cardinality restrictions are declared as an attribute of `RequirementSets` and uses the `mim:onRequirement` property to declare the `Requirement` it is for. For example the `Identifiers` set declares a cardinality restriction on the requirements `InChI` and `SMILES` as follows:

```
26   mim:hasRestriction
27     [ mim:exactCardinality
```

```
28         1 ;  
29     mim:onRequirement :InChI , :SMILES  
30 ] .
```

Listing 3.10: Declaring a Cardinality Constraint on the `InChI` and `SMILES` Requirements in the `Identifiers` Requirement Set

This restriction states that any data reporting the `Identifiers` requirement set must report exactly one `SMILES` value and exactly one `InChI` value.

By eliciting these five core vocabulary features from our checklist survey the MIM vocabulary supports the encoding of checklists as machine readable RDF. These checklist descriptions can then be published in the Web of Data. Once published each `Requirement` and `RequirementSet` becomes a uniquely identifiable Linked Data resource that data publishers may align data against. In the following section, we describe the vocabulary features that allow us to align RDF data against a checklist.

3.5.2 Reporting Against a Checklist

The objective when reporting against a MIC is to align existing RDF data with the checklist, making claims about which parts of the data report the requirements specified. In the vocabulary, we have defined two main classes to do this, `mim:Report` and `mim:ReportSet`. The class `mim:Report` also has two sub-classes, `mim:ObjectReport` and `mim:DataReport`.

Reports. We have already considered the two types of possible value in RDF data in the checklist description features of the vocabulary with `mim:ObjectRequirement` and `mim:DataRequirement`. We also define two types of report.

- `mim:DataReport` - to annotate RDF literal values and align them with `mim:DataRequirements`.
- `mim:ObjectReport` to annotate RDF URIs and align them with `mim:ObjectRequirements`.

Figure 3.9 shows an example reporting the `chemmim:Image` requirement with data from the chembox <http://purl.org/net/chembox/Ethane>. In this example we describe an `ObjectReport` by annotating the image's URI as the type `mim:ObjectReport` and align it to the checklist requirement using the `mim:reports` property.

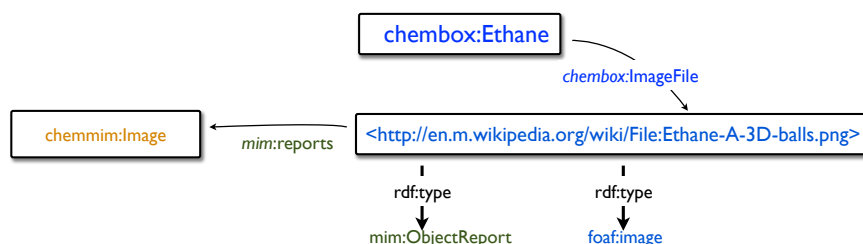


Figure 3.9: Creating an ObjectReport and Aligning it with a Checklist Requirement.

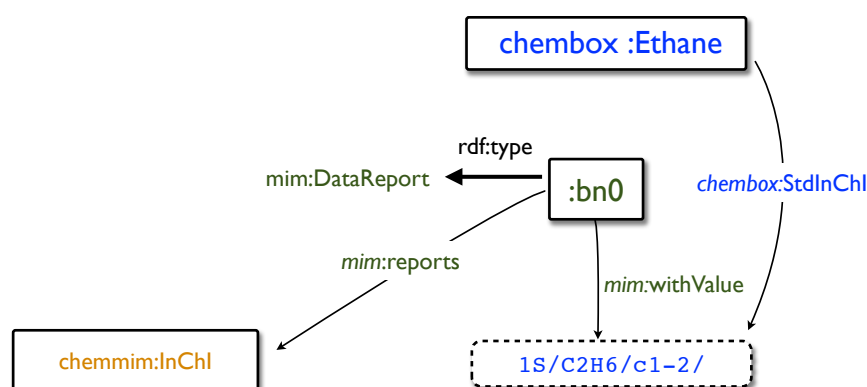


Figure 3.10: Creating a DataReport using a blank node and Aligning it with a Checklist Requirement.

To describe an RDF literal as a **DataReport** we are not able to link directly from the value as we have done with a **ObjectReport**. Figure 3.10 illustrates how we represent a **DataReport** instead as an RDF blank node. We use the **mim:reports** property to align the blank node with the checklist requirement as previously, and use the property **mim:withValue** to link the blank node to the actual reporting value.

Report Sets A report set is a collection of **Reports** that claim to report the requirements in a **RequirementSet**. We group reports together because in a given RDF graph there may be for example multiple instances of a chemical compound, each reporting an InChI value and SMILES value. So that we can subsequently evaluate completeness and cardinality constraints it is necessary to identify coherent report sets, and align them with the corresponding requirement set in our checklist.

As a concrete example consider the RDF presented in Fig. 3.11. This example

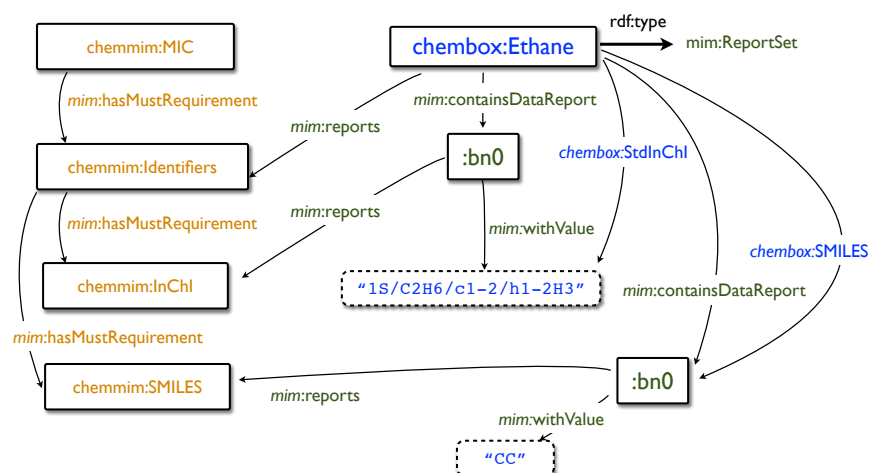


Figure 3.11: Creating a ReportSet and Aligning it with a Checklist Requirement Set.

shows two reports, one for `chemmmim:InChI` and one for `chemmmim:SMILES` grouped together in a `ReportSet` and aligned with the `chemmmim:Identifiers` requirement set. This is done by annotating the URI for the chembox itself `chembox:Ethane` as the type `mim:ReportSet` and using the property `mim:ContainsDataReport` to state that the two reports are part of this report set. We align the report set `chembox:Ethane` with the requirement set `chemmmim:Identifiers` using the `mim:reports` property. By grouping the two reports together into a coherent set we are able to evaluate the completeness i.e. whether it reports all of the requirements in `chemmmim:Identifiers`, and cardinality constraints by counting how many times each requirement is reported.

Report Generating Rules The task of aligning data with a checklist is only feasible by hand annotation for small-scale data. To retrospectively align existing data with MICs we anticipate the need to automate this process. The vocabulary terms `mim:Rule` and `mim:identifies` serve as a mechanism to align rules for report generation with particular requirements in the checklist. Figure 3.12 for example shows the use of these terms to associate a URI that represents a SPARQL query, with the `chemmmim:InChI` requirement. The vocabulary is agnostic to the particular implementation of the rule mechanism. The only requirement is that the rules can be uniquely identified with a URI.

Using this mechanism we can create a collection of rules for a particular

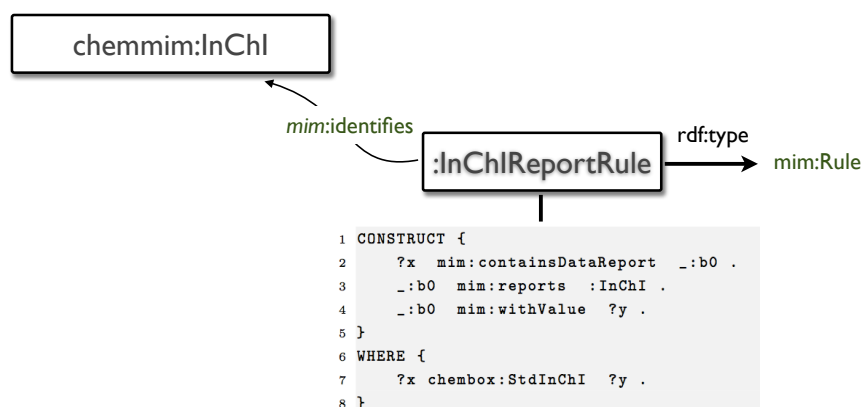


Figure 3.12: Associating the InChI Requirement with a SPARQL Query to Automatically Generate Reports.

dataset and associate them with each requirement in a checklist. We demonstrate this in our subsequent case-study by using a collection of SPARQL-based rules to align our chembox data with the chemmim checklist.

Furthermore, we can create additional collections of rules for different datasets that report chemical structure data, allowing the same checklist to be used across a variety of data representations.

3.5.3 Checklist Satisfaction

In this section, we describe the semantics of checklist satisfaction, and show how the vocabulary can be used to describe checklist satisfaction. In the next section we discuss an implementation of the semantics using a SPARQL-based rule mechanism.

The starting state for evaluating satisfaction is some RDF data aligned with a MIC using the reporting features described in the previous section. The goal is to evaluate how well the identified reports in the data satisfy the checklist. Satisfaction is a two phase process concerning: *Requirement satisfaction* and *Requirement Set satisfaction*.

Requirement satisfaction. The task of requirement satisfaction is to evaluate whether reports such as the blank node `:bn0` in Figure 3.10 satisfy a requirement it is aligned with. The conditions for a report to satisfy a requirement are:

1. The report must be declared as the type `mim:DataReport` or `mim:`

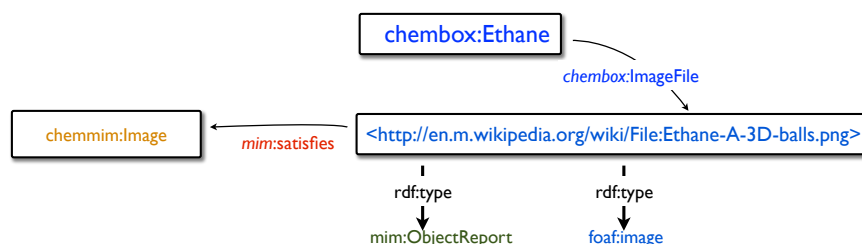


Figure 3.13: Requirement Satisfaction.

ObjectReport using `rdf:type`.

2. The report must use the term `mim:reports` to indicate the requirement it claims to satisfy.
3. The report must meet any type restrictions defined by that requirement. If it is an `ObjectReport` then the type must be declared on the `ObjectReport` using `rdf:type`. If the report is a `DataReport` then then any required data type must be defined for the RDF literal value declared by the `DataReport` using `mim:withValue`.

If the report meets the above conditions then we use the vocabulary term `mim:satisfies` to indicate this as shown in Figure 3.13.

Requirement Set satisfaction. The task for requirement set satisfaction is to evaluate whether `ReportSet` structures such as the one in Figure 3.11 satisfy the `RequirementSet` that they are aligned with. This is a process of checking whether the `ReportSet` contains reports that satisfy each requirement in the `RequirementSet`. As a result of the three requirement levels: *Must*, *Should*, and *Optional*, we have defined three levels of satisfaction, *Maximally satisfies*, *Adequately satisfies* and *Minimally satisfies*. We first define the conditions for *Maximally satisfies*:

1. The report set must be declared as the type `mim:ReportSet` using `rdf:type`.
2. The report set must use the term `mim:reports` to indicate the `RequirementSet` it claims to satisfy.
3. For *all* *Must*, *Should* and *Optional* requirements defined as part of the `RequirementSet`, the report set must contain a corresponding report using the term `mim:containsDataReport` or `mim:containsObjectReport`, that satisfies the requirement.

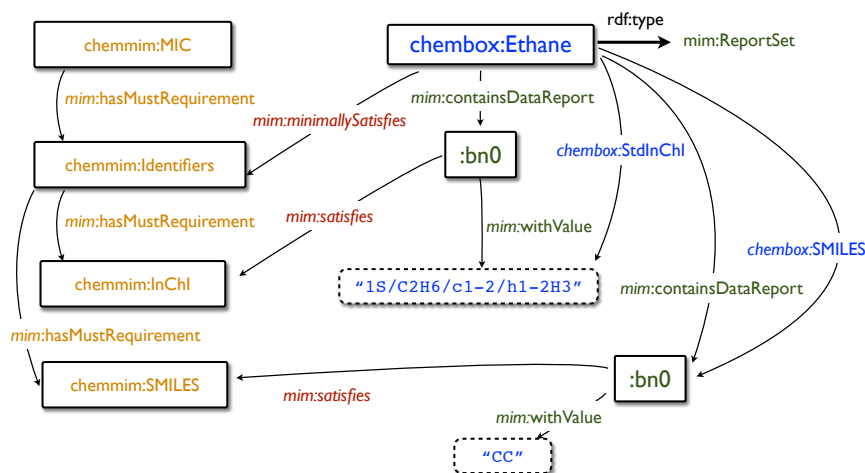


Figure 3.14: Requirement Satisfaction.

For Adequately satisfies, condition 3 applies only to Must and Should requirements. For Minimally satisfies, condition 3 applies only to Must requirements. The incentive for defining requirement sets is that upon validation we gain a MIM completeness assessment (minimally, adequately, or maximally) for the requirement sets, as well as the checklist as a whole.

If the `ReportSet` meets the conditions for minimally satisfies for example, we use the corresponding term `mim:minimallySatisfies` to indicate this (shown in Figure 3.14).

3.6 Implementation

In this section we describe the implementation of our MIC-based assessment approach. We first describe the implementation and results from our case-study. We then go on to discuss the iterative process of developing our assessment and Linked Data extraction using the Quality Knowledge Engineering Life-cycle.

To support our case study we have developed a prototype implementation of a MIM-based IQ assessment framework. The prototype has been implemented as a Java-based Web Service and provides two key functionalities:

- **Quality Evidence Alignment:** Automatically identifying reports in RDF data. This is done by using `mim:Rules` associated with the checklist to annotate and align data with the checklist using the reporting features of the vocabulary such as `mim:Reports` etc.

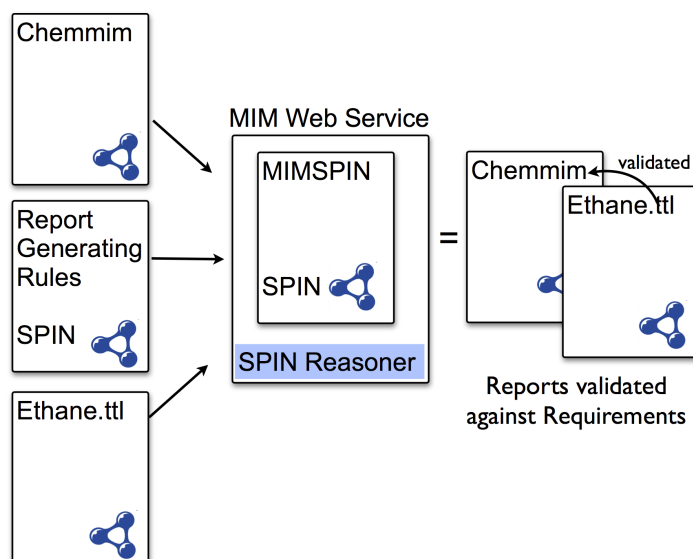


Figure 3.15: Implementation of the MIM Web Service

- **Orchestration** Validating data that is aligned with a checklist. This is done by executing rules that implement the semantics of checklist satisfaction. The data is then annotated to describe how well the data satisfies the checklist using the satisfaction features of the vocabulary such as `mim:minimallySatisfies`.

To implement both the report-generating rules and the rules governing our MIM satisfaction semantics we have used the SPARQL Inferencing Notation (SPIN) [Knulla]. SPIN is a standard in submission for representing SPARQL rules and constraints over RDF data. The standard defines a serialization that allows the definition and publication of SPARQL queries as RDF. These queries can then be applied to RDF data in two ways:

1. As rules to make inferences using SPARQL CONSTRUCT queries.
2. Perform integrity constraint checking using SPARQL ASK queries.

Therefore, it provides the exact mechanism that we need to encode our report generating rules and MIC satisfaction semantics. Tooling already built around SPIN such as the TopBraid composer [Top] makes it easy to support the definition of report-generating rules. Using SPIN we can specify a rule for example that encodes the SPARQL query shown in Figure 3.12 to create and align report of a `chemmim:InChI`.

Publishing these rules as RDF contributes to the pay-as-you-go approach of the Web of Data, meaning that others can reuse or extend them. This is also the case for the implementation of our MIM satisfaction semantics which are a collection of SPIN rules available on the web at <http://purl.org/net/mim/mimspin>.

The MIM Web service builds upon the TOPSPIN API [Knu11b], an Apache Jena based implementation of a reasoner that can process SPIN rules. Figure 3.15 details the process of calling the web service to validate a set of data against a checklist. Calling the web service requires three arguments:

1. A URI for the checklist defined using the MIM vocabulary.
2. A URI for the RDF data to be evaluated.
3. A URI for a set of SPIN encoded report-generating rules for that dataset.

The Web service hands these to the SPIN reasoner which executes the supplied rules over the data and returns the newly inferred triples, aligning and validating the data with the checklist. In the next section, we describe the implementation of the checklist satisfaction semantics using SPIN rules.

3.6.1 Checklist Satisfaction using SPIN

The SPIN modelling language provides a convenient mechanism for us to encode and publish the semantics of checklist satisfaction using SPARQL rules. We have implemented checklist satisfaction using 19 SPIN rules. The rules are encoded in a SPIN-based ontology called *mimspin* (the full set of *mimspin* rules is available in Appendix D). The purpose of the rules can be divided into two concerns, (1) rules for constraint checking, and (2) rules for constructing triples to described satisfaction.

In *mimspin* there are 13 rules for constraint checking and 6 rules for constructing triples. Figure 3.16 illustrates the hierarchy of the rules for constructing triples in the *mimspin* ontology. They have been implemented as subclasses `spin:ConstructTemplate` which make use of SPARQL CONSTRUCT queries. Figure 3.17 shows the hierarchy of the constraint checking rules. Constraint checks have been implemented as sub-classes of `spin:Functions` which make use of SPARQL ASK queries.



Figure 3.16: SPIN Rules in the mimspin Ontology for Constructing Triples.

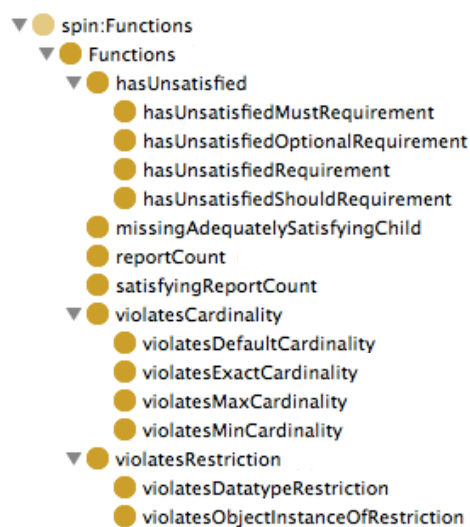


Figure 3.17: SPIN Rules in the mimspin Ontology for Constraint Checking.

The constraint checking rules are used by the construct rules to determine when to introduce triples describing requirement and requirement set satisfaction.

Listing 3.11 for example shows the SPARQL CONSTRUCT query defined by `mimspin:DataRequirementSatisfaction`.

```

1 CONSTRUCT {
2   ?z mim:satisfies ?y .
3 }
4 WHERE {
5   ?y a mim:DataRequirement .
6   ?z mim:reports ?y .
7   ?z mim:withValue ?v .
8   BIND (mimspin:violatesDatatypeRestriction(?y, ?v) AS ?result) .
9   FILTER (!?result) .

```



```
10 }
```

Listing 3.11: `mimspin:DataRequirementSatisfaction`

The query constructs a `mim:satisfies` triple between two URIs `?z` and `?y` if the conditions in the `WHERE` clause are met. The `BIND` function on line 8 makes use of the function `mim:violatesDatatypeRestriction` to check that the report `?z` meets any data type restriction defined by the requirement `?y`.

The corresponding SPARQL ASK query for `mimspin:ViolatesDatatypeRestriction` is shown in listing 3.12.

```
1 ASK WHERE {  
2   ?arg1 mim:hasRestriction ?r .  
3   ?r mim:type ?t .  
4   BIND ((datatype(?arg2) != ?t) AS ?result) .  
5   FILTER (?result) .  
6 }
```

Listing 3.12: `mimspin:violates DatatypeRestriction`

To generate the inferences the SPIN reasoner iterates over the entire collection of rules. The reasoner continues to iterate executing each SPARQL query until no additional triples are generated.

3.7 Evaluating chembox Data with Chemmim

In this section we present the results of our case-study assessing a Linked Data extraction of the data available from the WikiProject Chemicals pages on Wikipedia.

Data Preparation

DBpedia is the widely used Linked Data extraction of the data provided in Wikipedia’s Infoboxes. We have chosen to perform our own extraction of the chembox data for two reasons. Firstly, the current DBpedia extraction of chembox data does not sufficiently reflect the chemical data available in Wikipedia and as such is insufficient to test our approach. Secondly, by performing our own extraction we gain insight into how the process of generating and publishing Linked Data impacts the engineering aspect of the IQ Life-cycle.

We developed a tool to extract data from Wikipedia’s Infoboxes, building upon a Java-based MediaWiki API, the Java Wikipedia Library [ZMG08]. The Java Wikipedia API parses Wikipedia templates and provides a property value

pair for each value declared in a template. To process each chemistry page we specified the chembox template to be parsed. We created our initial RDF extraction by taking a naive approach using two steps:

1. Create a URI to represent the page e.g. `http://purl.org/net/chembox/<articleName>`.
2. For each property value pair parsed, create a corresponding triple, using the parsed property to create an RDF property and the parsed value as the RDF value.

For example the melting point value for Ethane:

```
1 | MeltingPtK = 90.4
```

Produces the following RDF triple:

```
1 <http://purl.org/net/chembox/Ethane>  
2   <http://purl.org/net/chembox/Ethane/MeltingPtK> "90.4" .
```

From this initial naive extraction we iteratively improve the RDF values created for example encoding the image value as a URI and declaring it as the type `foaf:Image`:

```
1 <http://purl.org/net/chembox/Ethane>  
2   <http://purl.org/net/chembox/ImageFileL1>  
3     <http://en.m.wikipedia.org/wiki/File:Ethane-A-3D-balls.png> .  
4 <http://en.m.wikipedia.org/wiki/File:Ethane-A-3D-balls.png>  
5   a    <http://xmlns.com/foaf/0.1/Image> .
```

Using this tool we extracted Linked Data representations of 7572 chemical compound pages from Wikipedia. The full Linked Data chembox extraction totals 376,282 RDF triples. Listing 3.13 shows an extract from `http://purl.org/net/chembox/Ethane.ttl` the chembox extraction for the article `http://en.wikipedia.org/w/Ethane`.

Assessment

To perform a MIC assessment over our data we have used the chemmim MIC encoded using the MIM Vocabulary discussed previously in this chapter. Along with the checklist we have also defined a set of 15 report-generating rules in the SPIN format to annotate and align reports in the chembox Linked Data with the chemmim checklist.

Using our prototype MIM assessment Web service we have assessed each of the 7572 chembox extractions against our chemmim checklist. The assessment

```
1 @prefix chembox: <http://purl.org/net/chembox/> .
2
3 chembox:Ethane
4   chembox:wikiPageUsesTemplate
5     <http://dbpedia.org/resource/Template:Chembox> ;
6   chembox:Appearance
7     "Colorless gas" ;
8   chembox:Autoignition
9     "472 C" ;
10  chembox:Beilstein
11    "1730716" ;
12  chembox:BoilingPtK
13    "184.6" ;
14  chembox:C
15    "2" ;
16  chembox:CASNo
17    "74-84-0" ;
18  chembox:ChEBI
19    "42266" ;
20  chembox:ChEMBL
21    "135626" ;
22  chembox:ChemSpiderID
23    "6084" ;
24  ...
```

Listing 3.13: chembox Linked Data Extraction for the Article Ethane (extract)

process generated a further 808,420 triples which align the data with the checklist, and detail requirement satisfaction. With this assessment data in a well-structured format we can query it and ask questions about how well chemical compound data is being reported across Wikipedia.

The graphs in Figure 3.18 and Figure 3.19 present two views over the data. Figure 3.18 shows how each individual requirement in our checklist is satisfied across the entire chembox dataset. The most well satisfied reporting requirement is Molecular Formula with 7263 (96%) of chemboxes satisfying the requirement. In contrast to this our results show that Melting Point data is particularly poorly reported across Wikipedia with only 1343 (18%) of chemboxes satisfying the requirement. This finding is further feedback and evidence to support recent activity within the online chemistry community to improve the availability of open melting point data [BLWC11].

Figure 3.19 shows how well the requirement sets; `chemmim:Identifiers`, `chemmim:Properties` and the MIC itself `chemmim:MIC` are satisfied. Currently 64% (4863) of chemboxes *minimally* satisfy the chemmim by satisfying all of the *Must* requirements. The number of chemboxes going beyond minimal satisfaction is significantly lower, with only 274 *adequately* (reporting all *Must* and *Should* requirements) and 168 *maximally* (reporting *all* requirements) satisfying the checklist. The previously discussed WikiProject Chemicals hand-classified

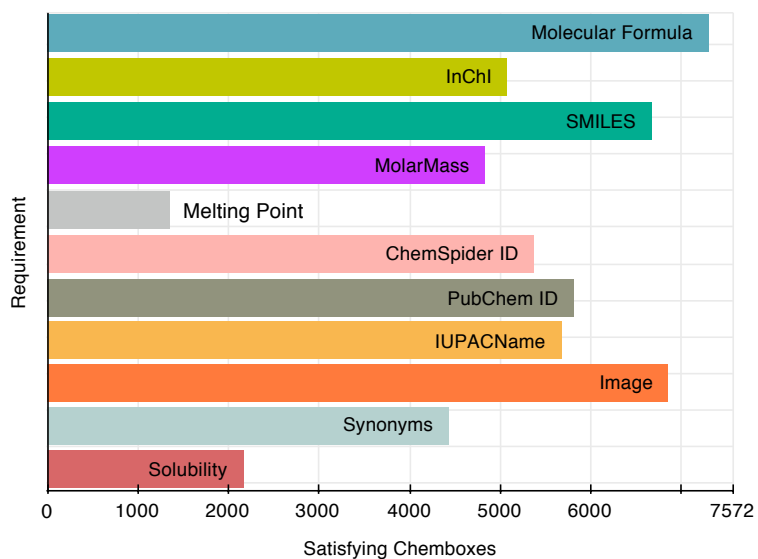


Figure 3.18: Individual Requirement Satisfaction Across All Chembox Instances.

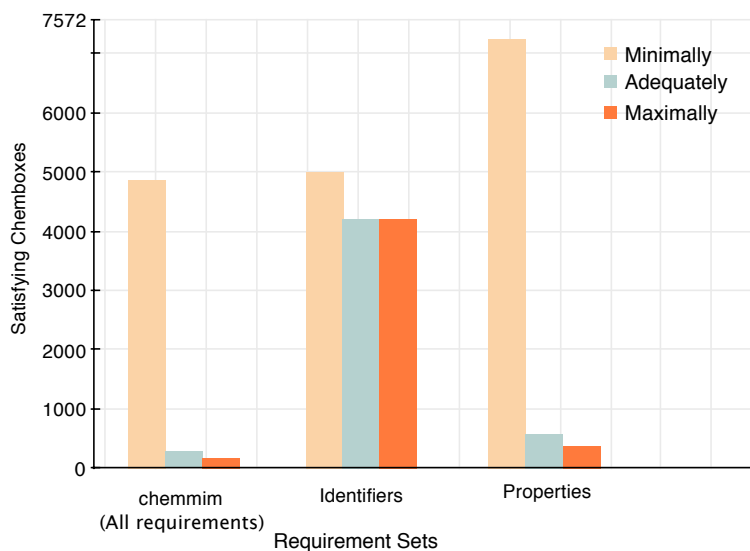


Figure 3.19: Requirement Set Satisfaction Across All Chembox Instances.

assessment can be seen to be in agreement with this finding. Though the remit of their assessments differs, their results suggest a similar discrepancy between the number of top quality and minimal quality articles.

By using conceptually grouped requirement sets we can examine how well the chembox data satisfies not only the whole checklist, but particular types of chemical compound data such as chemical identifiers, or chemical properties. Our results for example show that whilst Wikipedia may be a poor source for the particular chemical properties we have defined, it is a relatively good source for chemical identifiers, with 56% (4207) of chemboxes maximally satisfying the Identifiers requirement set. This more detailed view of the data afforded by our MIC assessment is lost in the WikiProject chemicals evaluation.

3.7.1 Iterative Assessment - The IQ Life-cycle

Reaching the final MIC assessment detailed in Figures 3.18 and 3.19 was an iterative process involving interactions between the Quality Knowledge Engineering and Linked Data extraction processes. To coordinate this process we manually derived the ground truth for 35 (approx. 0.5%) of the original chemical compound articles. For each chemical compound the ground truth detailed which requirements were reported, which requirements were not reported, and how well we expected the chembox to satisfy the checklist. By establishing this ground truth we were able to iteratively assess our:

1. Linked Data extraction.
2. Report generating rules.
3. mimspin checklist satisfaction rules.

Any anomalies in these ground truth resources therefore triggered a process of fault-finding. We discovered that anomalies primarily occurred due to one of two of reasons, *Inadequate report generating rules* and *Inadequate extraction* from the source data.

Inadequate extraction: chembox descriptions are split across multiple sub-templates as a result of its size, e.g. chembox properties, chembox hazards. Failing to declare these sub templates as part of our extraction resulted in the properties contained within them to be omitted.

Inadequate report generating rules: For a molecular formula we initially created the following report generating rule to align values with the checklist declared with the property `chembox:Formula`:

```
1 CONSTRUCT
2 { ?x mim:containsDataReport _:b0 .
3   _:b0 mim:reports chembox-mim:InChI .
4   _:b0 mim:withValue ?value . }
5 WHERE { ?x chembox:MolecularFormula ?value . }
```

There is however an alternative representation of Molecular formulas in chemboxes where each chemical element is described separately, for example the molecular formula for Ethane is describe as:

```
1 | C = 2
2 | H = 6
```

This produces the following successful extraction:

```
1 chembox:Ethane
2   chembox:C "2" ;
3 chembox:Ethane
4   chembox:H "6" ;
```

It was therefore necessary to create an additional report generating rule to align data using this alternative representation.

The iterative process of the MIC development is a useful tool in identifying the *existence* of an error in the RDF generation process or report generating rules. Finding the *source* of those errors, such as those above, however was a non-trivial task with scope for future work to support and improve this process (we discuss this in section 3.10).

3.8 Comparison with other approaches

There have been a number of developments in the validation of RDF data, some of which were highlighted in the recent RDF Validation Workshop [Pru13a]. In this section we discuss two particularly relevant solutions, *minim* [ZKGG13] and OWL Integrity Constraints [TSBM10].

3.8.1 *minim*

Beyond our implementation of the MIM framework using SPIN, the *minim* framework [ZKGG13] further validates our checklist-based approach. As part of the

Workflow 4Ever project [BCG⁺12], the minim framework was grounded on an early version of the MIM vocabulary. The development was subsequently branched off as minim to address specific requirements of the project. The resulting minim framework has taken a different conceptual approach by focusing on the type of tests that can be performed, rather than the checklist specification. As a concrete comparison consider the InChI requirement from the chemmim assessment.

In the MIM vocabulary the requirement is expressed as follows:

```
1 :InChI
2   rdf:type mim:DataRequirement ;
3   mim:hasRestriction
4     [ mim:onSelf "true"^^xsd:boolean ;
5       mim:type xsd:string ] .
```

A report for the above requirement is generated by a `mim:Rule` associate with the requirement for example, the following SPARQL query (encoded in RDF using the SPIN vocabulary):

```
1 CONSTRUCT
2 { ?x mim:containsDataReport _:b0 .
3   _:b0 mim:reports chembox-mim:InChI .
4   _:b0 mim:withValue ?value . }
5 WHERE { ?x chembox:StdInChI ?value . }
```

The same InChI requirement is expressed in minim as follows:

```
1 :InChI rdf:type minim:Requirement
2 minim:isDerivedBy
3 [ rdf:type minim:ContentMatchRequirementRule ;
4   minim:exists
5     """
6     ?x chembox:StdInChI ?value .
7     FILTER ( datatype(?value) == xsd:string )
8     """
9   minim:showpass "InChI identifier is present" ;
10  minim:showfail "No InChI identifier is present" ;
11  minim:derives :InChI
12 ] .
```

Where MIM separates concerns by defining the checklist, alignment rule, and satisfaction semantics separately, minim defines all three in the checklist requirement description. MIM focuses on a core set of common requirements expressed in most MICs to allow for the concise description of checklists, where the descriptions are *agnostic* to the data representation. Minim has instead focused on extensibility allowing for a greater range of bespoke tests to be described.

Minim has an explicit set of vocabulary terms for describing the different sets of tests that can be supported. For example minim also allows metadata

tests to be combined with other environment tests, such as a liveness test which test whether a URI is currently accessible. Anyone adopting the minim model that has new types of tests can extend the model and implement their own. A restriction however is that these tests can restrict the portability of the checklist assessment. The liveness test for example is implemented using a python script.

The current implementation of the MIM Framework retains portability by standardising its implementation to SPIN - any SPIN aware system can execute the MIM rules and assessment semantics.

3.8.2 OWL and OWL Integrity Constraints

As an approach that is based on Semantic Web technologies, the goals and features of our checklist-based framework can be seen to overlap with the Web Ontology Language (OWL) [MVH⁺04]. OWL ontologies support the definition of classes that describe the features necessary for an individual data item to be a member of that class. Class descriptions are therefore analogous to the description of requirements in our checklist. OWL also has an RDF serialisation and extends RDF semantics to operate over RDF data.

Using Manchester Syntax [HDG⁺06] we can express our InChI requirement in OWL as follows:

```
1 Class: InChIRequirement
2 SubClassOf: chembox:StdInChI some value .
```

Listing 3.14: InChI Requirement Described using OWL

This class description defines an InChI as individuals that provide a value for the property `chembox:StdInChI`. If we consider the class itself as the requirement, we can consider any individual that is a member of the class a report of that requirement. However, the current OWL 2 RDF semantics adopt two features that are incompatible with our quality assessment.

The Open World Assumption (OWA). If an InChI were to be defined without a corresponding InChIValue, this would not be highlighted as an error by an OWL reasoner. Instead the OWA results in the inference that there exists an InChIValue, but that it is not explicitly defined. This directly conflicts with our need for an existence check.

No Unique Names Assumption (UNA). We can extend the requirement in listing 3.14 to include a cardinality restriction to say that there must be one and only one InChIValue.


```
1 Class: InChIRequirement
2 SubClassOf: chembox:StdInChI exactly 1 value .
```

Listing 3.15: InChI Requirement with Cardinality Constraint Described using OWL

The presence of two different InChI values would not however raise an error. Instead the assumption would be made that the two InChIValues are in fact the same. This directly conflicts with our need for cardinality checks.

An alternative to the traditional OWL 2 Semantics are Integrity Constraint Semantics (ICs) [PUSC12]. ICs are a semantics for OWL that employ a Closed World Assumption (CWA) as well as a form of the UNA. These semantics allow OWL classes to be interpreted as constraints. The Stardog RDF database [Sta] for example currently provides an implementation of OWL with ICs. A practical implementation of ICs is achieved by transforming the OWL class restrictions into to SPARQL queries. Each axiom in an OWL IC Ontology is transformed into a corresponding SPARQL query. For example, our above InChI requirement class description interpreted as an IC would produce the following SPARQL query:

```
2
3 ASK WHERE {
4   ?x rdf:type :InChI .
5   OPTIONAL {
6     ?x chembox:StdInChI ?y .
7     ?y rdf:type :InChIValue .
8   }
9   FILTER ( !bound( ?y ) )
10 }
```

Listing 3.16: InChI Requirement Translated to SPARQL-based Integrity Constraint

The FILTER declared on line 8 of the SPARQL query acts as the required existence check. This means that if a value for chembox:StdInChI is not found then the ASK query will return false, meaning that the requirement is not satisfied. The implementation of ICs as SPARQL queries implies that any IC that is subsequently converted into SPARQL could be directly encoded as SPARQL in SPIN. By supporting a SPARQL based approach for requirement description, this suggests MIM achieves a similar expressiveness as an approach based upon OWL ICs.

A purely OWL ICs based approach presents a number of restrictions with respect to what can be expressed in our checklist requirements:

- Inability to express different requirement levels such as Must, Should, Optional.
- Unable to express rules that validate data present in data literals. This has previously been highlighted as a restriction of an OWL based approach to data validation in the Life Sciences [BGtUC12].

In contrast to SPIN, OWL has significant tooling, expertise, and existing ontologies available. One clear benefit of using OWL with ICs is therefore the potential to reuse the significant corpus of existing OWL ontologies with minimal effort. In the current IC specification, importing an OWL ontology using the term `ic:imports` as apposed to `owl:import` in effect ‘flicks the switch’ to interpret the class descriptions contained within the imported ontology as integrity constraints.

3.9 Discussion

We believe that with this work we are the first to address the challenge of assessing data published “in the wild” for MIC compliance. In the process we have shown the value in bringing objective standards to the Web of Data as reusable quality components. The objective nature of the Quality Knowledge means that we achieve an assessment that is broadly relevant. The ability to perform a large-scale standards-based assessment of scientific Linked Data provides IQ feedback not just for users, but each of the stakeholders involved: data consumers, data providers and the standards creators themselves.

Data consumers are presented with the opportunity to base their source selection on which source better satisfies the MIC requirements they are interested in, for example Wikipedia and chemical compound identifiers. For the maintainers of community data resources such as WikiProject chemicals, a large scale MIC assessment provides feedback that can be used to suggest where their strengths lie, and where efforts would be best placed to improve the quality of a resource, for example the improvement of melting point data in Wikipedia. A large-scale analysis can also provide feedback to the developers of checklists. The development of a MIC is a difficult process. Checklists creators aim to fulfil two criteria when developing a checklist: *sufficiency* and *practicability* [OALB⁺11] i.e. not be so burdensome as to prohibit use. Performing a MIC analysis provides feedback to the creators about where the checklists may be falling short of these criteria. If

there is some data that are considered vital for reporting, but are often omitted, this might suggest that there are issues relating to the gathering, or publication of this information.

As a further example of the potential of the approach consider the work of Hogan et al. [HUH⁺12] (discussed in Chapter 2), evaluating conformance to Linked Data principles. By combining the strengths of MIM and minim (proposed in future work) we could develop a checklist to evaluate the conformance of Linked Data to the principles. Encoding the checklist as a reusable quality component, it can be shared and used by anyone to evaluate their own, or other datasets for conformance.

A further finding of this work is the establishing of an interaction between the IQ life-cycle and the Linked Data extraction process. We have shown the potential of this interaction to support the extraction of high quality Linked Data representations, and in turn support the development of quality components. Our use of a ground truth to evaluate the success of our extraction continues and establishes further a precedent set by the SILK Linked Data integration Framework [VBGK09]. The authors create a small reference set of links for integration that act as a ground truth, against which they benchmark their general link specifications.

For the Web of Data, this interaction also highlights that whilst we gain an indication of the completeness and quality of an original data source, it is the Linked Data extraction itself for which we are making the true MIC assessment.

We have also identified that a successful separation of concerns for Quality Knowledge Encoding and Quality Evidence Alignment in an IQ solution means that we are better able to manage the heterogenous nature of the Web of Data, supporting varying representations of data both within, and across datasets. We believe this separation of concerns will be a key feature of successful IQ solutions in the Web of Data.

3.10 Future Work for MIM

In future work, we look to further validate our MIM vocabulary by encoding a broader range of checklists. Our first opportunity is through a collaboration with the Health e-Research Centre (HERC), managing metadata in Health care datasets as part of the CHIPSET Epidemiology Toolkit to effectively share data

and methodology [CHI].

At the modelling level we are also in the process of aligning the MIM model with the minim model, as well as looking to consolidate a number of related solutions that have recently emerged in for checklist-based IQ assessment. These include:

- The Open PHACTS Validator [HWZW13] which extends OWL using annotations to describe Must, Should and Optional constraints, and evaluates them in a custom Javascript application.
- Component Profiles for the Scalable Preservation Environments (SCAPE) project⁵ which use a bespoke XML schema to define the minimum requirements to describe reusable scientific Workflow components.

The semantics of MIC satisfaction are currently encoded in a series of SPIN rules. To ensure that they are functionally correct, and to better support alternative implementations useful future work would be to define a formal specification of the semantics of satisfaction. Given our current SPARQL-based implementation, we could build upon the compositional semantics of SPARQL presented in [PAG06] as an initial definition.

In our current application of the MIM framework we have used it to evaluate the quality of a single data resource. Working with the Web of Data we are not necessarily constrained to one data source in order to satisfy a MIC. Rather than evaluating the quality of a data resource the goal of a user might instead be to gather data from multiple resources in order to satisfy a MIC. This is particularly relevant to Life Science data, where related components of a study may be scattered across disparate resources [SRSF⁺12]. Our approach as it stands may make use of multiple datasets by either addressing different datasets in different alignment rules, or by making use of federated SPARQL queries that can address multiple datasets in one query. Incorporating data from multiple data sources raises the issues of provenance and trust.

We see the incorporation of detailed provenance of the report generation and MIC validation processes as a valuable avenue for future work for a number of reasons. *Provenance and versioning* - when revisiting data that claims to satisfy a MIC we wish to understand how it was determined that the data is compliant. *Provenance to aid fault-finding* - having detailed lineage about how reports were

⁵<http://www.scape-project.eu>

generated can aid in the fault-finding process when developing a Linked Data extraction and report-generating rules. *Provenance for attribution* - if one were to subsequently use data aligned with a checklist in a further study, detailed provenance about where and when that MIC assessment was performed is crucial to give correct attribution. To prove the origin of the MIM assessment for these provenance-based extensions we could introduce a mechanism to digitally sign the resulting RDF graph. To generate a signature, we could follow existing principles set by previous work using named-graphs to capture provenance [CBHS05]. This presents an opportunity to establish authoritative services who validate and sign data as proof of MIC conformance.

3.11 Chapter Summary

In this chapter, we have presented our work in exploring objective Quality Knowledge in the Web of Data. We have described an approach to bringing Minimum Information Checklists to the Web of Linked Data as reusable quality components. We have presented the MIM Vocabulary, a meta-modelling vocabulary suitable for encoding MICs as RDF, that was informed by an analysis of many of the existing checklists used by the Life Sciences community. We have also presented the MIM framework, a prototype implementation that orchestrates alignment and evaluation of MICs encoded in the vocabulary. We have demonstrated the application of the vocabulary and framework with a case-study evaluating a Linked Data extraction of chemistry data available in Wikipedia. With our case-study we have highlighted the broadly relevant and valuable feedback that can be obtained from an objective assessment, as well as establishing an interaction between the IQ Life-cycle and the Linked Data extraction process.

In the next chapter we move on to the predictive aspect of Quality Knowledge and describe a modelling approach based on Bayesian Networks.

Chapter 4

Quality Fragments: a Probabilistic Approach

“*The theory of probabilities is at bottom nothing but common sense reduced to calculus; it enables us to appreciate with exactness that which accurate minds feel with a sort of instinct for which oftentimes they are unable to account.*”

- *Pierre Simon Laplace*

4.1 Chapter Introduction

This chapter introduces the second study to be drawn from our OPS IQ classification. Our previous study focused on the task of supporting an objective quality assessment by implementing Minimum Information Checklists as a Quality Standard. In the following two chapters, we present a more general approach to modelling the predictive aspects of Quality Knowledge.

There are broadly three elements to our approach: (1) An encoding of Quality Fragments using Bayesian Networks, (2) a novel procedure to build Quality Fragments automatically using provenance, and (3) Evident, a prototype framework that supports the orchestration of Quality Fragments in the Web of Data. In this chapter we focus primarily on the first of these.

We approach the problem of predictive Quality Knowledge by proposing a novel approach to modelling IQ metrics using *template-based* Bayesian Networks.

With this approach we have three clear goals 1) to establish whether we can successfully model predictive Quality Knowledge in the Web of Data using Bayesian Networks, 2) to establish if a Bayesian Network-based encoding of a metric can successfully reproduce an existing encoding and 3) to understand how well the predictive elements of Bayesian Networks improve our ability to judge IQ in the face of incomplete metadata. We expect that given a sufficiently accurate probabilistic representation of Quality Knowledge, we should be able to provide an assessment that can be used to make IQ based decisions.

We begin the chapter with a general introduction to Bayesian Networks and motivate their suitability to our Quality Knowledge modelling problem. We then introduce Multi-Entity Bayesian Networks (MEBNs), the particular type that we have chosen to adopt. We go on to propose an application of the MEBN template-based approach for encoding Quality Fragments. Finally we show that this approach is suitable with an evaluation using an encoding of the GAQ metric to assess data from the Bio2RDF project. We also present Evident a prototype implementation suitable for the Orchestration of our assessment.

The work in this chapter builds upon our previous publication describing a Bayesian Network based approach to modelling Quality Knowledge [GG11a]: Matthew Gamble, Carole Goble. *Quality, Trust, and Utility of Scientific Data on the Web: Towards a Joint Model*. Proceedings of the International Conference on Web Science 2011 (WebSci11).

4.2 Modelling Predictive Quality Knowledge

In Chapter 2, we established that a significant aspect of the Quality Knowledge employed by scientists is predictive in its nature. This predictive aspect takes evidence, in the form of metadata and provenance data, and makes predictions about likely quality in some dimension of IQ. Because of this predictive nature there is an inherent level of uncertainty involved. This uncertainty is in contrast to an objective assessment such as the MIC assessment we have just presented. With a MIC assessment we can say with certainty whether some data does, or does not meet the checklist's requirements. Instead predictive assessments are a likelihood assessment i.e. given some evidence what is the likelihood that this data meets some requirement in a particular dimension of IQ.

Take as a concrete example the GAQ metric introduced in chapter 1. The

```

1 bio2rdf_uniprot:Q8IZF5
2     a                                uniprot_core:Protein ;
3     goa_vocabulary:go-annotation    goa_resource:GDB_74 ,
4                                     goa_resource:GDB_75 .
5 goa_resource:GDB_74 a                goa_vocabulary:GO-Annotation ;
6     goa_vocabulary:evidence         bio2rdf_eco:0000304 ;
7     goa_vocabulary:go-term          bio2rdf_go:0004930 ;
8     sn_goa:ec_label                 "IDA" ;
9     sn_goa:go-depth                 "10" .
12
12 goa_resource:GDB_75 a                goa_vocabulary:GO-Annotation ;
13     goa_vocabulary:evidence         bio2rdf_eco:0000304 ;
14     goa_vocabulary:go-term          bio2rdf_go:0016021 ;
15     sn_goa:ec_label                 "TAS"^^xsd:string ;
16     sn_goa:go-depth                 "5"^^xsd:integer .

```

Listing 4.1: Bio2RDF Gene Ontology Annotations Data (Extract).

GAQ score is an attempt to predict the likely quality of a gene annotation. The GAQ score uses metadata from gene associations data about the GO term depth and evidence code, to predict its likely quality. This makes use of prior knowledge about how the metadata relates to quality. Prior knowledge can come directly from a domain expert's intuitive understanding of the data, or can be derived from some empirical study that reveals some further understanding of the data. Armed with predictive Quality Knowledge in a reusable mechanism, users can make predictions about the likely quality of new and unseen data.

For a general approach to modelling predictive Quality Knowledge in the Web of Data, we require a formalism that can represent metadata in a domain, such as gene annotations data, and some prior knowledge about our understanding of that metadata, and how it relates to quality.

Consider as an example of metadata in a domain the gene annotations data. The Bio2RDF project publishes Linked Data versions of a series of high-profile Life Sciences datasets into the Web of Data, including the GOA database.

Listing 4.1 details an example of data extracted from the Bio2RDF's GOA data¹. The listing describes an example gene product `bio2rdf_uniprot:Q8IZF5` with two annotations `goa_resource:GDB_74`, `goa_resource:GDB_75`. The annotation describes the evidence code metadata using the property `goa_vocabulary:evidence` the GO term metadata using the property `goa_vocabulary:go-term`, and the depth of that GO term using `sn_goa:go-depth`.

Specifically for the GAQ metric, the metadata we want to model is the depth of an annotation `sn_goa:go-depth`, the evidence code `goa_vocabulary:evidence`,

¹We describe in detail how this data was generated in section 4.6.1

and how they inform the likely the quality of the annotation.

There are a number of approaches in the literature that address the problem of modelling and reasoning about uncertainty, including Bayesian Networks [JN07], Subjective logics [Jøs97], the Dempster-Shafer theory [Sha76], and Certainty Factors [Ada84]. We have chosen Bayesian Networks, a popular and well understood approach to encode predictive Quality Knowledge. To achieve this in the Web of Data we use PR-OWL2 [CLC13], an existing mechanism for encoding Bayesian Networks, built upon Semantic Web technologies.

Our aim is to support the engineer in modelling predictive Quality Knowledge and making it available as reusable quality components. We want to achieve this in a way that is robust in the face of inconsistent data and metadata. We also aim to follow the pay-as-you-go approach that is prevalent in the Web of Data. The objective is to build a corpus of Quality Fragments that can be published into the Web of Data and subsequently combined, compared, and reused by engineers and users.

Bayesian Networks are a common mechanism for representing domain knowledge about uncertainty, and have a history of application in several scientific domains such as genetics [FLNP00], and medical diagnosis [Nik00]. They are grounded in classical probability theory and are well supported with respect to tooling and implementations, including several emerging approaches in the Semantic Web [DPP06] [CLC13] [YC05]. An important consideration to make with regards to engineering is the ease of encoding domain knowledge. Typically networks are built by directly encoding an expert's domain knowledge, learnt through data, or a combination of both [KF09]. This is directly mirrored by our two primary sources of predictive Quality Knowledge.

In the next section we provide a brief introduction to Bayesian Networks and demonstrate their suitability to modelling Quality Knowledge.

4.3 Bayesian Networks

Bayesian Networks are a particular type of Probabilistic Graphical Model (PGM)², a broader class of techniques for modelling uncertainty. They provide a declarative representation for modelling uncertainty as *relationships* between connected sets of *variables*, where variables correspond to observable states in a particular domain. These representations can then be used to compute probabilistic beliefs about those variables. Crucially, as a declarative representation, Bayesian Networks separate out the knowledge representation task from that of reasoning. As a result the engineer is left to model the domain, free from any concern about how to use that data to subsequently reason about its uncertainty.

More formally a Bayesian Network is a pair $\mathcal{B} = (G, P)$ where G is directed acyclic graph $G = \{V, E\}$, and P a set of probability distributions associated with each of G 's nodes. The nodes (V) in the graph represents a set of variables $\mathcal{X} = [X_1, \dots, X_n]$, and the edges (E) correspond to influence from one variable to another. Each variable X_i represents a *discrete* or *continuous* series of states that the variable can take. In addition to the graph structure, for each random variable X_i there is also a corresponding Conditional Probability Distribution (CPD) in P that defines the likelihood of each state occurring. The likelihood of a state for a node X_i is defined in terms of the possible states of the node's parents $Pa_{X_i}^G$, in the graph G . This can be modelled in the form of a continuous function for continuous variables, or a Conditional Probability Table (CPT) for discrete variables.

To demonstrate how Bayesian Networks capture the relationships between variables consider the simple example in Figure 4.1. This example demonstrates the relationship between two variables. The example consists of two nodes, *chemmim* and *chemboxquality*, and one directed edge from *chemmim* to *chemboxquality*. *chemmim* represents the result of a chemmim assessment. *chemboxquality* represents our belief in the overall quality of that chembox. The directed edge reflects the intuitive belief that the overall quality of the chembox is influenced by the result of its chemmim assessment.

Both nodes model discrete states; *chemmim* models the potential results of a

²For a thorough overview of Probabilistic Graphical Models, including Bayesian Networks, the reader is recommended the text by Koller and Friedman [KF09] which provides an authoritative introduction. The notation used within this thesis is consistent with the referenced text.

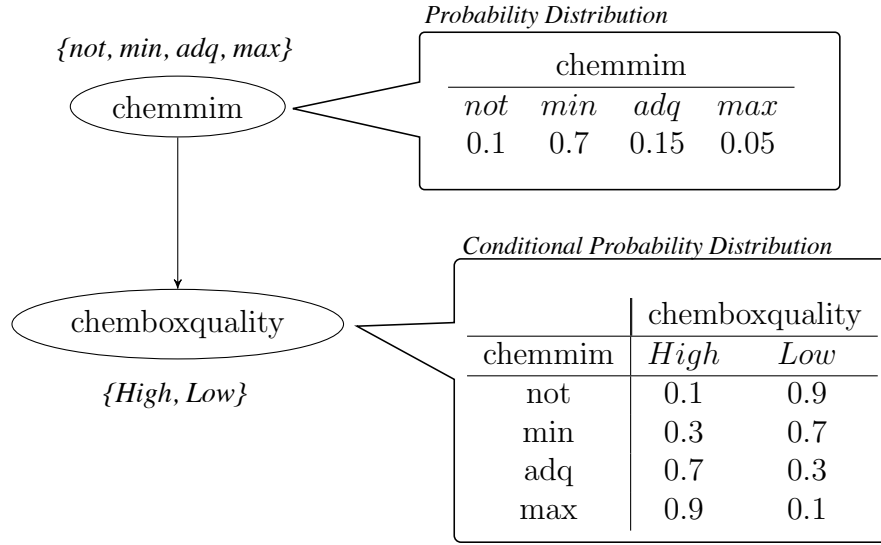


Figure 4.1: Bayesian Network for chembox Quality.

chemmim assessment [*not, min, adq, max*] and *chemboxquality* is represented by two discrete states [*High, Low*].

The CPT for *chemboxquality* represents our belief in the likelihood that the chembox quality will be high or low *given* the state of the chemmim assessment. Each entry in the table assigns a probability to the likelihood of the particular combination of states occurring. For example the likelihood of the chembox quality being *high* given a chemmim assessment of *min* is 0.3, or 30%. For the purposes of this illustration we have modelled this intuitively, assigning probabilities to each entry³.

The CPT for *chemmim* is more straightforward. There are no parents on which *chemmim* is dependent, therefore the CPT simply captures the prior probability of each state occurring. We are however less able to intuitively fill the entries in the probability distribution. Instead we can demonstrate the data driven aspect of Bayesian Network modelling and take the data from the results of our chemmim evaluation from the previous chapter. These results gives us the occurrence of each chemmim result in our dataset (c.f. the graph in Figure 3.19), and therefore the prior likelihood of each occurring.

To apply the network we use the probability distributions we have created to compute the likely states for each variable. That is, we wish to compute

³As with any probability distribution the individual likelihood estimates for a state must sum to 1.

$P(\text{chemmim}, \text{chemboxquality})$. The Bayesian Network as a whole represents a *joint probability distribution* $P(X_1, \dots, X_n)$ over all of the variables in the network. This joint distribution is defined in terms of a series of *factors* elicited from the network structure. These factors correspond to the CPDs for each node in the graph. Our example therefore defines two factors $P(\text{chemmim})$ and $P(\text{chemboxquality}|\text{chemmim})$. The result of the joint distribution is computed as a product of the two factors, that is:

$$P(\text{chemboxquality}, \text{chemmim}) = P(\text{chemboxquality}|\text{chemmim})P(\text{chemmim})$$

We can then use this joint distribution to evaluate the likelihood of a combination of events, for example the likelihood of the chembox quality being *high* and the chemmim assessment outcome being *min*:

$$\begin{aligned} &P(\text{chemboxquality}_{\text{High}}, \text{chemmim}_{\text{min}}) \\ &= P(\text{chemboxquality}_{\text{High}}|\text{chemmim}_{\text{min}})P(\text{chemmim}_{\text{min}}) \\ &= 0.3 \times 0.7 \\ &= 0.21 \end{aligned}$$

More generally for any network this factor product is computed via the *chain rule*:

$$P(X_1 \dots X_n) = \prod_{i=1}^n P(X_i | Pa_{X_i}^G) \quad (4.1)$$

The Chain Rule

It is the chain rule that allow us to compute the more complex joint probability over all the variables in our domain as a product of a series of simpler probability distributions. Due to this grounding in Bayesian probability, Bayesian Networks are a powerful tool for managing complexity. They allow us to take advantage of independence assumptions and model evidence as independent, so that we don't have to determine the relationships between all pieces of evidence. This is because a key property of their representation is that unless two variables are linked by an edge, we are not required to explicitly define any quantitative relationship between the two variables. Therefore, we are only required to describe the relationships

that we know about, or can measure. However if we have any path between two variables (via other nodes in the network) then we have implicitly defined a quantitative relationship. The reasoning potential of Bayesian networks therefore comes from their structure and corresponding CPDs.

The type of query that we wish to compute over a Bayesian Network is a *conditional probability query*. This is where we have a set of Evidence ($E = e$), and a query about a subset of variables $Y \subset X$. From this we wish to compute $P(Y = y|E = e)$ i.e. what is the likelihood of a state for the variables in Y *given* our evidence set. This is the case when we have evidence about some of the variables, and know their states. We therefore want to update our belief in the other random variables based upon this evidence. For example if we *know* the outcome of the chemmim assessment is *min* then our query is updated to:

$$\begin{aligned} &P(\text{chemmim}_{\text{min}}|\text{chemboxquality} = \text{High}) \\ &= P(\text{chemmim}_{\text{min}}|\text{chemboxquality}_{\text{High}}) \\ &= 0.3 \end{aligned}$$

This is done by eliminating all possibilities from our joint distribution that are incompatible with the observed evidence. The task of computing this conditional query can be shown to be NP-hard in the worst case, however the general case is often much better [KF09] .

TIMTOWTDI - There is more than one way to do it.

It is a general property of Bayesian Networks that there is often more than one way to represent the same underlying set of independence relations [KF09]. Even in our simple chemmim example we might reverse the direction of the edge. This models the result of a chemmim assessment as conditional on the overall quality of a chembox. This relationship intuitively makes sense, though it might be more difficult to populate the CPDs for this representation.

We can also choose to model random variables as continuous or discrete values. In the next section we briefly address this modelling decision, discussing it in the broader context of modelling quality and trustworthiness.

4.3.1 Modelling the Variables of a Metric: Continuous vs. Discrete

In the chemmim example we have used discrete states to model the two random variables. In our previous study [GG11a] we also restricted the modelling of our metrics to discrete states, in part due to a limitation of the underlying modelling framework. We are not however restricted to discrete representations of states in Bayesian Networks.

Quality and trust metrics also make use of both discrete and continuous representations. A number of aspects are considered when choosing a representation for a metric:

- *Computational Complexity* - What is the computational overhead of computing with the representation [Mar94].
- *Features and Expressiveness* - is the representation able to capture the complexity of quality or trustworthiness in the required context.
- *Understandability* - In the case where the metric needs to be considered by a user, is the metric intuitive and can the users interpret the metric [DAKPA11]?

The use of a trust continuum to represent and weight trust is common to many trust models and applications [Gol06]. In one of the earliest computational models of trust, Marsh [Mar94] defines a scale $[-1...1]$ incorporating the notion of distrust as well as trust. Subsequent approaches typically adopt a simpler scale from 0 to 1. We note that this is a convenient scale to adopt for a probabilistic representation. To further incorporate uncertainty some metrics chose a probability distribution such as the beta distribution [MZdS⁺06], where the mean of the distribution is the trust value and the confidence in that estimate is represented by the variance.

The potential over-simplification of trust as a value on a continuous scale is acknowledged. Some suggest that humans are better able to rate trust as discrete verbal statements [JIB07]. Marsh suggests that the simplicity and subsequent computational tractability outweighs the disadvantages. Conveniently Bayesian Networks are capable of making use of both discrete and continuous values. In the next chapter when we come to automating the manipulation of Quality Fragments, we restrict the result to continuous values for computational tractability.

4.3.2 Modelling the GAQ Metric

In this section we use the GAQ metric to demonstrate the ability of Bayesian Networks to model more complex metrics that use continuous and discrete variables. Bayesian Networks that make use of both types of variable are referred to as *hybrid* Bayesian Networks. For the data-driven aspects of the modelling we have used version 125 of the human sub-set of gene annotations from the GOA database, *gene_association.goa_human.125*, downloaded from the GOA ftp archive [GOF]. This subset contains a total of 383,912 annotations.

The GAQ score captures two distinct elements of Quality Knowledge: (1) the intuitive knowledge that an annotation using an ontological term from a deeper part of the ontology indicates a more specific term, and (2) that the method by which the annotation was produced will impact its likely quality.

Formally the GAQ score for an annotation a is the product of its depth in the ontology $Depth$ and the Evidence Code Rank (ECR):

$$GAQ(a) = ECR_a \times Depth_a \quad (4.2)$$

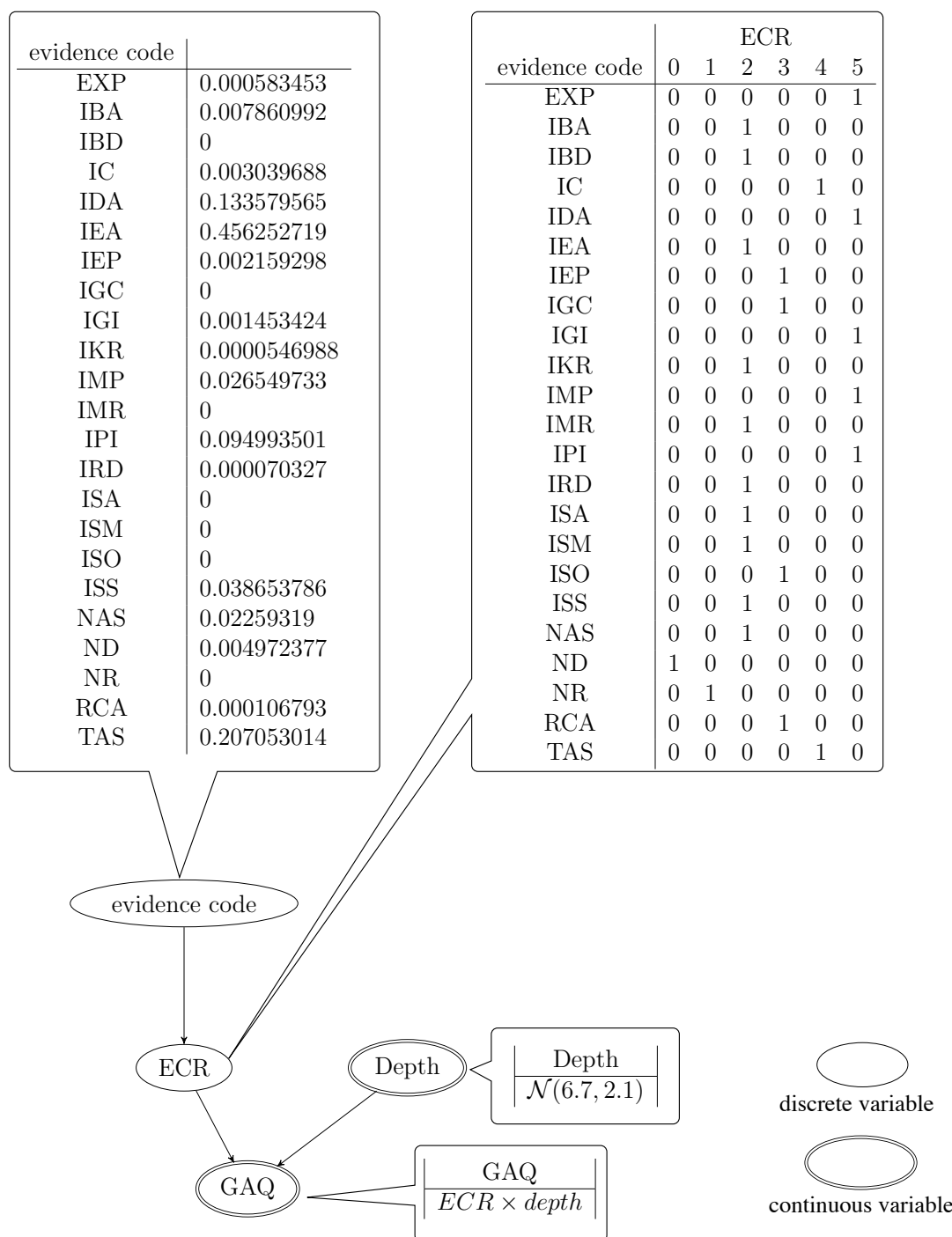
GAQ Score

The ECR is a numerical scoring from 0 to 5 of the gene annotations evidence code, which represents the process from which the annotation was created. Figure 4.2 presents one possible Bayesian Network representation of the GAQ score.

The network is constructed from both continuous and discrete valued variables. The states of the *evidence code* variable are represented as a discrete set of evidence codes used by gene annotations. The *evidence code* variable does not have any parents so the probability distribution represents our expectation that each of the processes will be used. The data for the CPT was determined systematically by analysing the frequency of evidence codes used in the annotations of our subset of the GOA database.

The ECR variable is represented by a CPT that is conditional on the state of the *evidence code* variable. The value of ECR is deterministic given an evidence code, this is modelled as shown by setting the likelihood for the corresponding value to 1 in the CPT.

In contrast to *evidence code* and ECR , the $Depth$ variable is modelled as a continuous variable. The variable has no parents and therefore models our prior expectation of the depth value. A standard approach to modelling the expected

Figure 4.2: Bayesian Network for *GAQ* score

value of a continuous node is to use a continuous probability distribution such as a Normal Distribution. A normal distribution $\mathcal{N}(\mu, \sigma)$ represents the expected likelihood of a value occurring in a given range. This range of values is defined

by the mean of those values μ and standard deviation from the mean σ . To determine these values we analysed the depths of GO terms used in annotations taken from our subset of the GO annotations database. From this analysis we established a mean of 6.7, and standard deviation of 2.1.

The GAQ metric score is a deterministic function of the ECR and depth. The state of the variable *GAQ* is therefore modelled as a continuous function of the depth and ECR values: $depth \times ECR$.

This network illustrates the potential of Bayesian Networks to manage with missing or incomplete data. Consider the case where we have partial metadata about a gene annotation. We have the depth but are missing the evidence code describing the process. With the current implementation of the GAQ score we are unable to make a prediction about its likely quality. However with the Bayesian Network representation we have the prior likelihood of each of the evidence codes. We can therefore make an informed prediction about the most *likely* quality based on this prior knowledge. We demonstrate the exploitation of this prior knowledge later in section 4.6.3 when evaluating datasets with incomplete metadata.

The result of the metric given by the *GAQ* random variable is a continuous value on a custom scale. Knowledge about how the score relates to quality is additional Quality Knowledge. We could include this by encoding it in the network, mapping the GAQ score to either a continuous quality value, or discrete quality state as we did with the *chemboxquality* variable in Figure 4.1. For the chembox example we modelled this mapping to a quality score intuitively. As a more systematic approach for the GAQ score we can map the *GAQ* variable to a quality value on a continuous quality scale based upon some data analysis.

This mapping is a function $f : \mathbb{R} \mapsto [0..1]$, that maps from our GAQ score to an estimate of the likely quality in the range 0 to 1. The function should represent the likelihood that the GAQ score we have is greater than other possible GAQ scores. A convenient representation of this is a cumulative distribution function of some sample GAQ scores from previous assessments. This acts as our prior knowledge about how the GAQ score relates to quality. Figure 4.3 shows an example cumulative distribution for the GAQ scores for 2443 gene annotations used in our previous study [GG11a]. A cumulative distribution function represents the likelihood that, for a given probability distribution and a value x , the value of the distribution will be less than or equal to x . We can use this function in a continuous variable to map a given GAQ score to a value from 0 to 1.

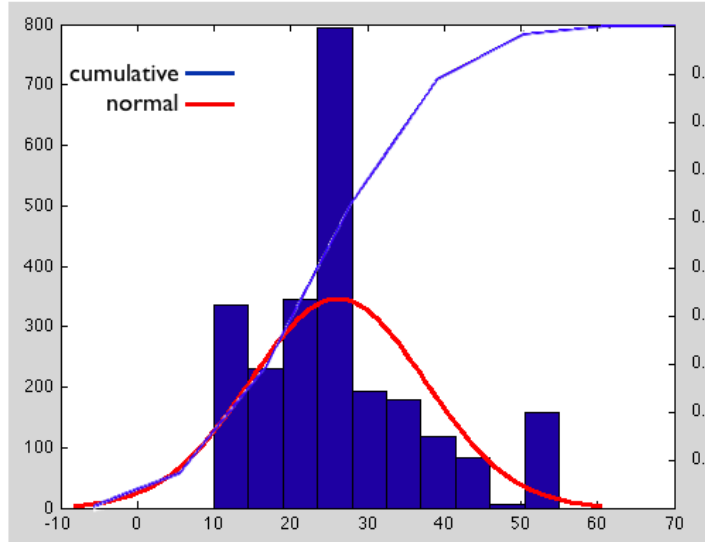


Figure 4.3: Normal and Cumulative Distribution of GAQ scores.

A challenge faced by a Quality Knowledge Engineer is ensuring that this distribution, and others modelled from data, continue to be relevant as data changes over time. The IQ Life-cycle includes feedback for reusable quality components. The engineering process is a cycle of *design*, *test*, and *deploy*. As reflected by our OPS IQ recommendations there are three strategies that the engineer can take when encoding such Quality Knowledge as a CPD:

- Compute a CPD once that then remains static.
- Initially compute a CPD but then updated it periodically.
- Calculated a CPD at query time given a reference dataset.

The advantage of mapping to a normalized quality representation is that it makes it easier to manipulate and combine metrics in an automated fashion. We demonstrate this further in the next chapter.

4.3.3 Modular Metrics

In addition to a score for a single annotation, Buza et al. also define two further metrics for GO annotations, the *Group GAQ score* and *meanGAQ*. The GAQ score for a group of gene products (e.g. proteins) S with a total of A annotations is defined as:

$$GAQ(S) = \sum_{a=1}^A (ECR_a \times Depth_a) \quad (4.3)$$

Group GAQ Score

meanGAQ is then defined for the set of gene products S with total number of annotations n as:

$$meanGAQ(S) = GAQ(S)/n \quad (4.4)$$

meanGAQ

These additional metrics reuse the original GAQ score metric in their assessment. The Group GAQ score re-uses the standard GAQ metric, and the meanGAQ score in turn reuses the Group GAQ metric. This highlights the often modular nature of metrics where they are combined and reused to construct additional more complex Quality Knowledge. Attempting to model these metrics also highlights a deficiency in the standard Bayesian Network representation. Consider an example where we are computing the group GAQ score for a group of gene products S , with a total of just two annotations. A Bayesian Network structure suitable for modelling this is shown in Figure 4.4. In order to incorporate the two annotations the original GAQ network structure is repeated twice. In reality however, the number of annotations is much greater. As a concrete example the protein ApolipoproteinA-IV (uniprot:P06272) currently has 81 annotations listed in the GO annotations database. This approach to modelling will not scale. Furthermore the number of annotations is not know at modelling time. One strategy to address this would be to maintain a collection of Bayesian Networks, each modelling a different number of annotations. However from the repetition in Figure 4.4 we see that Bayesian Networks can contain a modular structure. Template-based representations of Bayesian Networks take advantage of this modular structure. They allow the modeller to capture a general case at modelling time, which can be instantiated at run time for the data in question. The specific template-based approach we have adopted for our Fragments is Multi-Entity Bayesian Networks (MEBNs).

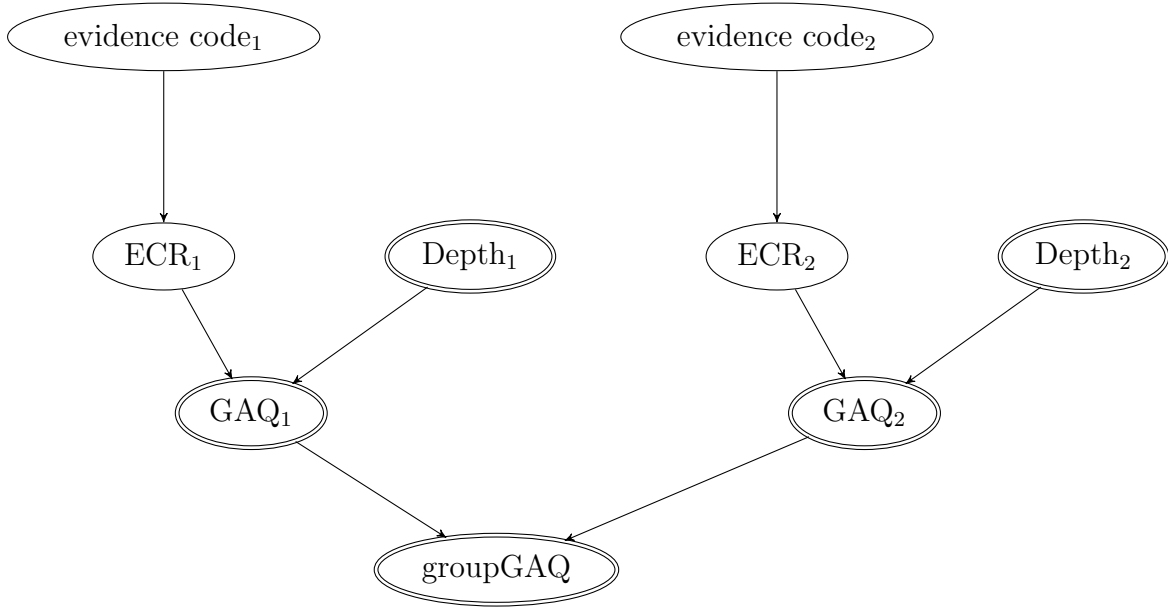


Figure 4.4: Bayesian Network for *groupGAQ* with two annotations.

4.4 Multi-Entity Bayesian Networks

The Bayesian Networks we have discussed so far have been constructed for a single task that remains static. However, for our encoding we need to:

1. Dynamically build networks in response to the data.
2. Align that data with resident nodes in our Bayesian Network.

Template-based models use Bayesian Networks to describe uncertainty in a modular fashion. Probabilistic Relational Models [Kol99], Objected Oriented Bayesian Networks (OOBNs) [KP97], and Multi Entity Bayesian Networks (MEBN) [Las08], provide a mechanism for defining reusable fragments of Bayesian Networks. These fragments can be combined at query time to create a *Situation Specific Bayesian Network* (SSBN), a probabilistic model suitable for a specific set of data.

Template-based models consist of a mechanism for describing Bayesian Networks, and a mechanism for how they apply to their chosen data model. They take a novel approach to both aligning instance data with the networks, and achieving a modular structure by building upon ‘modular’ data models such as object-oriented, entity-relation, or logical data models.

Template-based models provide a way to describe the uncertainty of attributes for instances of the data model, and divide those descriptions in a modular fashion, one per modular unit of the data model. For example OOBNs represent an objects attributes as variables, and are applied to instances of object's to reason about the likely values of attributes. Similarly PRMs use a relational schema's attributes as variables and are applied to data instances. MEBNs use First Order Logic (FOL) as their data model, they use predicates as variables and are applied to individuals in a knowledge base to reason about the likely values for those predicates.

We have chosen MEBNs as our modelling language for Quality Fragments for four reasons:

1. They provide the necessary template-based approach.
2. They are grounded in First Order Logic making them particularly suitable to the Semantic Web.
3. There is an available Semantic Web based implementation, PR-OWL2, that implements the MEBN modelling approach using the OWL ontology language.
4. The UnbBayes framework [CLC⁺10] provides tooling support to construct MEBNs.

MEBNs are constructed as a series of templates called MEBN Fragments or *MFrag*s. An MFrag is a Bayesian Network that represents probabilistic knowledge about one or more FOL predicates. In PR-OWL2 these predicates are RDF and OWL properties. Consider again the Bio2RDF gene ontology annotations data in Listing 4.2.

In this example we have included an additional property `sn_goa:GAQscore`. An MFrag description allows us to associate variables in a Bayesian Network such as Figure 4.2 with RDF properties e.g.:

- *Depth* with the property `sn_goa:go-depth`.
- *evidence code* with the property `sn_goa:ec_label`.
- *GAQscore* with the property `sn_goa:GAQscore`.

```

1 bio2rdf_uniprot:Q8IZF5
2     a                                uniprot_core:Protein ;
3     goa_vocabulary:go-annotation    goa_resource:GDB_74 ,
4                                     goa_resource:GDB_76 .
6
6 goa_resource:GDB_75 a                goa_vocabulary:GO-Annotation ;
7     goa_vocabulary:evidence          bio2rdf_eco:0000304 ;
8     goa_vocabulary:go-term           bio2rdf_go:0016021 ;
9     sn_goa:ec_label                  "TAS"^^xsd:string ;
10    sn_goa:go-depth                  "5"^^xsd:integer ;
11    sn_goa:GAQscore                  "" ;

```

Listing 4.2: Bio2RDF Gene Ontology Annotations Data with `sn_goa:GAQscore` property.

By associating a Bayesian Network variable with an RDF property, given some instance data we can do two things:

1. If the property has a value defined like `sn_goa:go-depth` we can use the value 5 as evidence for the *Depth* variable in the Bayesian Network.
2. If the property has no value defined like `sn_goa:GAQScore` then we can use the Bayesian Network to reason about its likely value.

The semantics of this reasoning are defined via an extension to FOL semantics developed by Laskey et al. [Las08]. The MEBN approach therefore assumes the existence of a FOL knowledge base.

Figure 4.5 represents a MEBN containing four MFrag structures to model the GAQ metrics. The original GAQ score is modelled by (a) *GAQscore*, the GAQ score for a group of gene products is modelled by (b) *groupGAQ*, the mean GAQ score is modelled by (d) *meanGAQ*. We have also created (c) *productGAQ* to model the GAQ score for all annotations of a single gene product.

MFrag structures consist of three types of node, *resident nodes*, *context nodes*, and *input nodes*. Each MFrag description has two concerns: (1) describing the Bayesian Network structure using resident and input nodes, and (2) describing the instance data to which the Bayesian Network applies using context nodes.

Resident nodes are variables. Each resident node is linked with a FOL predicate and defines a probability distribution for the possible values of that predicate. These can be discrete or continuous values and therefore discrete or continuous probability distributions. During reasoning the state for a variable is determined in one of two ways: (1) if there is a value declared for the predicate, that value is used as evidence, or (2) if there is no value declared for the predicate

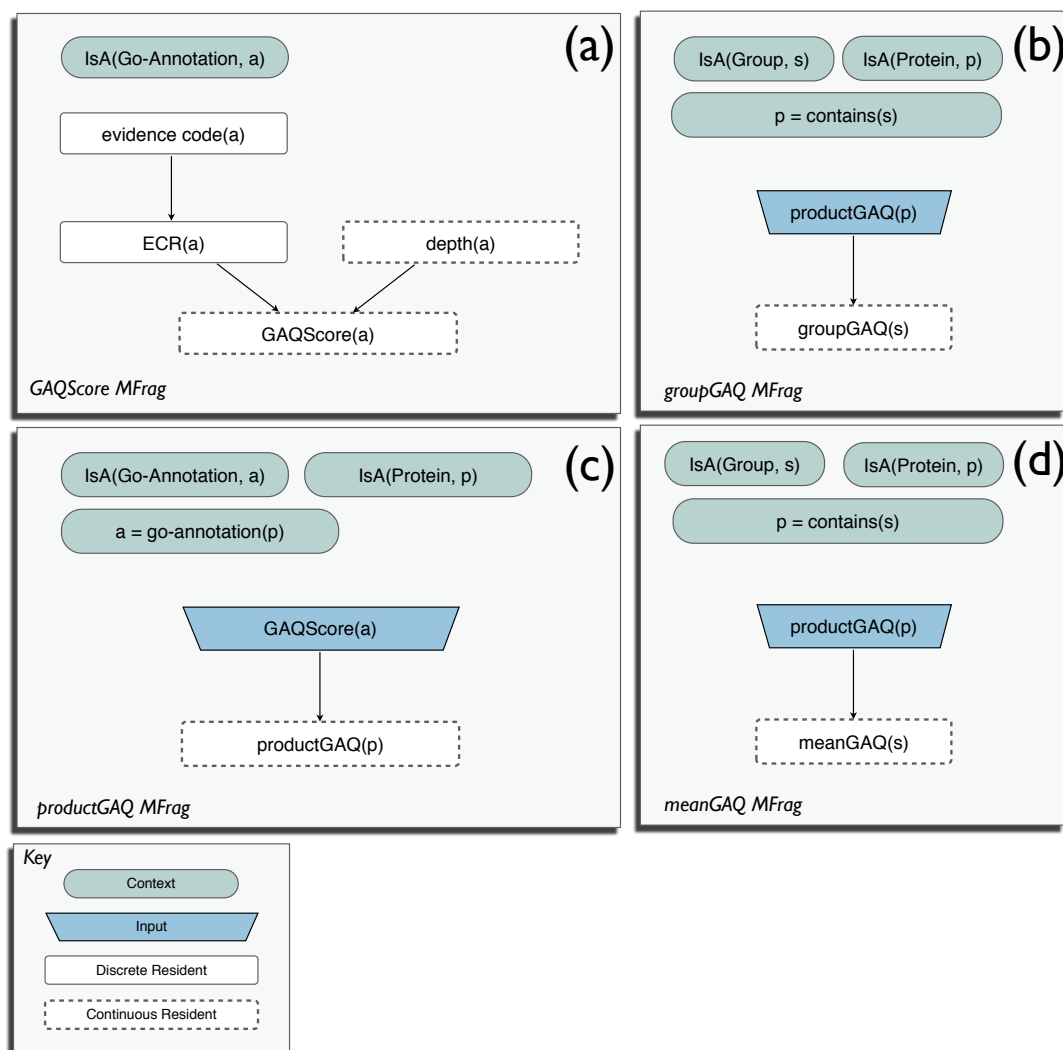


Figure 4.5: The GAQScore, productGAQ, groupGAQ, and meanGAQ MFrags.

the value is determined probabilistically using the probability distribution for that resident node.

Resident nodes modify variables that we have seen previously by including arguments, for example a for $\text{GAQscore}(a)$ in Figure 4.5. This allows the MFragment to refer to non-specific individuals at modelling time that can be instantiated at query time. To define which individuals can be used to substitute the variables the modeller uses context nodes.

Context nodes describe logical restrictions on the arguments to define which individuals in the knowledge base are suitable substitutions. MFrags also make use of FOL to describe these context constraints. For example in *productGAQ*

(Figure 4.5d) the context nodes declare three restrictions:

- $(isA(Protein, P))$ declares that individuals for argument p must be of the type *Protein*
- $(isA(Go-Annotation))$ declares that individuals for argument a must be of the type *Go-Annotation*.
- $(a = go-annotation(p))$ declares that a must be defined as an annotation of p in the knowledge-base using the *go-annotation* property. This is because we only want to consider an annotation a , if it is an annotation of p .

If a pair of individuals exists in the knowledge base that satisfy the constraints for a and p , then an instance of the *productGAQ* MFrag is instantiated with the variables assigned ready for use in reasoning.

Input nodes are how the modeller takes advantage of the modular aspect of MEBNs. Input nodes are a reference to a resident node in another MFrag. Once all MFrag instances have been created, if there are any that satisfy the context constraints for the MFrag that the input node is defined in, and the MFrag referenced by the input node, they are merged to create an SSBN.

Figure 4.6 is an example of an SSBN generated for the *productGAQ* MFrag for the gene product `bio2rdf_uniprot:Q8IZF5` in Listing 4.1. For the data in listing 4.1 there two instances of an annotation a that meet the context requirements defined in the *productGAQ* MFrag. As a result the two instances the *GAQscore* MFrag are instantiated and linked to the same resident node *productGAQ(bio2rdf_uniprot:Q8IZF5)*.

This many to one combining of templates is a common and powerful feature across template-based approaches. Each *GAQScore* MFrag instantiation that meets the context requirements is referred to as an *influencing configuration*. Figure 4.6 is therefore an example of an SSBN with two influencing configurations.

The probability distribution for the resident node *productGAQ(p)* must define a strategy for how to combine any number of influencing configurations that result from *GAQScore(a)*. These strategies are a standard approach in template modelling and are referred to as *aggregation functions* or *combining rules* [Las08]. For the *productGAQ* score the combining rule is straightforward, taking the sum of each influencing configuration:

$$productGAQ(p) = SUM(GAQScore(a)) \quad (4.5)$$

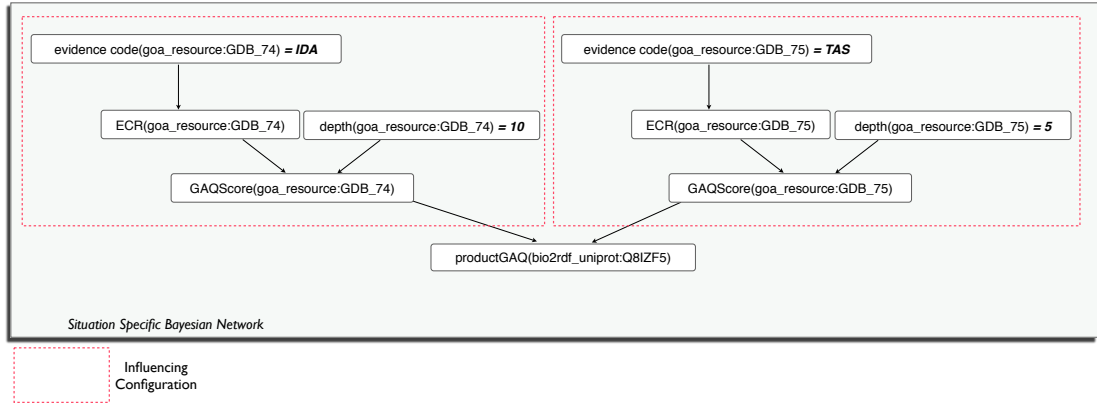


Figure 4.6: Example SSBN generated using the productGAQ MFrag for a gene product `bio2rdf_uniprot : Q8IZF5` with two annotations `goa_resource : GDB_74` and `goa_resource : GDB_75`.

The Laskey Algorithm

The procedure to manage the combining of MFrag templates to build SSBNs like the one in Figure 4.6 is defined in the Laskey algorithm [Las08]. The procedure is triggered in the same way as a conditional probability query. The query takes four arguments:

1. A description of a Multi-Entity Bayesian Network that contains one or more MFrag templates such as those in Figure 4.5.
2. A knowledge base that contains data we wish to use as evidence, e.g. the data in Listing 4.1
3. A resident node that we wish to know the likely value for e.g. `productGAQ(p)`.
4. An entity in the instance data to use as the argument for that resident node e.g. `bio2rdf_uniprot:Q8IZF5`

Algorithm 1 provides an outline of the Laskey Algorithm. The algorithm first manages the instantiating of MFrag templates based upon context constraints and the instance data (line 2). At this first stage *all* possible MFrag instances are created without any consideration as to whether they are needed for the final assessment. These instances are then merged via input nodes based upon any influencing configurations (line 3). The SSBN is then pruned to remove any MFrag instances that are not connected to the resident node in the query. Once pruned the SSBN can be reasoned over using standard Bayesian Network reasoning algorithms.

Algorithm 1: Pseudocode for Laskey Algorithm to Build an SSBN

Data: MEBN *mebn*, KnowledgeBase *kb*, query entity *e*, query resident node *r*

```

1 def void LaskeyAlgorithm(mebn, kb, e, r):
2   MFragInstances = Build Mfrag Instances(kb, MEBN) ;
3   SSBN = Merge Influencing Configurations (MFragInstances) ;
4   SSBN = Prune MFrag Instances Not Connected to Query (SSBN, e,
5     r) ;
6   results = Evaluate Probability Distributions (SSBN, e, r);

```

Quality Views using MEBNs

Whilst the focus of this chapter is modelling Quality Fragments, we discuss briefly here the role that MFrag can play in encoding Quality Views. The modular approach of MFrag provides us with a convenient mechanism for capturing a user's View. With MEBNs we can create an additional MFrag that reuses existing MFrag via input nodes and captures their subjective requirements.

Consider the MFrag represented in Figure 4.7. This reuses two resident nodes in the GAQScore MFrag. In the myGAQ resident node's CPD we can therefore define our own scoring of annotations based upon the *GAQscore* and the *evidencecode* values. For example it is common for users to filter out electronically derived annotations, denoted by the process code IEA. We might therefore define the expected value of *myGAQ* as 0 if the state of the process variable is IEA, or the value from *GAQscore* if otherwise. By using a template based approach we separate out the metric description from the View description. The user can therefore maintain their own collection of MFrag that reflect their own quality needs.

In the next section we demonstrate the application of MEBNs to Gene Ontology Annotations data from the Web of Data using the GAQ metric MFrag.

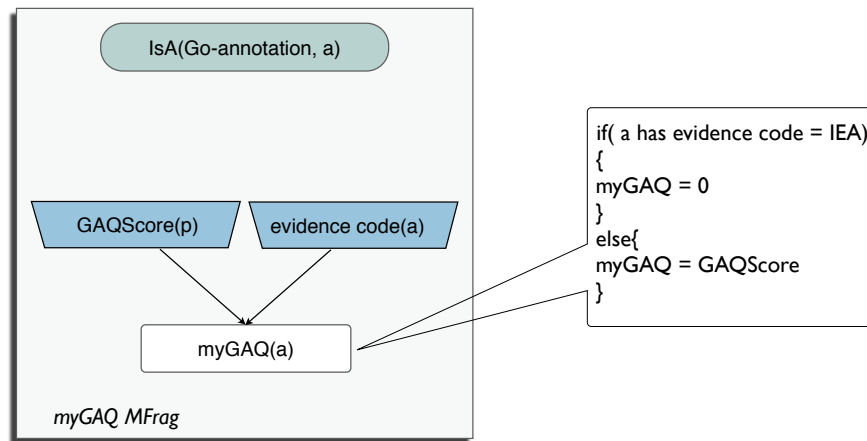


Figure 4.7: The myGAQ MFrag.

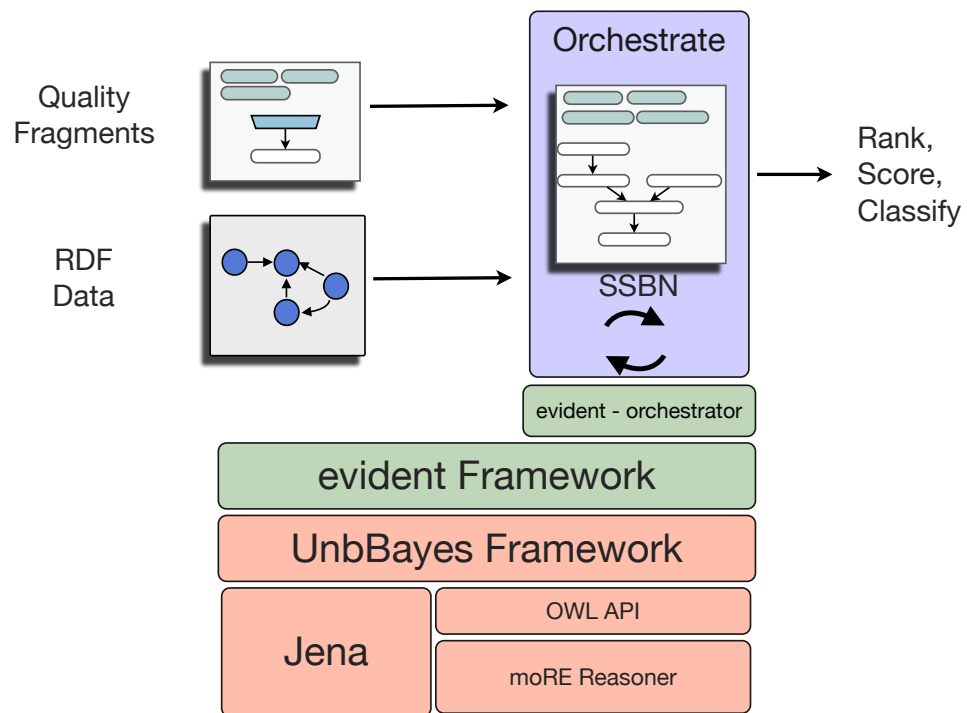


Figure 4.8: The Evident Orchestrator Framework

4.5 Implementation

In this section, we describe the implementation of our approach using MFragS to encode Quality Fragments for use in the Web of Data.

Our implementation is in two parts:

- *The encoding of Quality Fragments in a machine readable manner.* For this we have used two Semantic Web vocabularies: the existing PR-OWL2 vocabulary, and our own Evident vocabulary. PR-OWL2 is a meta-modelling vocabulary implemented as an OWL ontology that supports the description of MEBNs in RDF. The Evident Vocabulary is a smaller OWL ontology that extends the existing PR-OWL2 vocabulary to describe features specific to Quality Fragments⁴.
- *The Orchestration of Quality Fragments to perform a quality assessment.* This is performed by *Evident*, our extension of the UnbBayes framework.

The UnbBayes Framework

The UnbBayes framework is an open source Java-based framework that supports the construction and execution of MEBNs. The framework also provides an implementation of the Laskey algorithm to construct SSBNs from MFragS. The knowledge base used by UnbBayes is built upon the popular Apache Jena framework [CDD⁺04] for storing and reasoning about Semantic Web data. To evaluate hybrid Bayesian Networks, the UnbBayes framework uses the Direct Message Passing (DMP) algorithm [SC10].

UnbBayes provides a custom scripting language to define probability distributions. The language allows for the specification of CPTs or continuous CPDs. For continuous nodes UnbBayes uses a Gaussian Mixture Model (GMM) based reasoning approach for evaluation. This places a requirement on the probability distributions of all continuous nodes to declare a mean and variance. Practically this can be achieved by describing the CPD as a product of a normal distribution. The following illustrates an example script for the *meanGAQ* CPD function:

```

1 [
2 Mean( productGAQ ) * NormalDist(1,0)
3 ]

```

Orchestrate Process

Figure 4.8 provides an overview of the Evident framework and the orchestration process. To perform an IQ assessment the Evident orchestrator takes four arguments as inputs:

⁴We describe the features of the evident vocabulary in further detail in chapter 5.

- A list of URIs for RDF files to be used as instance data.
- A list of URIs for Quality Fragment files encoded in PR-OWL2, to be consulted for metrics.
- The name of the resident node from one of the Quality Fragments e.g. `productGAQ`. This will be used as the query node that we want to know the likely value of.
- The URI of an entity in one of the RDF files e.g. `bio2rdf_uniprot:Q8IZF5` to be used as the argument for the query node.

The orchestration process is then performed in three stages:

1. **Import:** The orchestrator imports the RDF data into an in memory knowledge base, and imports the PR-OWL2 Quality Fragment descriptions.
2. **Modified Laskey Algorithm:** The knowledge base and Quality Fragments are passed to a modified version of the Laskey algorithm to construct an SSBN. An OWL reasoner is used to evaluate the MFrag context constraints and build MFrag instances.
3. **SSBN evaluation:** Once generated the SSBN is passed to the UnbBayes framework to perform the Bayesian Network inference to compute the likely value for the query node.

We have modified the current implementation of the UnbBayes Framework in four key respects:

1. Extended the I/O to support the serialization of MEBNs in PR-OWL2 *with continuous nodes*. PR-OWL2 has previously been restricted to discrete nodes. We have introduced the class `pr-owl2:ContinuousResidentNode` to represent continuous nodes.
2. Modified the OWL reasoner used for evaluating context constraints from Hermit [SMH08], to the MORE modular reasoner. We discuss this implementation decision in section 4.7.1.
3. Extended the Laskey Algorithm to support the use of *any* RDF literal values as evidence. Evidence was previously restricted to OWL individuals and boolean values.

4. Modified the Laskey Algorithm to remove the use of the *uncertain reference strategy*, discussed below.

The uncertain reference strategy is concerned with the evaluation of context nodes to create MFrags instances. An OWL reasoner is used to evaluate the constraints described by context nodes. For example the constraints

```
1 isA(Go-annotation,a)
2 (a = go-annotation(p))
```

are interpreted as the following OWL class description (in Manchester Syntax):

```
1 :Go-annotation that (inverse :go-annotation value p)
```

Listing 4.3: Example of a Context Constraint interpreted as an OWL Class Description

The uncertain reference strategy is used in the case where no values for a can be found. The strategy instead searches for *any* a of the type `:Go-annotation`, and considers all matches found as possible substitutes for a . This is somewhat analogous to the OWL OWA, assuming that an annotation must exist, but has not been explicitly declared. The strategy extends this by assuming that it might be one of the annotations declared in the dataset. We have restricted the use of this strategy to take a closed-world approach. Therefore if the class restriction in Listing 4.3 returns an empty set, then we assume no possible substitution exists.

4.5.1 Quality Knowledge Encoding with PR-OWL and the Evident Vocabulary.

The encoding for our Quality Fragments is achieved using two vocabularies: the PR-OWL2 vocabulary and our own Evident vocabulary⁵.

The PR-OWL and Evident vocabularies allow us to encode Quality Fragments such as those in Figure 4.5 in a machine readable manner. Furthermore, by using an RDF based model, we are able publish these Fragments directly into the Web of Data, making them available to users as reusable quality components.

The majority of the MEBN modelling was performed using the UnbBayes Framework's graphical workbench shown in Figure 4.9, which supports the encoding of MEBNs in the PR-OWL2 format. A full PR-OWL2 based encoding of the GAQ Quality Fragments described in this chapter is provided in Appendix

⁵<http://purl.org/net/evident/>

F. In this section we detail extracts of the encoding in order to highlight key features.

The vocabularies perform two tasks:

1. Encoding: Describing the MFrag Network structure and probability distributions.
2. Alignment: Associating resident nodes in the MFrag with RDF properties in our data.

Encoding. Consider the MFrag in Figure 4.5. Lines 88-96 of Listing 4.4 defines the GAQscore MFrag:

```

88 gaqmetric:Domain_MFrag.GAQScore_MFrag
89   pr-owl2:hasOrdinaryVariable gaqmetric:GAQScore_MFrag.a ;
90   pr-owl2:hasResidentNode gaqmetric:Domain_Res.ECR,
91                           gaqmetric:Domain_Res.GAQScore,
92                           gaqmetric:Domain_Res.depth,
93                           gaqmetric:Domain_Res.process ;
94   pr-owl2:isMFragOf gaqmetric:MEBN ;
95   a pr-owl2:DomainMFrag, :NamedIndividual ;
96   rdfs:comment "The GAQ score MFrag"^^xsd:string .

```

Listing 4.4: The GAQscore MFrag in PR-OWL2

Line 89 declares the argument variable a that is used throughout the MFrag and lines 90-93 associate the MFrag with each of the resident nodes that make up the network structure, ECR, GAQScore, depth, and evidenceCode.

Listing 4.5 shows the definition of the depth resident node, declared using the new property `pr-owl2:ContinuousResidentNode` on line 151.

```

147 gaqmetric:Domain_Res.depth
148   pr-owl2:hasMExpression gaqmetric:MEXPRESSION_depth ;
149   pr-owl2:hasProbabilityDistribution pr-owl2:depth_Table ;
150   pr-owl2:isResidentNodeIn gaqmetric:Domain_MFrag.GAQScore_MFrag ;
151   a pr-owl2:ContinuousResidentNode, :NamedIndividual ;
152   rdfs:comment "Continuous resident for depth"^^xsd:string .

```

Listing 4.5: Describing a Continuous Resident Node using `pr-owl2:ContinuousResidentNode`

CPTs and continuous functions are encoded using the scripting language specific to UnBBayes. The function for the meanGAQ resident node is shown in Listing 4.6 lines 756-758:

```

755 pr-owl2:meanGAQ_Table
756   pr-owl2:hasDeclaration "[
757   Mean( productGAQ ) * NormalDist(1,0)

```

```

758 ]""^^xsd:string ;
759 a pr-owl2:DeclarativeDistribution, :NamedIndividual .

```

Listing 4.6: The Continuous Function for the meanGAQ Resident Node

This example describes a simple continuous function to calculate the mean value of influencing configurations from the productGAQ input node.

Alignment is achieved by linking the resident nodes with OWL and RDF properties used in the instance data. Lines 455-457 of Listing 4.7 illustrate how the Bio2RDF data is aligned with the resident nodes using the PR-OWL2 property `definesUncertaintyOf`.

```

455 gaqmetric:RV_depth
456   pr-owl2:definesUncertaintyOf
457     "http://sierra-nevada.cs.manchester.ac.uk/goa#go-depth"^^xsd:anyURI
      ;

```

Listing 4.7: Aligning the depth Variable with the RDF property `go-depth`

The resident node for depth is aligned with the vocabulary URI `goa_vocabulary:go-depth` used to describe the depth value in the instance data. During the SSBN building process, the Laskey Algorithm will inspect this property to discover evidence for the depth resident node.

By using this property, alignment is performed directly as part of the Quality Knowledge Encoding. It is therefore tied to the specific vocabulary terms used in the instance data. This is in contrast to the alignment in our MIM implementation where we achieved a separation of concerns by using an intermediate mapping via SPIN that allowed us to be flexible to representation. We discuss the potential to perform a similar mapping for Fragments in future work (section 4.9).

To support the automatic discovery of metrics we have also introduced the term `evident:definesMetricFor` to align data types with resident nodes. This enables the discovery of metrics for data described with a prescribed data type.

```

524 gaqmetric:RV_productGAQ
525   evident:definesMetricFor
526     'http://purl.uniprot.org/core/Protein'^^xsd:anyURI ;

```

Listing 4.8: Aligning the productGAQ variable with the type `uniprot-core:Protein` using `evident:definesMetricFor`

The example on lines 524-526 for example denotes that the productGAQ variable defines a metric for data described with the type `http://purl.uniprot.org/core/Protein`.

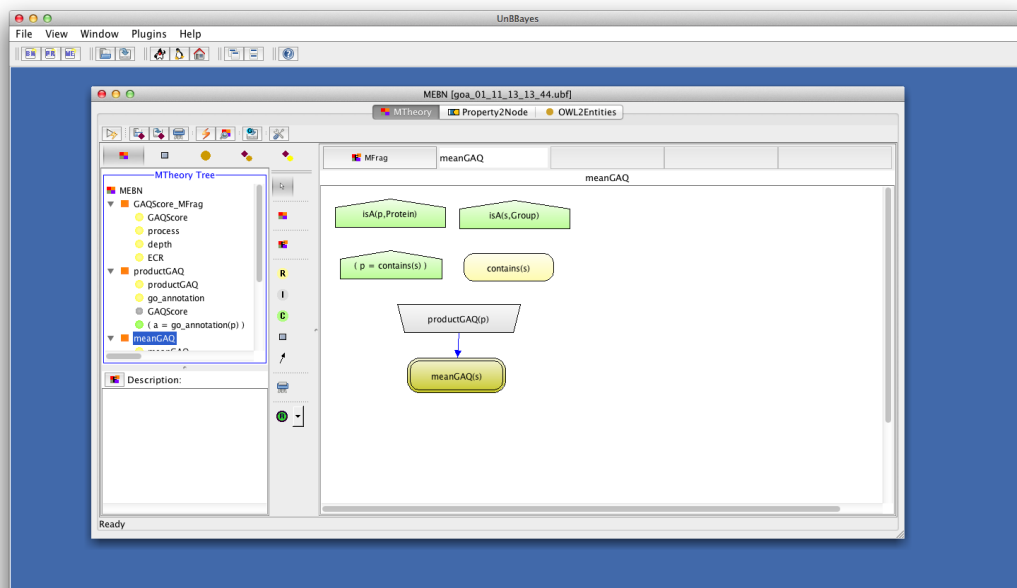


Figure 4.9: The UnbBayes Framework Workbench

4.6 Assessing Bio2RDF data with Quality Fragments

In this section, we use our Evident framework to evaluate the three main goals of our Quality Fragment approach: 1) to establish that we can successfully build SSBNs to assess Linked Data, 2) to establish that a MEBN based implementation of a metric can reproduce the same result as a reference implementation, and 3) to establish how well the probabilistic aspects of our MEBN-based Fragments improve our ability to provide an IQ assessment in the face of missing evidence.

To do this we have designed two series of experiments:

- **Replicating the GAQ:** Evaluate the ability of the Quality Fragment implementation of the GAQ metrics to a) assess data in the Web of Data and b) replicate an existing GAQ implementation.
- **Variation with missing metadata:** Evaluate the GAQ Fragments' ability to provide an IQ assessment despite missing evidence.

Replication: Buza et al. provide an encoding of the GAQscore metric as a Perl script `GA2GAQ.pl` [Buz]. The script processes *gene association* files, a

standard tab-delimited representation of gene annotations. We use the script to perform the productGAQ, meanGAQ and groupGAQ assessments. We then repeat the assessments using RDF versions of the data from the Bio2RDF project and the GAQ Fragments, and compare the two sets of results.

Variation: To simulate missing evidence we have generated a series of datasets with incrementally increasing levels of missing metadata. We perform the product GAQ assessment on each of these datasets in order to observe how well the metric performs. The intended impact of these datasets is to cause the assessment to rely on the probabilistic prior knowledge.

This experiment provides us with two evaluations. Firstly it evaluates the general approach to modelling probabilistic Quality Knowledge and with it the ability to provide an IQ assessment in the face of missing metadata. Secondly it assesses our specific modelling of the GAQ metric and how well we have estimated the true value of any missing evidence code or depth metadata. Whilst the focus of this thesis is not in the specific task of modelling, it is none the less useful to gain an insight into the engineering process.

Our expectation is that the greater the level of missing metadata, the greater the impact on our ability to accurately estimate the true GAQ score. However, we also expect that given sufficiently accurate probabilistic Quality Knowledge we should be able to provide an estimate of the likely result that we can use to make IQ decisions.

4.6.1 Data Preparation

For our evaluation we have downloaded *gene association* files by querying the QuickGo GO annotations browser [BDH⁺09], a Web-based front-end that can be used to query the GOA database. Specifically we have downloaded gene association files for 9 individual gene products, and 3 groups of gene products. Each of the 9 individual gene products's gene association file describes all of the known annotations for that gene product. The gene association files for the 3 groups of gene products represent subsets of the GOA data listed on QuickGo datasets page [Qui] that are provided by specific data sources, the Human Genome Database (GDB), Developmental Functional Annotation at Tufts (DFLAT), and the LifeDB database and describe annotations for multiple gene products.

For our evaluation we have converted the gene association data into an RDF representation using the Bio2RDF scripts used for the official Bio2RDF data

releases [CCtA⁺12]. The current official release of Bio2RDF is based on an old version of the GOA data that is no longer searchable with the QuickGo browser. So that we can compare our results with the `GA2GAQ.pl` encoding we have sourced the data directly from QuickGo, and converted it into RDF.

We have also created an additional Linked Data dataset that details the depth of each GO term in the GO ontology⁶. This dataset was generated from a flat file provided as part of the GAQ metric.

The data required for each assessment is distributed across three Linked Data datasets, two from Bio2RDF and our own GO depth dataset. To gather the data together for each assessment we have executed a series of SPARQL queries that integrate the data, generating a single RDF graph per assessment.

Listing 4.9 illustrates an example SPARQL query used to generate data from the GDB subset of Bio2RDF. The structure of the data that results from the query is the same as the previously discussed Listing 4.1. Table 4.1 provides a summary of each of the datasets that we have generated to assess the GAQ Fragments.

For our missing metadata experimentation we generated a series of missing metadata scenarios using the datasets for individual gene products. For each gene product dataset we have created two groups of data, one to simulate missing depth metadata, and one to simulate missing process metadata. For both types of metadata we have generated 20 additional datasets by randomly removing metadata from annotations starting at 5% and increasing up to 100% in increments of 5%.

4.6.2 Replicating the GAQ Metric.

We first establish whether we can successfully generate suitable SSBNs that replicate the `GA2GAQ.pl` encoding of the GAQ score given complete metadata.

For each of the individual gene product datasets we have performed the `productGAQ` assessment. For the group datasets we have performed the `groupGAQ` and `meanGAQ` assessments. Figure 4.10 shows part of an SSBN generated for the `productGAQ` assessment of the gene product Apolipoprotein A-IV (uniprot P06727) and illustrates the size and complexity of the generated network. P06727 has 81 annotations in its dataset. As a result the GAQscore network structure has been reproduced 81 times during the SSBN generation to create a Bayesian

⁶http://purl.org/net/goa/go_depth.ttl

```

1 prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
2 prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
3 prefix bio2rdf_uniprot: <http://bio2rdf.org/uniprot:>
4 prefix goa_resource: <http://bio2rdf.org/goa_resource:>
5 prefix goa_vocabulary: <http://bio2rdf.org/goa_vocabulary:>
6 prefix sn_goa: <http://sierra-nevada.cs.manchester.ac.uk/goa/>
7 prefix uniprot: <http://purl.uniprot.org/uniprot/>
8 prefix goa_vocabulary: <http://bio2rdf.org/goa_vocabulary:>
9 prefix uniprot_core: <http://purl.uniprot.org/core/>
10 prefix bio2rdf_eco: <http://bio2rdf.org/eco:>
11 prefix bio2rdf_go: <http://bio2rdf.org/go:>
12
13 CONSTRUCT{
14   sn_goa:group_gdb a sn_goa:Group .
15   sn_goa:group_gdb sn_goa:group_contains ?p .
16   ?p rdf:type <http://purl.uniprot.org/core/Protein> .
17   ?p goa_vocabulary:go-annotation ?a .
18   ?a goa_vocabulary:evidence ?e .
19   ?a rdf:type goa_vocabulary:GO-Annotation .
20   ?a goa_vocabulary:go-term ?t .
21   ?a sn_goa:go-depth ?d .
22   ?a sn_goa:ec_label ?r .
23 }
24 WHERE {
25   GRAPH <http://bio2rdf.org/bio2rdf.dataset:bio2rdf-goa-GDB>
26   {
27     ?p goa_vocabulary:go-annotation ?a .
28     ?a goa_vocabulary:evidence ?e .
29     ?a goa_vocabulary:go-term ?t .
30   }
31   GRAPH ?graph2
32   {
33     ?t sn_goa:go-depth ?d .
34   }
35   GRAPH ?graph3
36   {
37     ?e sn_goa:ec_label ?r .
38   }
39 }

```

Listing 4.9: SPARQL Query to integrate Bio2RDF data for evaluation

Network specific to this assessment. Table 4.2 provides a summary of our first evaluation, comparing each of the scores obtained by the `GA2GAQ.pl` perl script and the GAQ MFragments.

The results for our productGAQ, meanGAQ and groupGAQ assessments demonstrate that we have successfully replicated the GAQ assessment, and can therefore *replicate* an example of existing Quality Knowledge in the Web of Data using Multi-Entity Bayesian Networks. We now move on to explore how our modelling of Quality Fragments can go beyond the current encoding and manage with missing metadata.

Individual Gene Products			
Protein ID (Uniprot)	# annotations	#triples	URI
A4GW67	85	512	http://purl.org/net/goa/A4GW67
A2I9Z0	78	470	http://purl.org/net/goa/A2I9Z0
H2ETH1	86	518	http://purl.org/net/goa/H2ETH1
I6LPK7	86	518	http://purl.org/net/goa/I6LPK7
L7XCZ9	54	316	http://purl.org/net/goa/L7XCZ9
O00231	57	344	http://purl.org/net/goa/O00231
P06727	81	494	http://purl.org/net/goa/P06727
P55345	56	388	http://purl.org/net/goa/P55345
Q12802	28	170	http://purl.org/net/goa/Q12802
Gene Product Groups			
Database Name (Uniprot)	# annotations	#triples	URI
DFLAT	759	3972	http://purl.org/net/goa/DFLAT
GDB	130	876	http://purl.org/net/goa/GDB
LIFEdb	470	3756	http://purl.org/net/goa/LIFEdb

Table 4.1: Summary of Evaluation Datasets

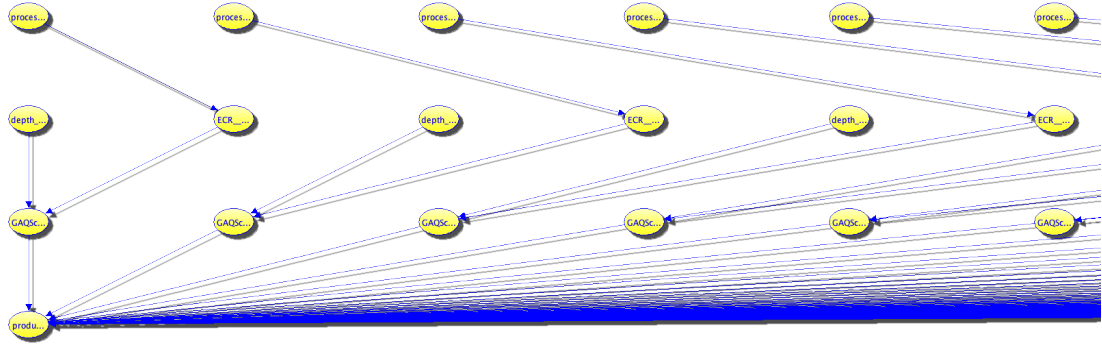


Figure 4.10: Part of the productGAQ SSBN for Apolipoprotein A-IV (uniprot:P06272) visualized in UnbBayes

4.6.3 GAQ Assessment with Missing Metadata

We have performed the productGAQ assessment for each of the datasets that simulate missing metadata described in section 4.6.1. The graphs in Figures 4.11 and 4.12 show the results of the productGAQ assessments for each gene product with missing depth metadata. In each we compare the GA2GAQ.p1 GAQ product score with complete metadata with the productGAQ MFrag scores for increasing levels of missing depth metadata.

Missing metadata has the expected impact of affecting the accuracy of the GAQ score. However, the score is often not significantly affected and approximates the original, even with significant levels of missing metadata. Consider as

Individual Proteins				
Protein ID (Uniprot)	product	GAQ MFrag	GA2GAQ.pl	
A4GW67	1276		1276	
A2I9Z0	1196		1196	
H2ETH1	1292		1292	
I6LPK7	1346		1346	
L7XCZ9	842		842	
O00231	1795		1795	
P06727	2070		2070	
P55345	1439		1439	
Q12802	534		534	
Protein Groups				
Database Name (Uniprot)	GA2GAQ.pl		GAQ Mfrags	
	group	mean	group	mean
GDB	3549	75.51	3549	75.51
DFLAT	24392	167.07	24392	167.07
LIFEdb	16435	35.19	16435	35.19

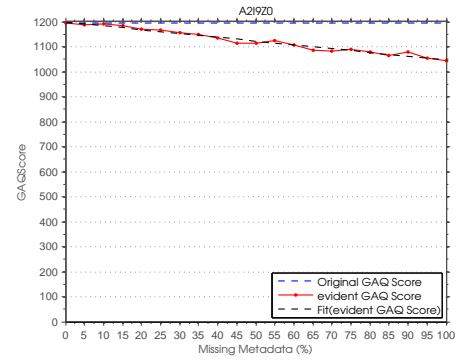
Table 4.2: Results for GAQ Mfrags compared with GA2GAQ.pl

a representative example the assessment for the protein P06727 shown in figure 4.12d. The original GAQ score for P06727 with 0% missing metadata is 2070. With 10% of the depth metadata missing the GAQ score is 2086, deviating by only 16 points (0.8%). As we get to 50% of the depth metadata missing we observe the average level of deviation with a score of 2190 and a deviation of approximately 5.7%. At its worst, with 100% of the depth metadata missing and using the probabilistic estimation of depth only, the GAQ score is 2231, only deviating by 7.8% when compared with the fully informed assessment. The largest deviation from the original score that we see in our assessments is in that of O002331, shown in Figure 4.12a. At 100% missing depth metadata we observe that the GAQ score deviates by exactly 300 points from 1794 to 1494, (approximately 17%). The most consistent assessment is that of protein P55345 in Figure 4.11a. Across the whole assessment for P55345 the productGAQ score approximates the original GAQ score well, the largest deviation is only 62 points (4.3%) at 40% missing metadata. At 100% missing metadata the productGAQ score recovers to deviate by only 32 points (2.2%). The average deviation across all assessments for P55345 is 1.8%.

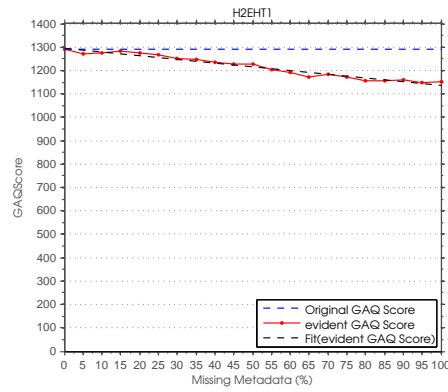
In order to establish whether we are still able to make IQ based decisions using these assessments we consider the relative rankings of the scores as availability of metadata decreases. Figure 4.13 compares each of the productGAQ scores



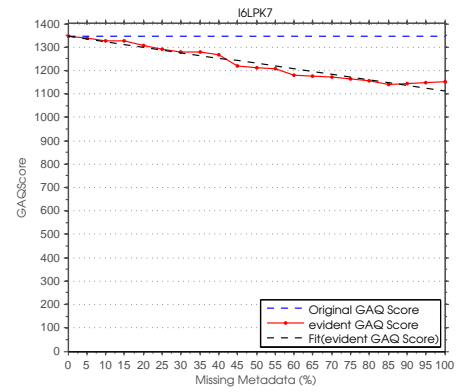
(a) P55345



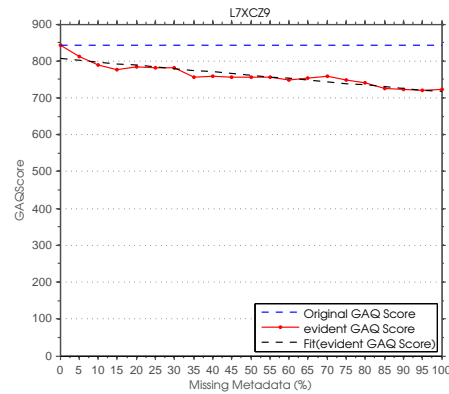
(b) A2I9Z0



(c) H2EHT1



(d) I6LPK7



(e) L7XCZ9.

Figure 4.11: GAQ Score Assessment for annotations with Missing Depth Metadata.

as the percentage of missing metadata increases. The original ranking of the

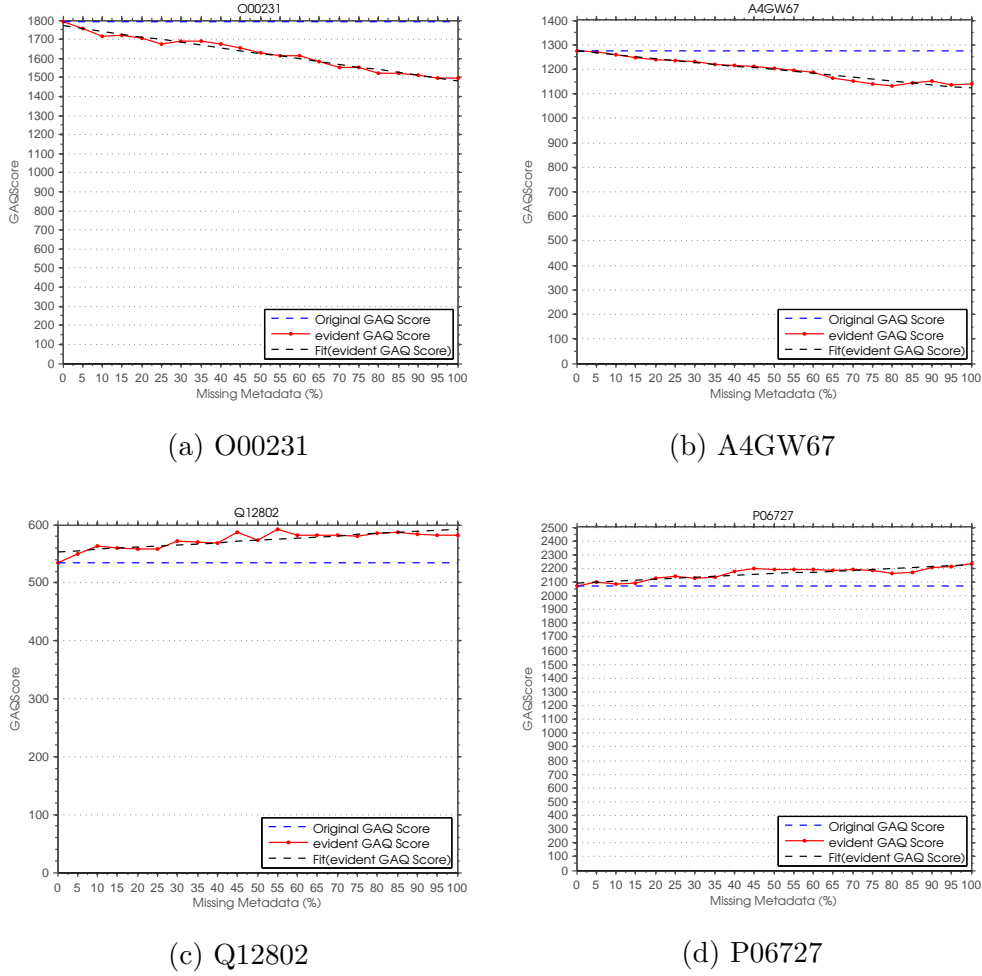


Figure 4.12: GAQ Score Assessment for annotations with Missing Depth Metadata.

gene products' GAQ scores remains intact until approximately 45% of the depth metadata is missing.

The largest permutation of the ranking occurs at 60% of missing metadata where 3 of the gene products alter their positions. We can quantitatively compare the correlation original ranking and permuted ranking by calculating a spearman's rank coefficient on a scale $[-1...1]$, where 1 means a perfect positive correlation and -1 is a perfect negative correlation. The order of the ranking at 60% compared with the original ranking provides a spearman's rank coefficient of 0.7, indicating that there is still a strong positive correlation between the original and altered ranking.

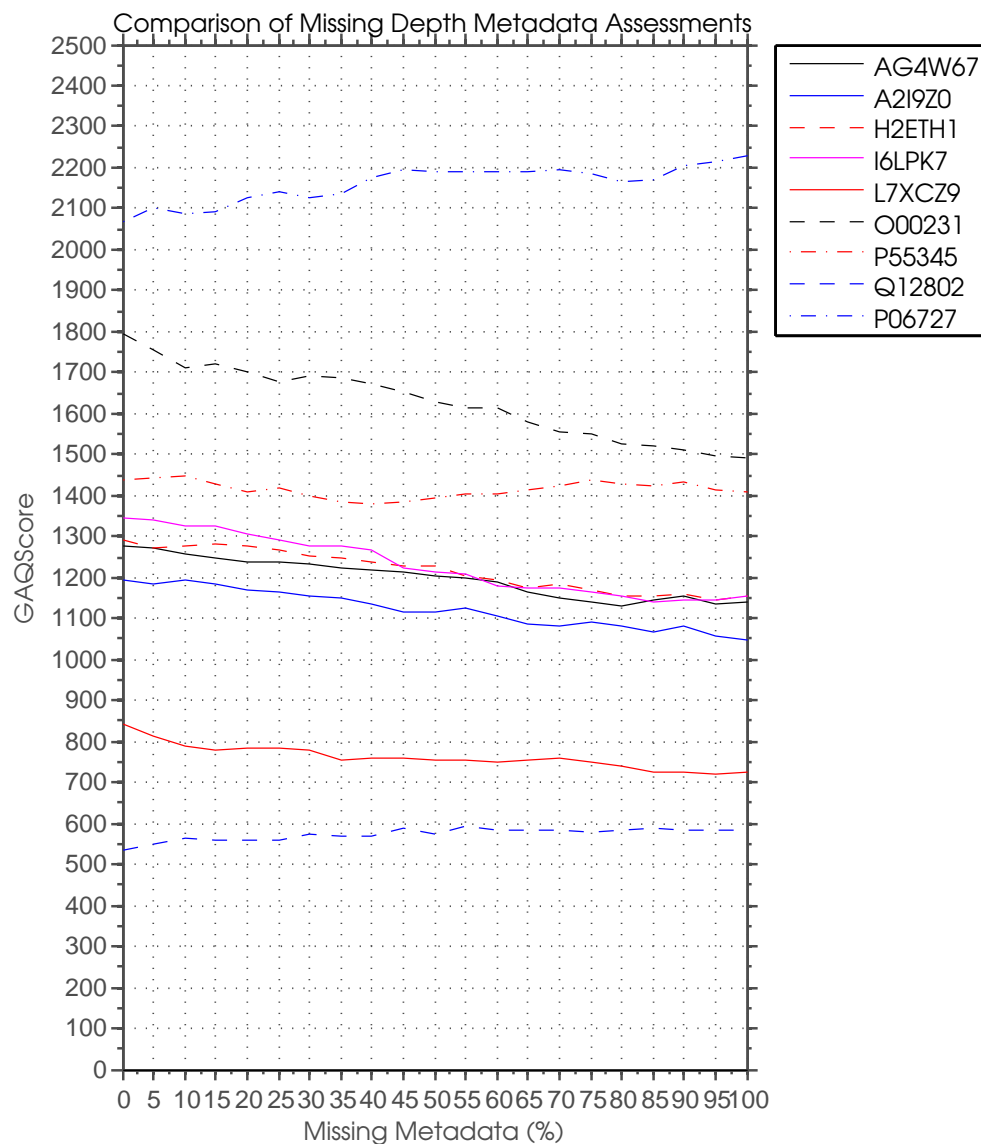


Figure 4.13: Comparison of each Gene Product productGAQ Scores with Missing Depth Metadata.

The GAQ Quality Fragments perform less well in the face of missing evidence code metadata. Figure 4.14 details the results of the experiment of missing evidence code data for gene product P06727. The metric initially performs well with 40% of the metadata missing the GAQ score is still within 10% of the original assessment at 1880, a deviation of 190 points (9%). Beyond this however the score deviates significantly from the original GAQ score. This result is consistently replicated across each of the gene products for missing evidence code metadata.

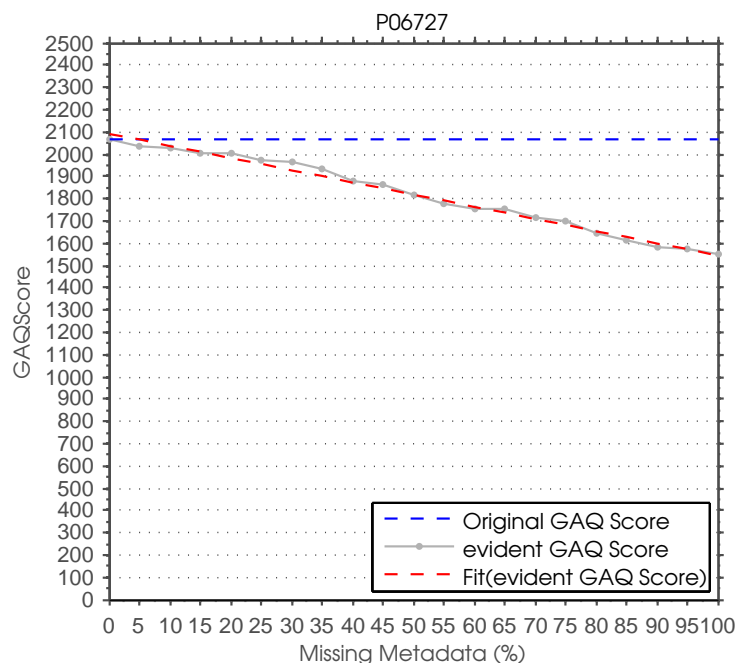


Figure 4.14: GAQ Score Assessment for P06727 with Missing evidence code Metadata.

Figure 4.15 compares each of the productGAQ scores as the percentage of missing evidence code metadata increases. In this case the original ranking of the gene products is altered at 10% missing metadata. By 20% the ranking no longer displays any positive correlation with the original ranking, providing a spearman's rank coefficient of -0.2.

The Quality Knowledge encoded in the productGAQ MFrag to estimate evidence code is less successful than that used to estimate depth. This demonstrates that the knowledge to estimate the likely evidence code is more difficult to model for an engineer than that of depth, and that the *quality* of the Quality Knowledge is key to the success of the metric.

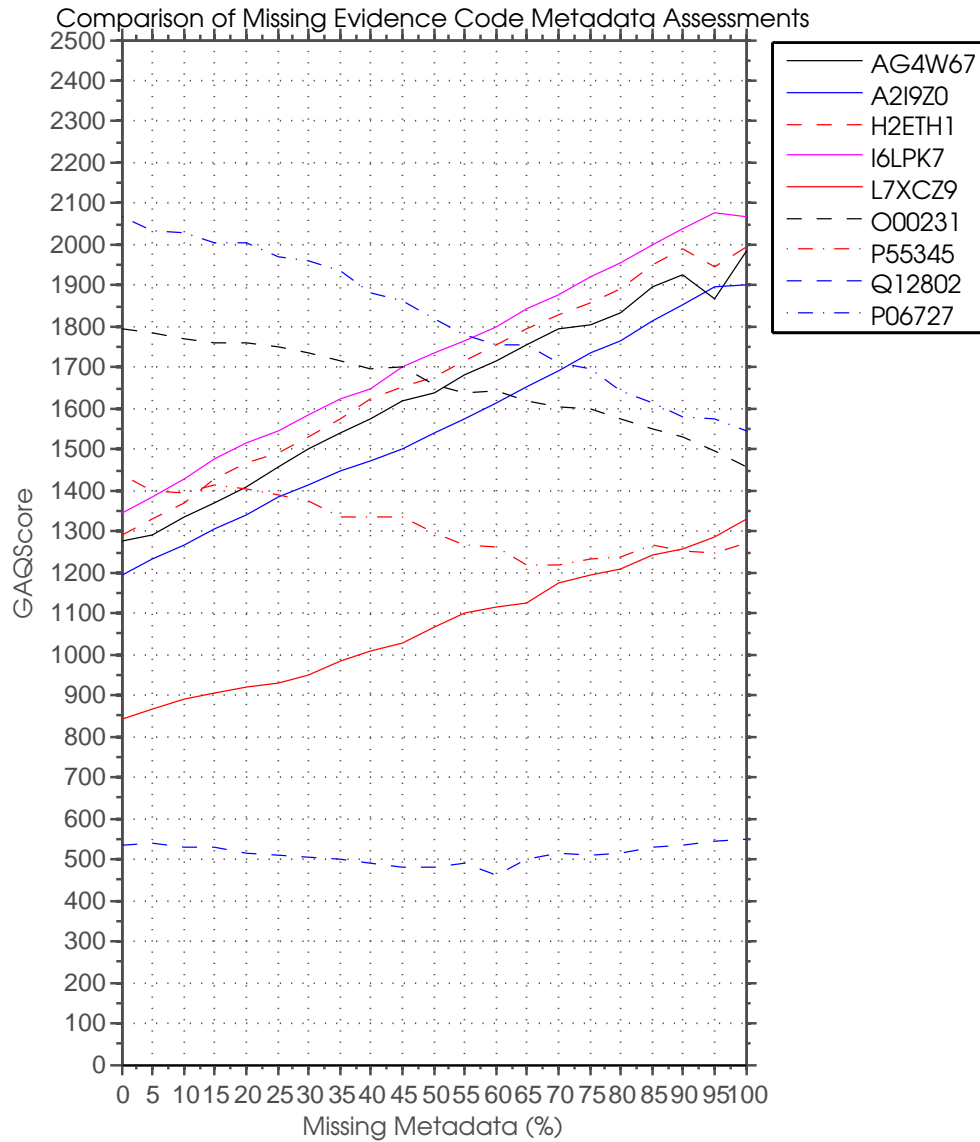


Figure 4.15: Comparison of each Gene Product productGAQ Scores with Missing evidence code Metadata.

4.7 Discussion

Multi-Entity Bayesian Networks provide an intuitive representation for Quality Knowledge allowing for expert-driven and data-driven modelling that benefits users and engineers. Engineers can benefit from the dual mechanisms for creating Bayesian Networks and then share them in the Web of Data using the PR-OWL2 and Evident vocabularies.

We have shown that template-based approaches such as MEBNs allow us to support the often modular nature of Quality Knowledge and quality metrics with three metrics, the GAQscore, productGAQ and meanGAQ.

With our evaluation we have shown that we can successfully build suitable SSBNs from these modular Fragments to assess scientific Linked Data. We have also shown that these SSBNs are able to reproduce exactly the results of an established metric implementation, in this case the GAQ metric for Bio2RDF GOA data.

Previous work has shown that uncertainty representation can manage with missing or incomplete evidence for quality assessment. We have confirmed that MEBNs are similarly capable. These results have demonstrated that the additional probabilistic Quality Knowledge that we can encode in MEBNs makes them particularly suitable to the Web of Data, where metadata availability is uneven. The probabilistic knowledge allows us to estimate the likely value of a metric with reasonable accuracy even with significant levels of missing metadata.

We observe however that the ability of the metric to successfully cope with missing metadata is dependent on the *quality* of the Quality Knowledge. That is, it is dependent on how well the probabilistic knowledge encoded in the Quality Fragment estimates the true value for the missing Quality Evidence. This is true of any uncertainty modelling. It is clear however that given a good estimation such as the depth estimation, Quality Fragments provide the ability to encode this Quality Knowledge and make use of it provide a previously unavailable assessment.

Our evaluation provides an insight into the analytical capabilities of our probabilistic approach, and the potential to evaluate the quality of the Quality Knowledge encoded within them. With these Fragments, we can begin to estimate how important particular features of an IQ metric are to the overall assessment. In our evaluation for example, we observed that our estimation of depth evidence is sufficiently accurate to make useful predictions. In contrast, our estimation of the evidence code is less successful. This provides two insights. Firstly that a Quality Knowledge Engineer modelling the GAQ metric should focus efforts on improving process estimation. Secondly it informs us that it is more valuable for a user to obtain evidence code metadata than depth metadata for a successful evaluation. This insight and analytical capability supports a well known type of investigation in Bayesian Network modelling known as *value of information*

[JL94] [JN07], where the goal is to establish which evidence the most important to observe, in order to improve the reliability of the evaluation.

Establishing MEBNs and PR-OWL as suitable representations of predictive Quality Knowledge is the first step towards our goal of developing a broader framework for reusable quality components in the Web of Data. With PR-OWL2 engineers can begin to build a library of Fragments that can be shared into the Web of Data. As a general and uniform approach to modelling Quality Knowledge, engineers can adapt and combine these MEBN-based Fragments to create new quality components. Users can similarly reuse these Fragments as they are, combine them, or adapt them to build their own Views.

4.7.1 Some Notes on Implementation

The focus of our implementation was a proof of concept for Quality Fragments encoded as Multi-Entity Bayesian Networks, and not the performance of the implemented solution. We choose to discuss here however two observations related to Bayesian Network evaluation and the use of OWL reasoning and the scalability of our current implementation.

A significant impact of uncertainty is computational complexity. This is particularly true of large scale Bayesian Networks with significant numbers of variables unknown. In our experimentation with missing evidence code metadata we observed a significant increase in the running time and memory usage as the number of pieces of missing metadata increased. This is because the complexity of the DMP algorithm, and other algorithms used to evaluate hybrid bayesian networks, is particularly affected by the number of unknown discrete states [SC10]. In our evaluation memory usage restricted our ability to consider group gene association datasets larger than those described.⁷

The DMP algorithm allows for the tuning of parameters to reduce computational complexity in return for a loss in accuracy of the assessment. This is achieved by approximating intermediate results in the network. There is also ongoing research to evaluate and improve the performance of the DMP algorithm [CCL⁺10]. We are already providing an estimation of the result with our missing metadata experiments where no assessment was previously possible. As a result a legitimate line of investigation would be to trade off of accuracy to continue

⁷Experiments were performed on a dedicated machine with a 1.7 GHz Intel quad-core i7 processor and 8GB of RAM

to provide an assessment. Future work could investigate the parameters of this trade-off.

A further impact on the performance of our prototype was the reliance on an in memory RDF knowledge base, and the use of an OWL reasoner to evaluate the constraints. As the size of the knowledge base grew for larger datasets, so did the time taken to evaluate the axioms defined by the context constraints. The current UnbBayes implementation makes use of the Hermit reasoner. As an immediate remedy to this we altered the reasoner used by UnbBayes. We consulted the results of the recent OWL reasoner benchmarking workshop [BGG⁺13] and identified the MORE modular reasoner [RGH12] as a candidate to improve reasoning time. MORE integrates several reasoners, and identifies subsets of an ontology that can be reasoned over with faster, less feature complete reasoners, so that slower reasoners make as few computations as possible.

We propose that a further solution to managing this complexity would be to use a rule-based mechanism such as SPARQL to identify evidence. Whilst we would lose the inferencing capabilities of OWL, we expect that we would see an improvement in performance.

4.8 Comparison with Related Work

Previous work has demonstrate the use of Bayesian Networks to model IQ and trustworthiness assessment in domains such as peer-to-peer networks and Web recommender systems. Existing approaches do not currently use template-based modelling. Wang et al. [WV05] use bayesian networks to model the trustworthiness of peers in a p2p network. The author suggest the need to maintain a separate Bayesian Network for each peer in the network. This limitation could be addressed by using a template-based model such as MEBNs that can be instantiated to create an SSBN. Other approaches build situation specific networks procedurally, for example [YGL11] builds Bayesian Networks for the purposes of content filtering in Social Networks. Quality Knowledge is not however encoded in a declarative representation like Quality Fragments that can be shared and reused, but instead entirely encapsulated by the procedure.

A limitation we see with our current approach is that the score we give to the user does not provide information to distinguish between uncertainty that is built into the model, and uncertainty that comes about through missing evidence.

Previous work by Ceolin [CVHF10] [CNF12b] demonstrated the use of subjective logics to assess the trustworthiness of annotations in the Web of Data. One of the strengths of approaches based on subjective logics and Dempster-Shafer modelling of uncertainty is that the units of uncertainty are multi-dimensional, capturing belief, disbelief and uncertainty. The approach proposed by Ceolin however, loses the intuitive representation of the relationship between variables that is provided by Bayesian Networks, and probabilistic graphical models in general. This is an important consideration where we are trying support the elicitation of domain knowledge from expert scientists. Jøsang [Jøs08] however has proposed an approach that combines subjective logic based reasoning with Bayesian Networks that might be used to leverage the strengths of both mechanisms.

Beyond PR-OWL there are a number of alternative approaches to modelling Bayesian Networks using Semantic Web technologies such as BayesOWL [DPP06], OntoBayes [YC05], and OMEN [MNJ05]. OntoBayes and OMEN focus more on the task of including Bayesian approaches to the semantics of OWL and their reasoning, and are primarily aimed at the task of Ontology Mapping [MNJ05] [DPP06]. As such they are limited in their ability to fully encode a Bayesian Network. We are interested in PR-OWL2 primarily as a mechanism for describing a MEBN using Semantic Web techniques, that could be aligned with existing RDF data. In that respect OntoBayes is the closest alternative to PR-OWL for expressing Bayesian Networks associated with OWL ontologies.

Using OntoBayes, Bayesian Networks are created by directly annotating the properties of the existing ontology using a new property `rdfs:dependsOn`, to describe a probabilistic influence between two properties. OntoBayes descriptions are however more heavily integrated into OWL vocabulary descriptions, where PR-OWL2 is a mapping between MEBNs and OWL and RDF properties that happens to also be described using OWL. A benefit therefore of using PR-OWL2 is that it clearly separates the Bayesian Network description from the Ontology description, allowing them to be shared separately as reusable quality components.

4.9 Future Work for Quality Fragments

We are currently restricted by the UnBayes implementation with what we can express as the logic for the CPDs. As a result we have to make the assumption

that evidence values have been calculated beforehand and are available as metadata, as we have with the depth of a GO annotation using `sn_goa:go-depth`. This metadata may not be available, or may change over time. As such it would be useful to be able to calculate this metadata at query time. In Missier’s eScience setting [Mis08], Web Services are used to calculate evidence values. We believe there is scope to extend our approach in a similar manner, and make calls to other services to generate Quality Evidence.

A related limitation lies in the process of alignment. Currently data is expected to be described using specific OWL and RDF classes or properties such as `sn_goa:go-depth` or `sn_goa:ec-label`. With our MIM solution we were successful in separating the concerns of encoding and alignment through the use of the report generating rules. We propose that a rule-based mapping layer between the context nodes and the data could broaden the potential application of the Quality Fragments. This is also in line with our proposed solution to improving the performance of the current OWL-based context node evaluation.

Finally, we mentioned a number of approaches to modelling uncertainty in addition to Bayesian Networks. Useful future work would be to attempt to model the same Quality Knowledge using alternative representations for uncertainty such as subjective logics. The goal would be to attempt to measure the expressiveness of alternative representations, as well as the modelling complexity to establish their relative strengths and weaknesses for Quality Knowledge Encoding.

4.10 Chapter Summary

In this chapter we have presented a novel approach for encoding Quality Fragments using Multi-Entity Bayesian Networks. We have illustrated the suitability of Bayesian Networks for modelling predictive Quality Knowledge, and have shown that template-based approaches are particularly well suited. We have grounded MEBN-based Fragments in the Web of Data using PR-OWL2 and provided an encoding of the GAQ series of metrics. We have demonstrated the ability of the approach to assess the quality of heterogeneous Linked Data, producing Situation Specific Bayesian Networks that replicate a reference implementation of the GAQ metric. We have also shown that the probabilistic aspects of Bayesian Networks are well suited Web of Data by evaluating a series of datasets with missing metadata.

In the next chapter we extend our use of MEBN-based Quality Fragments and present a procedure to automatically build them that is informed by provenance data.

Chapter 5

Procedurally Building Quality Fragments using Provenance

“It is by logic that we prove, but by intuition we discover.”

- Henri Poincaré

5.1 Chapter Introduction

In this chapter we continue our modelling of predictive Quality Knowledge and propose a novel procedure to automatically generate Quality Fragments from provenance data. The goals of our procedure are:

- To increase the scope of assessment for Quality Fragments by considering how we can use them to assess not only the types of data that they have been designed for, but also the quality and trustworthiness of other related resources in a provenance graph.
- To support the engineer by bootstrapping the engineering process and automatically creating Quality Fragments.

Our aim is to establish the necessary processes and metadata required to automatically build MEBN-based Fragments from provenance data.

We begin the chapter discussing the intuition that allows us to propose this procedure, examining the role of provenance in predictive Quality Knowledge. We observe a commonly applied property in provenance-based assessments where the quality of data is informed by the quality of other entities, actors, or processes involved in its production. This property enables us to propose a general solution for automatically generating Fragments, by joining together existing Fragments using provenance information.

To ground our discussion of provenance in the Web of Data we focus our attention on the PROV provenance model. In the process we identify a limitation to PROV in capturing the metadata required for our IQ assessment. To address this limitation we propose *influence factor* as an additional feature for provenance descriptions to *quantify* the influence one provenance entity has on another. We model this feature as an extension to the PROV provenance model.

We go on to describe our procedure to build Fragments informed by provenance. To evaluate our approach we have designed an experiment to replicate an existing Bayesian Network-based metric for Wikipedia revisions proposed in Zeng et al. [ZAD⁺06]. Using our procedure, we generate a Fragment from provenance for Wikipedia revisions and compare our results with the existing network.

5.2 Provenance-based Quality Knowledge

In chapter 2 we established the prominent role of provenance in predictive Quality Knowledge and the increasing availability of provenance metadata in the Web of Data. In a distributed environment such as the Web of Data, provenance information is an important component of IQ assessment. As such the evaluation of quality, reliability and trustworthiness has been a primary use-case for the development of the PROV specification [GM13].

The motivation to use provenance data is that we do not always have Quality Knowledge that is suitable to assess the quality of a particular Web resource. Previous work has demonstrated that metrics that make use of provenance data can increase the number of Web resources we can assess by considering the quality and trustworthiness of related resources in a provenance graph [Gol05] [ZDSM05] [GMM⁺09] and how their quality relates to the resource in question. Furthermore a combination of both these provenance-based metrics and regular metrics has been shown to improve performance beyond the application of either type of

metric in isolation [CGvH⁺12].

In this chapter, we explore the potential to use MEBN-based Quality Fragments to model provenance-based metrics. Further to this we demonstrate the ability to build these Fragments automatically based on common features of provenance-based metrics. In chapter 2 we characterised this as a combination of two types of Quality Knowledge, *intrinsic* and *provenance based*. Specifically we exploit a common intuition that any resource that has influenced the production of another Web resource will have affected its likely quality. Therefore if we can evaluate the quality of these influencing resources, we can use that information to inform us of the likely quality of the resources that they have influenced.

This intuition that makes use of two types of provenance

- *lineage* provenance that describes the lineage of the web resource i.e. the other resources that were involved in its production.
- *how* provenance that describes to what extent those other resources contributed to its production.

Practically, our goal is to develop a procedure $G(Pr, Q) \rightarrow q$ that takes two arguments:

- A provenance graph Pr describing the production of a Web resource e_i
- A collection of Quality Fragments Q that encode intrinsic Quality Knowledge for other Web resources in that provenance graph

From this we build a new Quality Fragment q that defines a quality assessment for the Web resource e_i , based upon the intrinsic quality of the resources that produced it. The aim is to make q reusable for any Web resources of the same type as e_i . We are therefore proposing that we can automatically infer the provenance-based Quality Knowledge from the provenance data.

Figure 5.1 shows an overview of a concrete example of this intuition using an example Web resource for a chembox dataset <http://purl.org/chembox/Ethane>.

The chembox Web resource was generated by extracting data from the Wikipedia article <http://www.wikipedia.org/en/Ethane>. This *lineage* provenance is captured in the metadata for our chembox with the following provenance statement:

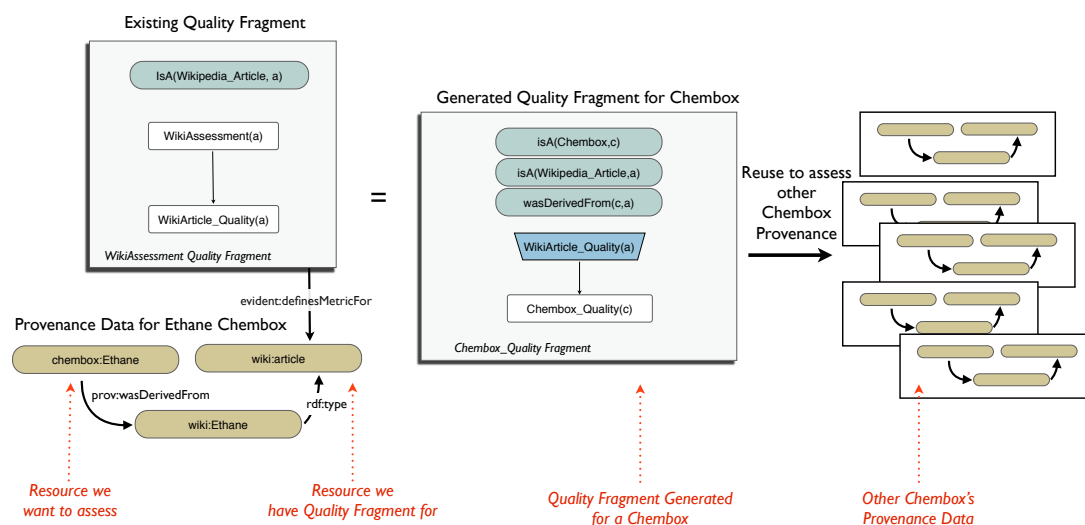


Figure 5.1: Quality Fragment Generation for Chembox based on WikiArticle Quality Fragment and Provenance Data.

```

1 <http://purl.org/net/chembox/Ethane>
2   a <http://purl.org/net/chembox/chembox> ;
3   prov:wasDerivedFrom <http://en.wikipedia.org/wiki/Ethane> .

```

Listing 5.1: Provenance Describing that the chembox *wasDerivedFrom* a Wikipedia article.

We wish to judge to quality of the Web resource `<http://purl.org/net/chembox/Ethane>` but don't have a Fragment suitable to directly assess it. Instead we have the *WikiAssessment* Fragment that models the intrinsic quality of the original Wikipedia resource from which the chembox was derived.

The WikiProject Chemicals group have actively assessed many of the chemical structure articles, and assigned them a discrete quality assessment value. Listing 5.2 shows a Linked Data representation of this assessment information.

The WikiAssessment Quality Fragment in Figure 5.1 captures this, modelling the quality of Wikipedia article as conditional on the WikiProject assessment value.

Given these two pieces of information: 1) that the chembox was derived from the Wikipedia article and 2) the Fragment to assess the Wikipedia article, it is possible to intuitively form some estimate as to the likely quality of the chembox, by modelling its quality as conditional on the WikiProject Chemicals assessment.

```

1 assessments:Ethane
2   dbprop:name "Ethane" ;
3   assessment:assessmentValue "B" ;
4   assessment:importance "Top" ;
5   assessment:importanceDate "2010-05-09" ;
6   assessment:qualityDate "2010-05-09" ;
7   assessment:release "" ;
8   assessment:review "" ;
9   assessment:score "1550" ;
10  foaf:page <http://en.wikipedia.org/w/index.php?title=Ethane> .

```

Listing 5.2: RDF Representation of WikiProject Chemicals Assessment Data.

This conditional modelling is shown in the structure of the generated *Chembox* Quality Fragment in figure 5.1. The Fragment’s structure and context nodes have been directly informed by the provenance data, and the existing *WikiAssessment* Quality Fragment.

Whilst the provenance information is sufficient to infer the structure and context nodes, to create the CPD for the *Chembox_Quality(c)* resident node, we require additional knowledge about *how* the chembox was derived and to what extent the Wikipedia page has influenced the chembox. For example we might decide that the WikiProject Chemicals assessment applies directly to our chembox data because it is the only source of data used, and assign a quality value on the same scale in a one-to-one mapping. Here we are choosing to model the chembox quality weighted by a *quantitative* measure of how much the Wikipedia page contributed to the chembox. This is part of the *how* provenance, and is not currently captured in the provenance metadata for our chembox. We discuss later in section 5.4 how we propose to model this information using a new provenance feature called *influence factor*.

5.2.1 Our Intuition

We generalise our intuition is as follows:

Let E represent a set of Web resources, T a set of data types indexed by E , and Q a set of Quality Fragments indexed by T . Let P be a function $P : E \rightarrow \{E' \subset E\}$, where P_{e_i} refers to the set of Web resources $E' \subset E$ that have directly influenced the production of e_i (the *lineage* provenance). Let the function $I : \langle E, E \rangle \rightarrow \mathbb{R}$, provide a quantitative measure of the influence between two Web resources $e_i \in E, e_j \in E, e_i \neq e_j$ (the *how* provenance).

Consider two web resources $e_1, e_2 \in E$ where $e_1 \in P_{e_2}$, that is the production of e_2 was influenced in some way by e_1 ; and Quality Fragment $q_{t_1} \in Q$, $t_1 = T_{e_1}$, i.e. a Quality Fragment for type of e_1 .

Our intuition is that the Fragment for e_1 : $q_{t_1}(e_1)$ *informs* the Fragment $q_{t_2} \in Q$, $t_2 = T_{e_2}$, such that $q_{t_2}(e_2)$ is conditional on $q_{t_1}(e_1)$. Crucial to modelling how q_{t_1} informs q_{t_2} is understanding the nature of *how* e_2 was influenced by e_1 . We use information from I_{e_i, e_j} to inform the conditional relationship.

To model the set of influencing relations P for Web resources in the Web of Data we adopt the PROV model of provenance. The set of types T will be defined by the data that we are assessing, for example line 2 in Listing 5.1 that declares the Web resource `<http://purl.org/net/chembox/Ethane>` as the type `<http://purl.org/net/chembox/chembox>`. In order to align the Quality Fragments in Q with the data types in T so that we can discover for example $Q_{chembox}$, we use the evident vocabulary as shown previously in Figure 5.1 using the `evident:definesMetricFor` property. We discuss further the use of the evident vocabulary to model provenance-based Quality Knowledge later in section 5.6.4.

We use this intuition to propose a procedure that uses existing provenance data for a Web resource to inform the construction of a Fragment. The generated Fragment is then a template that can then be reused to assess the quality of Web resources of the same type, given a similar provenance graph.

Our ability to propose a solution that creates Fragments that can be *reused* from existing provenance data relies on the fact that provenance metadata for electronic data is typically automatically generated [Mor10], and as such retains a relatively uniform representation for a particular type of data. We can therefore take a single instance of provenance metadata as an exemplar, and use it to generate a reusable quality component that can be applied to current and future data produced by the same process.

In order to automatically generate new MEBN-based Fragments we identify three specific modelling tasks:

1. Model the Fragment's network structure to capture the conditional relationships described by the provenance. The information to support this modelling comes from the *lineage* provenance in P .

2. Model the Fragments context nodes to capture the context in which the new Fragment is applicable. The information for this modelling comes from the *lineage* provenance in P , and the types in T .
3. Model the CPDs for each resident node, defining how to evaluate the conditional relationship between the existing and new Fragments. The information to model the CPDs comes from the *how* provenance in I .

The exact procedure to support each of these tasks is dependent upon the features available in the provenance. In the process of grounding our work in PROV, we define the scope of its features that will support our procedure.

5.3 PROV

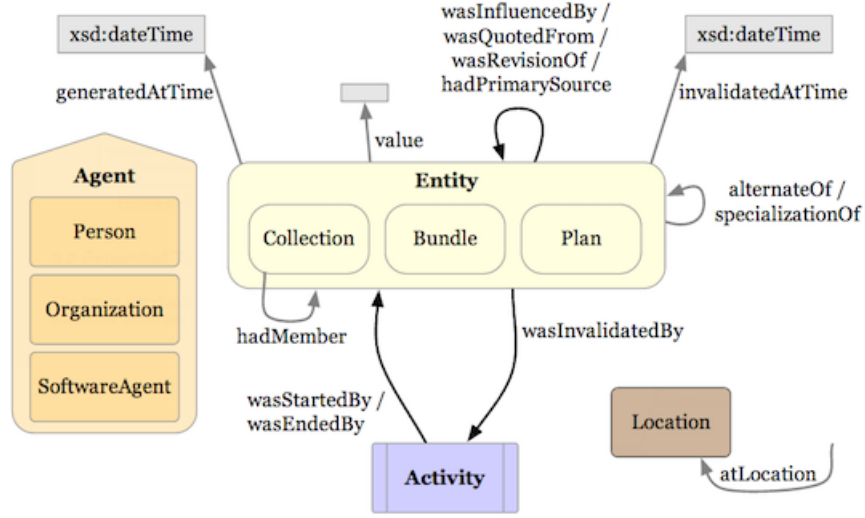
In Chapter 2, we introduced PROV, a recently established series of specifications developed by the W3C's Provenance Working Group. These specifications support the encoding, publication, and interchange of provenance data on the Web in a machine readable fashion. The PROV specifications were developed as the result of an extensive evaluation of existing approaches and a diverse set of use-cases [GCG⁺10], including those from the Life-Sciences domain.

To better understand the provenance metadata available in PROV-based provenance descriptions we consult two resources: 1) the PROV specification documents, and 2) the ProvBench datasets from the first ProvBench Workshop [BZMGPS13]. The recently established ProvBench datasets provide a series of exemplar provenance data that the research community can use to develop and evaluate provenance-based applications. There are currently 9 collections of provenance data available describing provenance in a range of domains including scientific Workflow systems, simulation experiments, and Web-based resources such as Wikipedia. A number of the ProvBench submissions also provide tooling to support the user in generating additional provenance data. In this chapter we make particular use of the **wikipedia-provenance** dataset and tooling [MC13].

Our previous discussion of PROV introduced the core concepts of the PROV model: Entities, Agents and Activities, and its approach to modelling the production of data as a graph. Figure 5.2 details an expanded version of the model.

For our procedure we need to establish:

1. What constitutes a valid PROV graph irrespective of domain.

Figure 5.2: The Extended PROV Model [LSM⁺13]

2. Which features of a valid PROV graph are currently out scope for the procedure.
3. What features of a valid PROV graph are relevant for the purposes of our procedure. The key features are the *lineage* provenance to inform the construction of the network structure, and the *how* provenance to inform the CPDs.

For 1, we refer to the PROV-Constraints specification [CMM13] which declares a set of definitions, inferences and constraints that define the subset of all possible PROV instances that are *valid* PROV instances. We make the assumption that all PROV graphs are *valid* PROV graphs with respect to the PROV constraints. For 2 and 3 we consult the PROV Data Model (PROV-DM) document and examine the usage of relevant PROV features in the ProvBench datasets.

PROV-DM defines a core set of provenance concepts that covers the fundamental descriptive features available in the PROV model. Table 5.1 details each feature and its related term in the PROV-O vocabulary, and the coverage of the features in the ProvBench datasets.

Central to our procedure is the *lineage* provenance, descriptions of relationships between provenance elements about whether they influenced each other. Lineage provenance is captured in the PROV model primarily by using the **was-InfluencedBy** relation. The subject of the influence relation is the *influencee* and

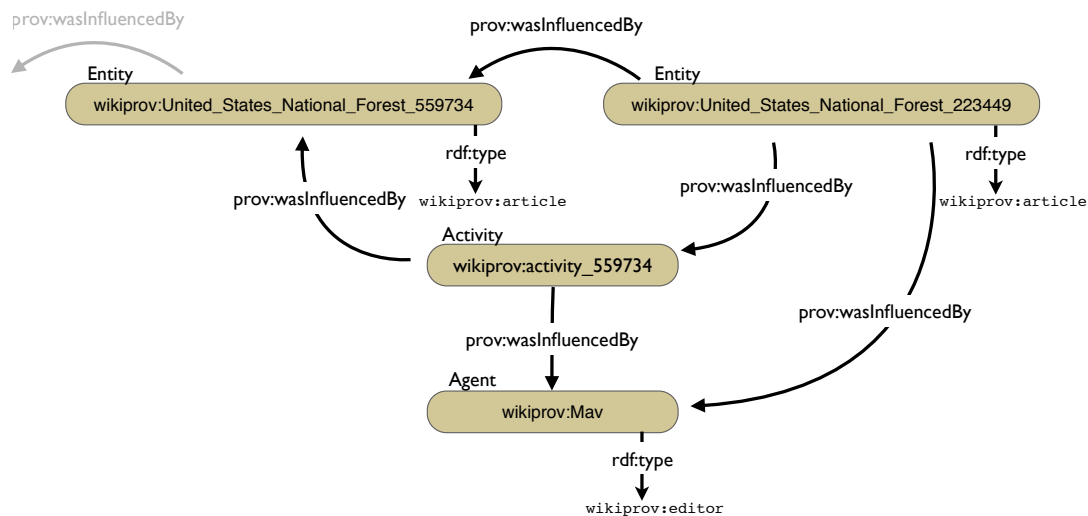


Figure 5.3: Example of a Provenance Graph from ProvBench Wikipedia using `wasInfluencedBy`

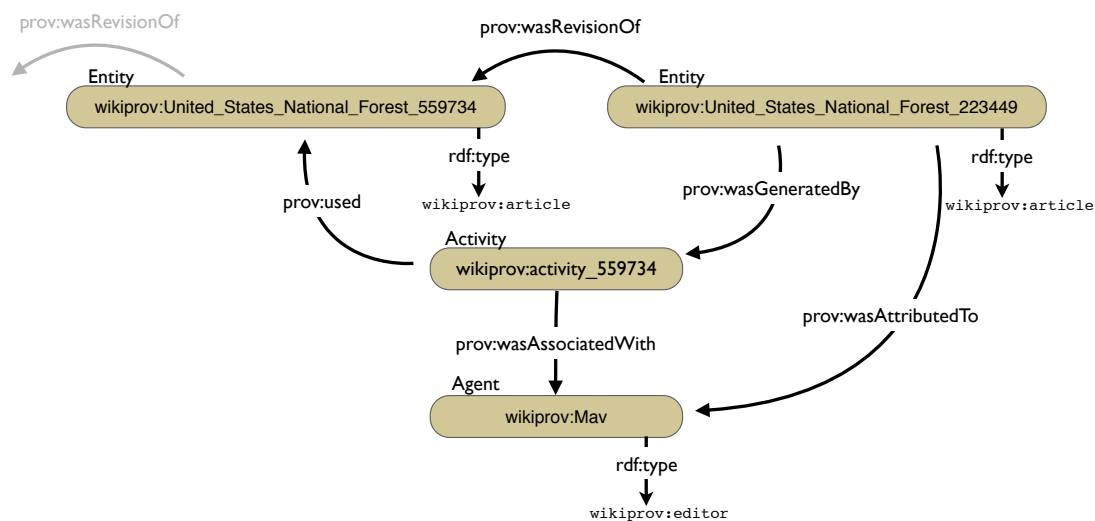


Figure 5.4: Example of a Provenance Graph from ProvBench Wikipedia using sub-properties of `wasInfluencedBy`

the object is the *influencer*. Figure 5.3 shows an example provenance graph for a Wikipedia revision from the wikipedia-provenance dataset that captures the influence relations between 4 provenance elements using the `wasInfluenceBy` relation. The dataset describes that the production of a `wikiprov:article` (`United_States_Forest_223449`) is influenced by one entity, one activity and one agent.

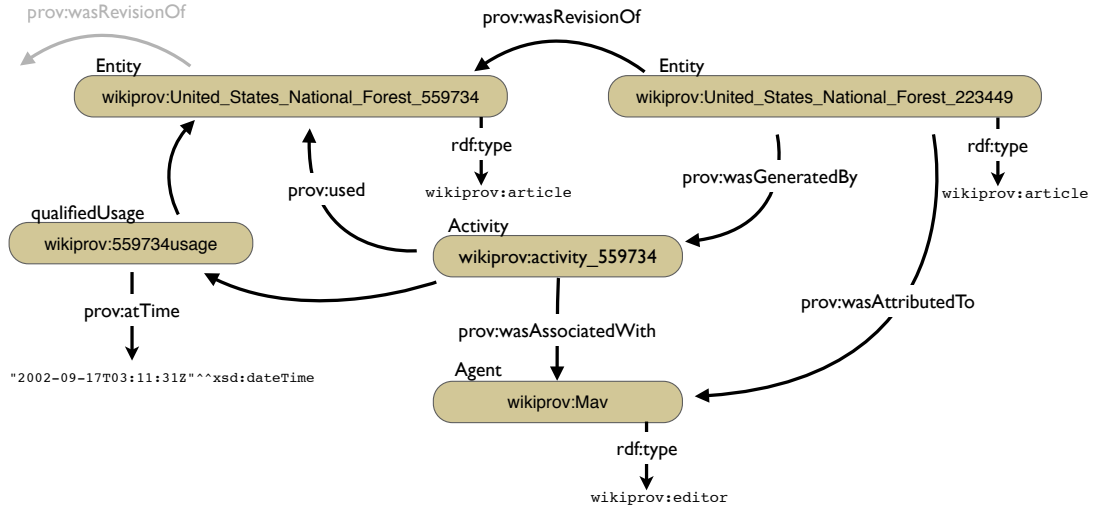


Figure 5.5: Example of a Provenance Graph from ProvBench Wikipedia using qualifiedGeneration

Using just the `wasInfluencedBy` relation is sufficient to capture lineage provenance by stating that there was an influencing relationship. It does not however capture any information about the nature of that relationship.

How provenance is captured in the PROV model in two ways: 1) using sub types of `wasInfluencedBy` and 2) qualified influences. PROV-DM provides 13 sub properties of `wasInfluencedBy` to better describe *how* the influencer influenced the influencee. Figure 5.4 illustrates an alternative modelling the wikipedia-provenance example, this time using the available sub properties of `wasInfluencedBy` so we can see more clearly the type of influence that each Web resource had.

To understand the how provenance publishers use PROV to describe influence we consulted the ProvBench datasets. For each dataset we have summarised the usage of the 13 sub classes of `wasInfluencedBy` based upon an analysis of the datasets and information from their supporting publications¹. The findings suggest that publishers of provenance information are willing to provide information beyond a simple `wasInfluencedBy` relation and describe *how* the influencer affected the influencee. Indeed whilst there are currently PROV features that are not used in the ProvBench data, it is still the case that *all* datasets make use of between 3 and 7 of the more specific influence properties of PROV to describe

¹For datasets that were not provided explicitly in PROV we consulted information from their supporting publication only

PROV Concept	Prov-O Vocab	ProvBench Usage	In Scope
PROV-DM Types			
Entity	prov:Entity	9/9	✓
Activity	prov:Activity	9/9	✓
Agent	prov:Agent	8/9	✓
PROV-DM Relations			
Generation	prov:WasGeneratedBy	9/9	✓
Usage	prov:Used	9/9	✓
Communication	prov:WasInformedBy	4/9	✓
Derivation	prov:WasDerivedFrom	6/9	✓
Attribution	prov:WasAttributedTo	4/9	✓
Association	prov:WasAssociatedWith	7/9	✓
Delegation	prov:ActedOnBehalfOf	1/9	✓

Table 5.1: The Core Elements of the PROV Data Model

influence.

In addition to sub properties of `wasInfluencedBy` PROV also provides *qualified influences*. Qualified influences use an N-ary relation to provide more detail descriptions for influence relations. Figure 5.5 shows our wikipedia-provenance example this time with a `qualifiedUsage` providing additional information about the `prov:usage` relation, in this case the time of that usage using the property `prov:atTime`. Each influence type has a corresponding qualified influence in PROV. Along with `atTime`, PROV provides `hadRole` and `hadPlan` that can be used to enrich qualified influences.

Despite these qualified influences and additional properties we are still missing some necessary data to make our provenance based quality assessment. In order to model the quality of the chembox as conditional on the Wikipedia page that it was derived from, we required a quantitative measure of *how* provenance. The information that we are missing is *to what extent* did the influencer contribute to the influencee.

PROV Plans provide detailed and specific information about a `qualifiedAssociation`, and might therefore provide this quantitative information. However for our purposes they have number of limitations. Firstly they are restricted by the PROV specification to *only* be used with `qualifiedAssociations`, we instead want to be able to quantify any influencing relationship. Secondly PROV Plans are not restricted by the PROV specification in their representation. As a result, whilst they may provide the quantitative information we require, they

might not be in a known representation, or even a machine readable representation. Thus we see the need for a vocabulary feature to enrich any qualified influence and provide the quantitative *how* provenance.

				Datasets								
Influencee	Property	Influencer	In Scope	Taverna	Wings	Wiki	SRI	OBAMIO	CSIRO	Vis	Chiron	Swift
*	wasInfluencedBy	*	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Entity	wasGeneratedBy	Activity	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Entity	wasDerivedFrom	Entity	✓	✓	×	×	✓	✓	✓	×	✓	✓
Entity	wasAttributedTo	Agent	✓	×	✓	×	×	✓	✓	×	✓	×
Entity	hadPrimarySource	Entity	✓	×	✓	×	×	×	×	×	×	×
Entity	wasQuotedFrom	Entity	✓	×	×	×	×	×	×	×	×	×
Entity	wasRevisionOf	Entity	✓	×	×	✓	×	×	×	×	×	×
Entity	wasInvalidatedBy	Activity	×	×	×	×	×	×	×	×	×	×
Activity	wasInformedBy	Activity	✓	✓	×	×	✓	✓	✓	×	✓	×
Activity	used	Entity	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Activity	wasAssociatedWith	Agent	✓	✓	✓	✓	✓	✓	✓	✓	✓	×
Activity	wasStartedBy	Entity	✓	×	×	×	×	×	×	×	×	×
Activity	wasEndedBy	Entity	✓	×	×	×	×	×	×	×	×	×
Agent	actedOnBehalfOf	Agent	✓	×	×	×	×	×	✓	×	×	✓

Table 5.2: Summary of PROV Influence Types in ProvBench

5.4 Influence Factor

We define influence factor as a *quantitative measure* of the influence that one PROV entity, agent, or activity has had over another. To automatically reason about how the quality of one provenance element informs another in our provenance data, the specific aspect of *how* provenance we require is the degree of influence, or *influence factor*. This information can be used to subsequently determine the quality or trustworthiness of that PROV element in terms of its influencers. This degree of influence is currently suggested with certain properties of the PROV vocabulary such as `wasQuotedFrom`, `wasGeneratedBy` and `hadPrimarySource`. With influence factor we are making this explicit.

For example if an Activity generated an Entity, declared by `wasGeneratedBy`, and it is the only influencer described, then we might make the assumption that it had exclusive influence. There are however cases where the metadata describing the degree of influence falls short and as such makes the ability to reason automatically about quality difficult. For example, for two revisions of the same Wikipedia page the `prov:wasRevisionOf` relation between the two entities and the `qualifiedRevision` description falls short of fully describing the relationship between the two, specifically how much of the previous revision has remained. This is similarly the case for the `qualifiedAttribution`. If the author is not considered trustworthy then our belief in the likely quality of the resulting revision will differ depending on, for example, whether they have modified the whole page, or just contributed to a small part of it.

In many cases, in the production of data it is possible to quantify this degree of influence. A mechanism for quantifying the difference between two revisions of an article in Wikipedia is a ‘diff’ between the two. A quantitative measure of this diff can be easily included as additional metadata.

Influence factor might not just reflect a physical attributes such as the diff. We might believe that some parts of the Wikipage are more important than others, for example Infobox data. Instead of an influence measure based on the size of contribution, we might weight it by where in the article that contribution is made.

A further example comes from the scholarly communications domain. When describing the creation of a scholarly artifact we can describe and attribute that creation to one or more creators indicating their role such as lead author, contributor, supervisor. Vocabularies for scholarly communications such as the Semantic

Publishing and Referencing Ontologies (SPAR) suite of ontologies [Sho10] capture this type of contribution description in the Publishing Roles (PRO) ontology, using classes such as editor, contributor, copy-editor etc. Whilst these categories of contribution are not numerical, they provide a spectrum of influence to which we can apply our own consistent weighting. Given these observations we believe that a mechanism for describing a *degree of influence* would increase the ability of the PROV vocabulary in its stated purpose to support the assessment of the quality, reliability and trustworthiness of data.

5.4.1 Modelling evident:influenceFactor

To capture the degree of influence one entity has on another we have introduced `evident:influencefactor` as an additional property in our evident namespace as an attribute for any of the PROV qualified influences. The property allows the provision of additional information quantifying the degree to which the influencing class has influenced the influenced class.

We have extended the `evident:influenceFactor` property to model two core types of influence factor, `evident:discreteInfluenceFactor` and `evident:continuousInfluenceFactor`. `evident:discreteInfluenceFactor` can be extended to model discrete states of influence such as those from the SPAR vocabularies.

Using `evident:continuousInfluenceFactor` we can model influence factor using a continuous numerical value. A subproperty of `evident:continuousInfluenceFactor` that we include as part of our extension is `evident:normalInfluenceFactor`. This property describes a degree of influence as a real number on a scale [0..1]. Figure 5.6 illustrates the use of the `evident:normalInfluenceFactor` for the ProvBench Wikipedia revisions data to enrich a qualified revision. To quantify the influence the influencer `wikiprov:United_States_National_Forest_559734` has had on the influencee `wikiprov:United_States_National_Forest_223449` we add the influence factor to the qualified revision description between the two. We introduce two terms relating to influences that have been enriched with an influence factor.

- ***quantified influence***: To distinguish between a qualified influence that has been enriched with an influence factor and one that has not, we refer to a qualified influence that has been enriched as a *quantified influence*.

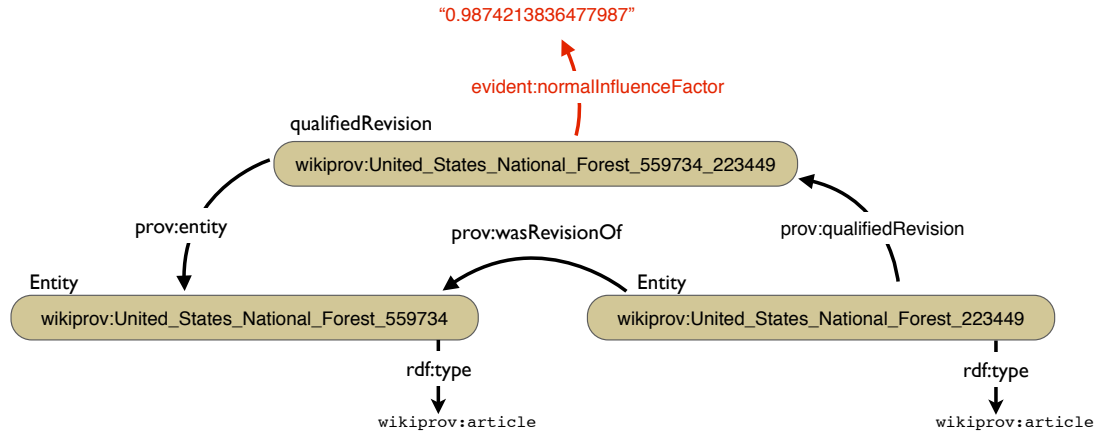


Figure 5.6: Using evident:influenceFactor in Wikipedia Provenance

- **quantified path:** We refer to any transitive path of influences between two entities in a graph such that at least one of the influences is quantified as a *quantified path*.

In the case of `normalInfluenceFactor` one might expect that by modelling influence factor on a scale of $[0...1]$ we should modify the conditions for provenance validity such that the sum of all influence factors that directly influence a given element should sum to 1. However we believe that in a distributed publishing environment such as the Web of Data such a restriction would be prohibitively difficult for a data publisher to comply with. Instead we leave it to the consumer of the provenance to evaluate, and if needed normalize any influence factors for a given entity. Section 5.5.4 illustrates an example of such a normalization strategy in the context of our IQ assessment.

As with many modelling approaches there is scope for human error when describing influence factor. One particular scenario we highlight is a case we define as *overstating influence*. This refers to a modeller attributing the same conceptual influence from one entity, agent or activity to more than one qualified influence. Consider the Wikipedia revision in Figure 5.7. To quantitatively capture the influence that the author agent had in the revision, the modeller has to decide where to describe the influence factor. The modeller could quantify either the qualified attribution between the author and revision entity, or the qualified generation between the activity and revision entity. The modelling approach shown in figure 5.7 would constitute overstating influence, where the same conceptual contribution to the revision is duplicated by quantifying both

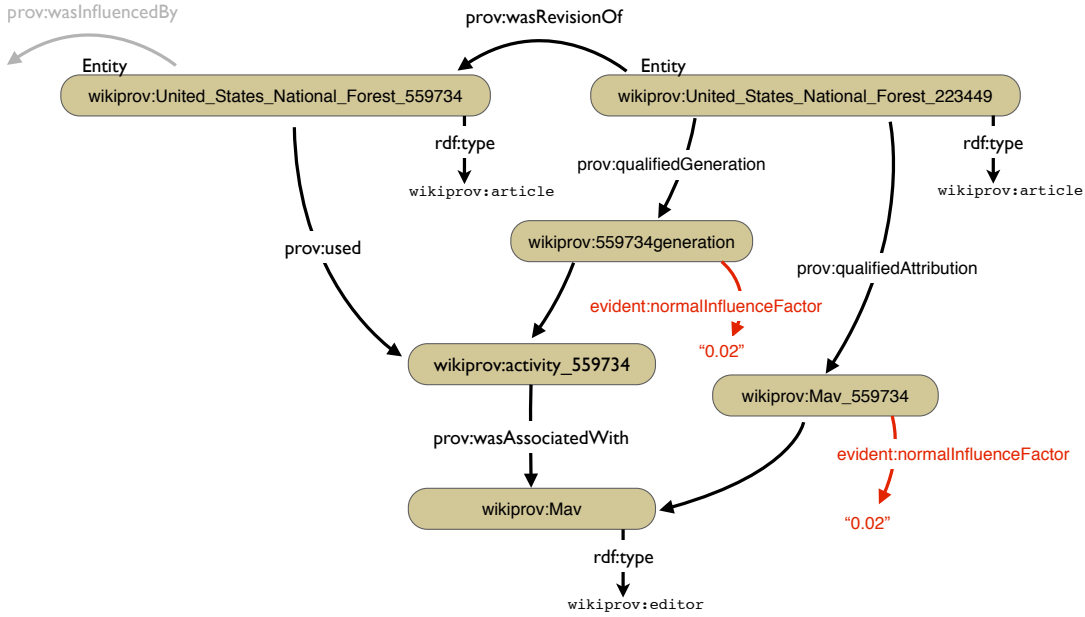


Figure 5.7: Overstating `evident:normalInfluenceFactor` in Wikipedia Provenance

the qualified generation and qualified attribution. The overstating of influence is difficult to account for retrospectively by a data consumer because its occurrence is ambiguous. Given the example we would not know for example if the two quantified paths between author and entity captured the same conceptual influence, or two unrelated types of influence. For the purposes of our procedure we make the assumption that if influence factor is explicitly stated, then it is correctly stated.

With the addition of `evident:influenceFactor` we now have sufficient provenance metadata to inform the automatic generation of Quality Fragments.

5.5 Quality Fragment Generation Procedure

In this section we describe the procedure we have developed to generate quality fragments. We divide the description into three concerns:

1. A procedure to generate the Quality Fragments network structure (Section 5.5.2)
2. A Combination Strategy to create CPDs for resident nodes in the network from 1 (Section 5.5.3).

3. A strategy to normalize influence factor when using the Quality Fragments generated by 1 and 2 to create an SSBN (Section 5.5.4).

First we describe some general assumptions and restrictions with respect to the input data to our procedure.

5.5.1 Assumptions and Restrictions

The inputs to the Generation Procedure are:

- A PROV provenance graph Pr that describes the provenance of a set of Web resources E .
- A set of existing Quality Fragments Q .
- An entity $e_i \in E$ for which we wish to know the likely quality.

The Quality Fragment generation procedure assumes the following restrictions regarding the input:

- The Prov graph Pr is valid with respect to the PROV-Constraints.
- $t_1 = T_{e_i} \wedge t_2 = T_{e_i} \rightarrow t_1 = t_2$, $T_{e_i} \neq \emptyset$ i.e Each Web resource has exactly one corresponding type.
- $q_1 = Q_{t_i} \wedge q_2 = Q_{t_i} \rightarrow q_1 = q_2$ i.e. Each type has *at most* one existing Quality Fragment.
- If influence factor has been described between two Web resources in Pr then it is of the type `normalInfluenceFactor` and declared on a scale $[0...1]$.
- The result of each existing Quality Fragment in Q is a quality score on a scale $[0...1]$.

Although we have designed and developed `evident:influenceFactor` to support our procedure, the Quality Fragment Generation Procedure does *not* make the assumption that influence factors have been declared. Section 5.5.4 describes how we manage different levels of influence factor description.

5.5.2 Structure Generation

Here we describe the procedure that generates the Quality Fragment structure using the *lineage* provenance in the provenance graph. We begin with a general overview of the procedure and follow with a detailed walk through using the example from Figure 5.1.

Overview

The procedure **Generate** presented in Algorithm 2 (on page 211) is at a high-level a recursive depth-first tree walk of the PROV graph. The walk begins at the entity e_i that we want to assess the quality of, and initially creates an empty Fragment to represent it. The procedure then examines each influencing Web resource e_j defined in P_{e_i} . For each influencing Web resource we perform one of two actions:

- If we find an *existing* Fragment in Q_{e_j} for the influencing Web resource then we link it to the Fragment for e_i using an input node.
- If we do not find an existing Fragment for e_j then we create a *new* empty one, link it to the Fragment for e_i , and recursively repeat the procedure along the path, using the new Fragment and inspecting e_j 's influencing Web resources.

The procedure continues to walk along the PROV graph linking together new and existing Quality Fragments.

The procedure terminates when we have either visited each influencing Web resource reachable on a path from e_i , or have found an existing Quality Fragment for a Web resource on each path.

In Detail

To illustrate the procedure concretely we reuse the example in Figure 5.8 of a chembox, this time updated to include influence factor, and annotated to highlight stages of the procedure.

The procedure begins on line 2 of Algorithm 2. The initial function **Initialize**($e_{current} \in E$) initializes the procedure and covers the special case that we already have a Quality Fragment in Q to assess $e_{current}$. In the annotated example $e_{current}$ corresponds to the **chembox:Ethane** Web resource (a.). The **if**

statement on line 6 checks if Q contains a Quality Fragment for $e_{current}$'s type $t_{current}$. If we do have an existing Quality Fragment for $t_{current}$ then no further generation is required and we return that Quality Fragment.

If we do not have an existing Quality Fragment in Q for the type $t_{current}$ then we initialize one. Figure 5.9 shows the state of the Fragment from our example after **Initialize**. To do this, we create an empty Fragment on line 9, in our example this is the *Chembox Quality Fragment*. We then add to the empty Fragment a resident node that represents the quality of $e_{current}$ (b.) and a context node that constrains the type to $t_{current}$ (c.).

Once we have initialized this new Fragment we then begin the recursive phase of the procedure **Generate**, which examines the provenance graph and existing Fragments. **Generate** takes three arguments:

- $e_{current}$ - the current Web resource we are examining in the provenance graph (a.).
- $q_{current}$ - the Quality Fragment for that the Web resource.
- $path$ which maintains the path to the current node $e_{current}$, from the original web resource that started the procedure e_{root} . This is tracked so that we can prioritize quantified paths.

The **for** loop that begins on line 18 examines each of the Web resources that are declared in the provenance graph as influencers of $e_{current}$. In our example this is just the **wiki:Ethane** Web resource (d.). For each influencing Web resource we either check to see if we have an existing Quality Fragment, or initialize a new one as before, and assign it to $q_{influencer}$. In our example we would discover the *WikiArticle Quality Fragment* (e.).

The final stage of the **Generate** procedure is to link together $q_{current}$ and $q_{influencer}$ using input nodes and context nodes. In our example line 26 creates the context node for a Wikipedia article (f.), line 27 creates the influence relationship context node (g.). Line 28 creates the input node (h.) to link the two Fragments together and Line 29 creates the edge to model that the Chembox Quality is conditional on the WikiArticle Quality (i.). Lines 30 and 31 also introduce a Resident Node (j) and Edge (k.) to model the InfluenceFactor of the WikiArticle.

If we did not find an existing Fragment to assess the quality of $e_{influencer}$, but instead just initialized a new one during this execution, then we do not currently

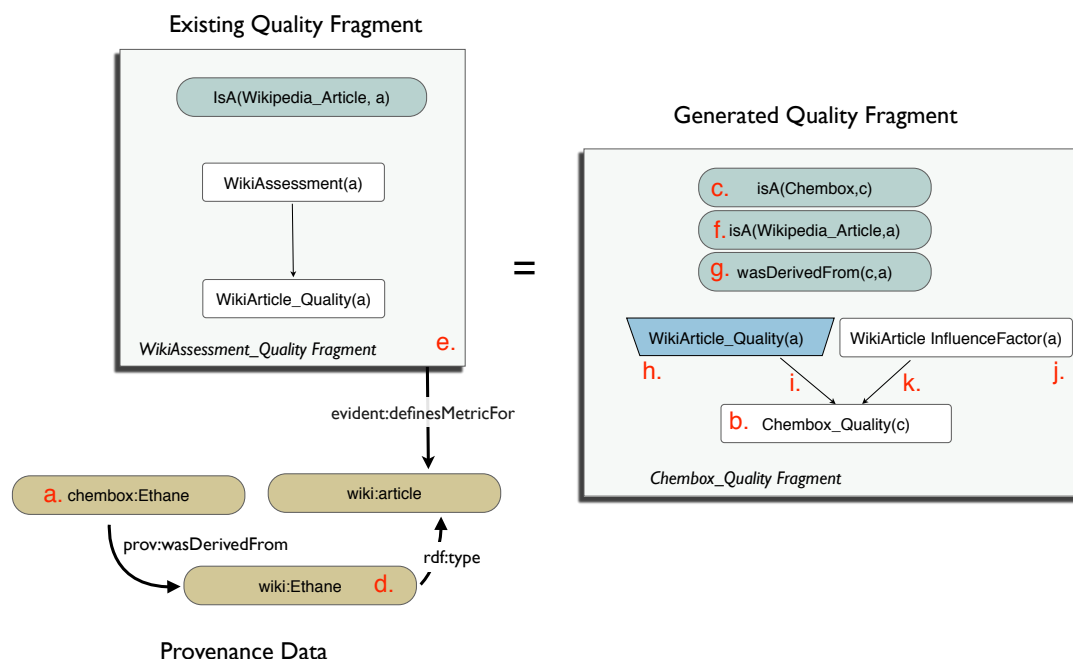


Figure 5.8: Quality Fragment Generation for Chembox using Quality Fragment Generation Procedure.

have an existing intrinsic assessment along this path. We therefore keep walking along the provenance graph depth first searching for an existing Fragment that can inform $q_{influencer}$. This is done on line 33 by passing $e_{influencer}$, and $q_{influencer}$ to the **Generate** procedure.

The **if** condition on line 19 is a strategy to ensure that we prioritise quantified paths if they are available. This is to avoid considering influence from the same Web resource more than once, unless it explicitly stated.

Once we have built the structure for the Fragments we must populate the CPDs for the resident nodes we have created, in our example these are the *Chembox_Quality* resident node (b.) and the influence factor resident node (j.) The *Chembox_Quality* CPD is created using the Combination Strategy defined in section 5.5.3 below, the influence factor values are determined during SSBN creation and described in section 5.5.4.

5.5.3 Combination Strategy

The second modelling task required is to create the CPD representations for the newly generated resident nodes that are informed by input nodes, such

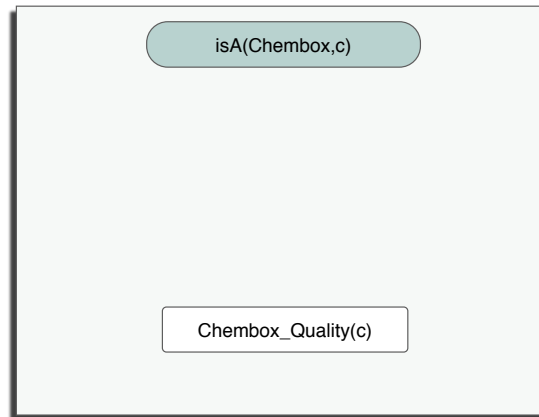


Figure 5.9: Quality Fragment Generation for Chembox after Initialization.

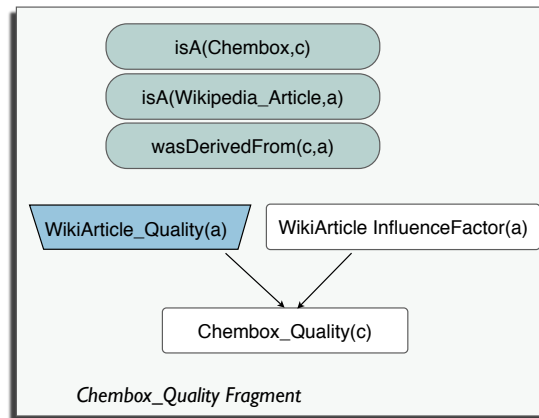


Figure 5.10: Quality Fragment Generation for Chembox after First Pass of Generation.

as *Chembox_Quality(c)*. The task of the CPD is to combine the values from any influencing configurations into one continuous value. Figure 5.12 illustrates this with an SSBN generated using the *Chembox_Quality* Fragment and the provenance graph in figure 5.12. The chembox in the example has been derived from two Wikipedia articles *wiki:a1* and *wiki:a2*. The example SSBN therefore has two influencing configurations of the *WikiArticle_Quality* Quality Fragment, *WikiArticle_Quality(a1)* and *WikiArticle_Quality(a2)*, each with a corresponding influenceFactor resident node informing *Chembox_Quality(c1)*. SSBNs created from our generated Quality Fragments will follow this general pattern, where each influencing configuration that represents a quality value

for an influencer e.g. *WikiArticle_Quality(a2)*, will have a corresponding influence factor resident node e.g. *WikiArticle_InfluenceFactor(a1)*. The CPD for *Chembox_Quality(c1)* must therefore combine each of these pairs into one continuous value. The combination strategy we apply is a sum of each influencing configuration's quality value, weighted by its influence factor. For the Chembox example this would be:

$$\begin{aligned} & \text{Chembox_Quality}(c1 \mid \text{WikiArticle_Quality}(a1), \\ & \quad \text{WikiArticle_Quality}(a2)) = \\ & \quad [\text{WikiArticle_Quality}(a1) \times 0.4] + \\ & \quad [\text{WikiArticle_Quality}(a2) \times 0.6] \end{aligned}$$

More generally the combination strategy is defined as follows. Let C be the set of influencing configurations indexed by E , and I the set of influence factors as defined previously. The combination strategy for the resident node $q(e_i)$ given the influencing configurations $[C_{e_j}, C_{e_{j+1}}, \dots, C_{e_n}]$ is computed as:

$$q(e_i \mid C_{e_j}, C_{e_{j+1}}, \dots, C_{e_n}) = [C_{e_j} \times I_{e_i, e_j}] + [C_{e_{j+1}} \times I_{e_j+1, e_i}] + \dots + [C_{e_n} \times I_{e_n, e_i}] \quad (5.1)$$

Our combination strategy makes two assumptions

1. Each influencing Web resource is treated as independent.
2. Each influencing Web resource has defined an influence factor.

To ensure 2 at SSBN generation time we make use of the Influence Factor Normalization Strategy described in the next section.

5.5.4 Influence Factor Normalization

The final part of our procedure concerns the values of the influence factor resident nodes. The influence factor resident nodes values are determined at SSBN creation time when we are using the Fragment to assess some new data. This is because it is only at the SSBN generation stage that we know which influence factors have been declared and which have not. We introduce a

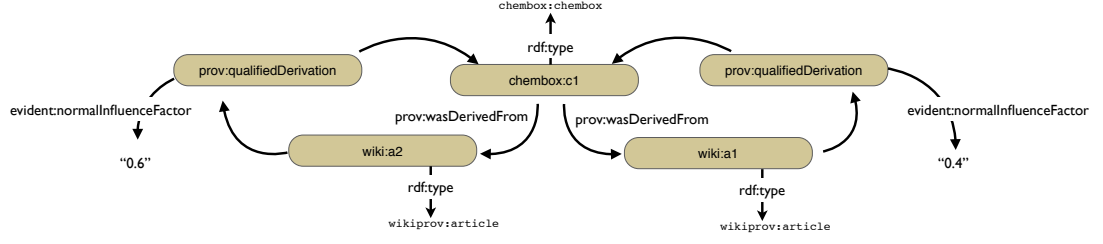


Figure 5.11: Provenance graph for Chembox Derived From two Wikipedia Articles.

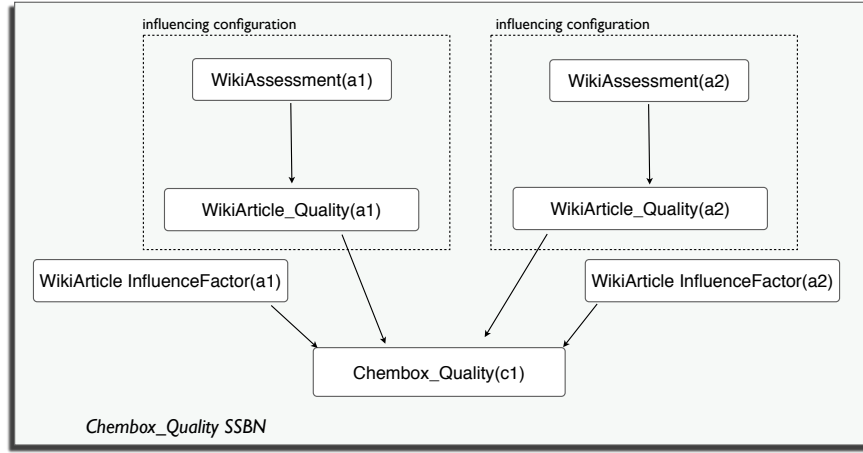


Figure 5.12: SSBN for Chembox derived from two Wikipedia Articles.

strategy to ensure each influence factor has a value. In our example in figure 5.12 both influencers have defined influence factor values, and the two influence factors sum to 1. There may be instances where this is not the case.

Let $\sum_{i=0}^n I_e^i$ be the sum of all defined influence factors for the entity e . We identify three possible scenarios and corresponding strategies to adjust *defined/undefined* influence factors:

$$\sum_{i=0}^n I_e^i = 1 \implies \text{do nothing/set to zero} \quad (5.2)$$

$$\sum_{i=0}^n I_e^i > 1 \implies \text{normalize/set to zero} \quad (5.3)$$

$$\sum_{i=0}^n I_e^i < 1 \implies \text{do nothing/redistribute} \quad (5.4)$$

In case 5.2 where all influence factors sum to 1 then we do not need to make any adjustments to defined influence factors and we set undefined influence factors to zero.

In case 5.3 where the influence factors defined are greater than 1, we normalize each defined influence factor as:

$$I_{e_i, e_j} \times 1 / \sum_{i=0}^n I_e^i$$

Finally where the defined influence factors sum less than 1 and there are influence factors undefined then we redistribute the remaining influence factor amongst the undefined as:

$$(1 - \sum_{i=0}^n I_e^i) / |undefined|$$

These three cases ensure that all influencing Web resources have a corresponding influence factor. This is just one possible strategy and we see the analysis of alternative normalization approaches as useful future work (see section 5.8).

In the next section we demonstrate and evaluate our approach using a modified version of the ProvBench Wikipedia data. We have also extended the Evident framework in order to support our evaluation.

Algorithm 2: Pseudocode for Quality Fragment Generation Procedure

Data: PROV graph Pr , Set of Quality Fragments Q , Set of Web Resources e , Set of types T

```

1  $e_{root} \leftarrow \emptyset$  ;
2 def void Initialize( $e_{current} \in E$ ):
3    $q_{current} \leftarrow \emptyset$ ;
4    $t_{current} \leftarrow T_{e_{current}}$ ;
5    $e_{root} \leftarrow e_{current}$  ;
6   if  $Q$  contains existing Quality Fragment that definesMetricFor  $t_{current}$ 
   then
7     return  $Q_{t_{current}}$ ;
8   else
9      $q_{current} \leftarrow$  new Quality Fragment for  $t_{current}$ ;
10    createResidentNode " $q_{current}(e_{current})$ " in  $q_{current}$ ;
11    createContextNode " $isA(e_{current}, t_{current})$ " in  $q_{current}$ ;
12    Generate( $e_{current}, q_{current}, \emptyset$ );
13 def void Generate( $e_{current} \in E, q_{current} \in Q, path \subset E$ ):
14    $q_{influencer} \leftarrow \emptyset$ ;
15    $t_{influencer} \leftarrow \emptyset$ ;
16    $path \leftarrow path \cup e_{current}$ ;
17    $Q \leftarrow Q \cup q_{current}$  ;
18   for Web resource  $e_{influencer}$  in  $P$  that influences  $e_{current}$  do
19     if  $path$  is a quantified path or no alternative quantified path exists
       between  $e_{influencer}$  and  $e_{root}$  then
20        $t_{influencer} \leftarrow T_{e_{influencer}}$ ;
21       if  $Q$  contains existing Quality Fragment that definesMetricFor
         for  $t_{influencer}$  then
22          $q_{influencer} \leftarrow Q_{t_{influencer}}$ ;
23       else
24          $q_{influencer} \leftarrow$  new Quality Fragment for  $t_{influencer}$ ;
25         createResidentNode " $q_{influencer}(e_{influencer})$ " in  $q_{influencer}$ ;
26         createContextNode " $isA(e_{influencer}, t_{influencer})$ " in  $q_{current}$ ;
27         createContextNode " $wasInfluencedBy(e_{current}, e_{influencer})$ " in
            $q_{current}$ ;
28         createInputNode " $q_{influencer}(e_{influencer})$ " in  $q_{current}$ ;
29         createEdge from " $q_{influencer}(e_{influencer})$ " to " $q_{current}(e_{current})$ " ;
30         createResidentNode " $influenceFactor(e_{influencer})$ " in  $q_{current}$ ;
31         createEdge from " $influenceFactor(e_{influencer})$ " to
           " $q_{current}(e_{current})$ " ;
32         if  $q_{influencer}$  was created during this execution then
33         | Generate( $e_{influencer}, q_{influencer}, e_{root}, path$ );
34       else
35         return ;
36   end

```

5.6 Evaluation

Our evaluation is in two stages. The first stage is to evaluate our ability to generate Fragments from provenance data. The second is to assess how well a generated Fragment can assess new provenance of the same data type.

To perform this evaluation we have designed an experiment to replicate an existing study by Zeng et al. [ZAFM06] that uses a Bayesian Network to evaluate the quality of Wikipedia article revisions. The goal is to automatically generate a network using our procedure, and compare assessment results from this network against the manually created Bayesian Network in Zeng.

We generate a new Fragment to assess the quality of a Wikipedia article revisions using PROV provenance data for a Wikipedia article. We reuse this generated Fragment to assess the quality of Wikipedia articles from their provenance data.

By using Wikipedia we have a widely recognized ground truth for the quality of a Wikipedia article. It is common in the Wikipedia research community to use the article statuses assigned by Wikipedia administrators; *featured*, *cleanup* or *normal* as a ground truth for their quality:

- *Featured* articles are considered high quality by the Wikipedia community because they have been reviewed for style, completeness, accuracy and neutrality.
- *Cleanup* articles are considered poor quality and have been indicated as needing major revision.
- *Normal* articles are the remaining articles that are neither featured, nor clean-up.

We compare our Fragment with the Bayesian Network provided in Zeng et al. in two respects:

1. Treat the two solutions as “black boxes” and compare the results that each network provides for the assessment of Wikipedia revisions. Zeng use their network and a set of training data to establish a threshold to classify featured and cleanup articles given an assessment value on a scale [0...1]. Our aim is to compare the ability of our automatically generated Quality Fragment to Zeng in classifying featured and cleanup articles.

2. Compare the networks themselves. This is useful to understand from a Quality Knowledge Engineering perspective: how they might be reused, adapted, maintained.

We have taken a number of steps in the design of our evaluation to ensure that we can meaningfully compare our automatically generated Fragment with the network presented in Zeng. We have replicated the part of the Zeng network that represents the intrinsic trustworthiness of an author. This is modelled as a Fragment and used as input to our procedure. The modelling for this is described in section 5.6.1. Like Zeng we have chosen Wikipedia articles from the Geography category of Wikipedia to generate provenance data. The preparation of this provenance data is discussed in section 5.6.2

Our evaluation is presented as follows. We introduce the Zeng et al. study and describe the Bayesian Network used by the authors. We then describe the steps taken to prepare the data for our evaluation, creating an RDF version of the wikipedia-provenance ProvBench data. Next we describe the extensions to the Evident framework and Evident vocabulary required to perform our evaluation. Finally we present the results of our evaluation, we discuss the automatically generated network for evaluating the quality of Wikipedia articles, and compare the results of our network with the results provided by the Zeng study.

5.6.1 The Zeng Network

The study of Zeng et al. employed a manually built Bayesian Network to assess the quality of Wikipedia articles. The network uses evidence about the author of a revision and the amount of text inserted and deleted to provide a quality score for the revision of a Wikipedia article on a scale [0..1]. Figure 5.13 shows the Bayesian Network developed by Zeng. The network is a type called a *Dynamic* Bayesian Network. Dynamic Bayesian Networks perform their evaluation in an iterative manner, where information from the current step of the evaluation can propagate to the next. The dotted arrows illustrate how the information propagates.

The state of the i^{th} iteration of the network is represented as a quad of variables $(t_{A_i}, t_{V_i}, i_i, d_i)$. There is an iteration of the network per revision of the article being assessed, therefore the i^{th} iteration represents the i^{th} article

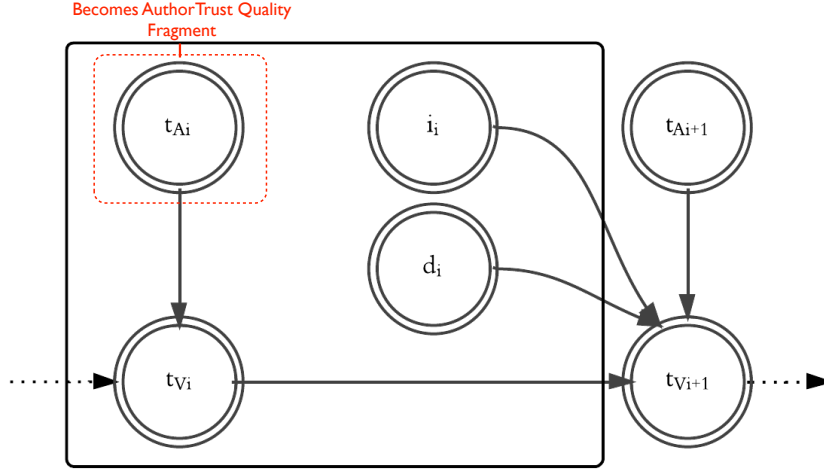


Figure 5.13: Dynamic Bayesian Network from Zeng et al. to Estimate Quality of Wikipedia Revisions

revision. The variables represent the following:

- t_{Vi} represents the quality of the i^{th} revision on a scale $[0...1]$.
- t_{Ai} represents the trustworthiness of the author A_i of the i^{th} revision on a scale $[0...1]$.
- i_i and d_i represent the number of words inserted and deleted by author A_{i+1} to the current revision i to create the revision $i + 1$.

To calculate the words inserted and deleted Zeng et al. use a word-based diff based on the *longest common subsequence* algorithm. The authors do not specify exactly which diff implementation is used.

To replicate the assessment provided by Zeng we need to model the Quality Knowledge for the trustworthiness of the author as an existing Fragment. This is because the trustworthiness of the author is intrinsic, and not based on provenance information. To represent the possible states of the author random variable t_{Ai} , Zeng partitions authors into four types defined by their editing privileges: Administrator, Registered, Anonymous and Blocked.

Zeng uses a Beta Distribution to approximate the trustworthiness of each type of Wikipedia Author. The CPT for t_{Ai} is shown in table 5.3. The value for t_{Ai} is the mean of the corresponding beta distribution. The distributions have been chosen by Zeng to intuitively represent the relative trustworthiness

Author Type of A_i	t_{A_i}
Administrator	$Beta(190, 10)$
Registered	$Beta(23, 10)$
Anonymous	$Beta(15, 10)$
Blocked	$Beta(10, 190)$

Table 5.3: t_{A_i} CPT

Author Type	isBlocked	AuthorTrust
Administrator	false	$Beta(190, 10)$
Registered	false	$Beta(23, 10)$
Anonymous	false	$Beta(15, 10)$
Administrator	true	$Beta(10, 190)$
Registered	true	$Beta(10, 190)$
Anonymous	true	$Beta(10, 190)$

Table 5.4: AuthorTrust CPT

placed in each type of author. The distributions result in means of 0.95 for Administrators, 0.7 for Registered authors, 0.6 for Anonymous authors and 0.05 for Blocked authors.

We have replicated t_{A_i} by creating the AuthorTrust Quality Fragment shown in Figure 5.14. The CPT for $AuthorTrust(author)$ is shown in Table 5.4. In our modelling of $AuthorTrust$ the trustworthiness of the author is based upon two pieces of metadata, the type of author [Administrator, Registered, Anonymous], and whether the author is blocked [true, false]. We can see by comparing the two CPTs in Tables 5.3 and 5.4 that the result for each type of author is the same.

The probability distribution for the quality of a revision $t_{v_{i+1}}$ is less straight forward. Equation 5.5 shows the function used to determine the value for $t_{v_{i+1}}$ in the network.

$$t_{v_{i+1}} = \{t|V_i| + \alpha_{i+1}|I| - \min((1 - \alpha_{i+1})|D_i|, t|V_i|) - \max(\alpha_{i+1}|D_i| - (1 - t)|V_i|, 0)\} / V_{i+1} \quad (5.5)$$

We shall not discuss in detail the workings of the equation but include it here in order to illustrate the complexity of the CPD, and serve as further motivation for automatically generating Fragments and CPDs. Indeed Zeng et al. discuss a series of four detailed insertion and deletion scenarios that have

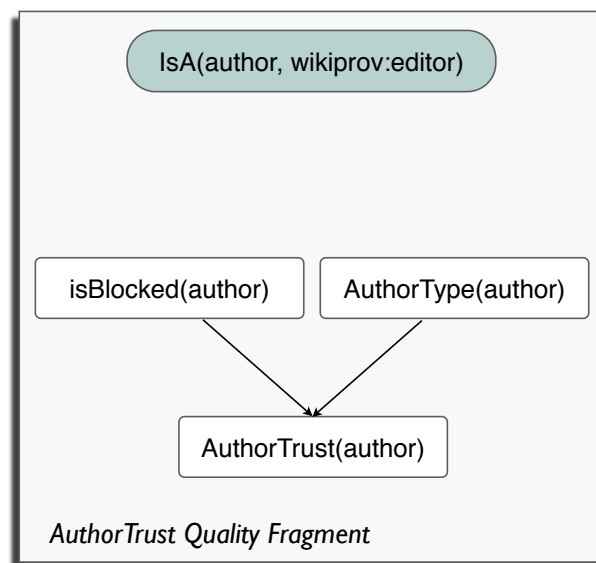


Figure 5.14: Author Trust Quality Fragment

been considered in the process of modelling of the CPD.

Zeng et al. used the Bayesian Network described to assess the quality of 868 Wikipedia articles collected from the Geography category of Wikipedia in January 2006². Of these 868 articles 50 were featured articles, 50 were cleanup articles and 768 were normal articles. The authors use more normal articles than cleanup or featured articles, because only 0.1% of Wikipedia articles are featured and only 1.3% of Wikipedia articles are clean-up articles. Table 5.5 gives the results from Zeng showing the average value of t_{V_i} for the final revision of each type of article.

	Featured articles	Clean-up articles	Normal articles
Average $t_{V_{final}}$	0.885	0.768	0.808

Table 5.5: Results from the Zeng Network

Zeng use their results to train a classifier to predict featured and cleanup articles. The authors train the classifier with 100 pairs (x, y) where x is the calculated quality value of the article revision and y is the class, featured or cleanup.

In this training set the assessment value is not taken from the final article revision, but instead from the article revision where the article was either first

²Zeng et al. do not disclose in what format the data was stored.

promoted to featured status, or assigned cleanup status. This is based upon the assumption that it is this article revision to which the ground truth applies, and any future revisions have not been verified.

The resulting classifier divides the classes such that:

- $t_{V_i} > 0.842$ is featured
- $t_{V_i} \leq 0.842$ is cleanup

Using the network to assess a test set of 200 new articles³ Zeng correctly predicts 82% of featured articles and 84% of cleanup articles based upon the above classification, with an overall successful classification of 83%. In our evaluation, we compare our results with Zeng by similarly creating a classifier to predict feature and cleanup articles.

5.6.2 Data Preparation

For our evaluation we have created a series of RDF PROV provenance graphs for Wikipedia articles:

- A graph for a Wikipedia article to generate a Quality Fragment.
- A training set of 500 graphs to evaluate using the generated Quality Fragment and train a classifier.
- A test set of 100 graphs to evaluate using the generated Quality Fragment and test a classifier.

To create our training dataset we have collected articles from the Geography category of Wikipedia. We have chosen the articles from the Geography category so that our results will be comparable to those of Zeng. For each Wikipedia article we have collected the data for each revision starting from the articles first revision up to the last revision on January 31st 2006. We have chosen this date to be comparable to the Zeng study. In particular we are accounting for the fact that the relative quality of a featured or clean-up article may have changed over time. Our data consists of 400 normal articles, 50 featured articles and 50 clean-up articles. Table 5.6 provides a summary of the statistics of the training dataset.

³Zeng et al. do not specify what types of articles make up this test set.

	featured	cleanup	normal
Average Triples	36857	6093	1597
Average Revisions	616	102	27
Average Administrators (%)	7.29%	13.02%	10.92%
Average Registered (%)	40.44%	52.33%	57.00%
Average Unregistered (%)	50.26%	34.66%	32.08%
Average Blocked (%)	1.70%	1.38%	1.19%
Average Final Revision Size (Words)	2179	789	354
Average Influence (%)	3.68%	6.30%	9.98%

Table 5.6: Statistics of the Training Dataset

	featured	cleanup
Average Triples	37760	5152
Average Revisions	623	85
Average Administrators (%)	15.94%	16.42%
Average Registered (%)	42.45%	50.65%
Average Unregistered (%)	41.61%	32.93%
Average Blocked (%)	2.95%	1.66%
Average Final Revision Size (Words)	5152	657
Average Influence (%)	2.74%	5.97%

Table 5.7: Statistics of the Test Dataset

To create our test set data we collected a further 100 articles this time in the Biology category from January 31st 2006. These consist of 50 featured and 50 cleanup articles. Table 5.7 provides a summary of the statistics of the test dataset.

To find articles that were featured on January 31st 2006 we consulted the revision of the featured articles page from that date [wik06]. To find articles that were cleanup articles we inspected the MediaWiki mark-up of article revisions on that date for the clean-up tag “`{{cleanup}}`”.

To create PROV versions of the Wikipedia articles we have used the wikipedia-provenance tool [Mis12] used to generate the existing ProvBench wikipedia-provenance data. We have extended the tool in two ways [Gam12]. Firstly we have extended the tool to generate RDF serialisations of the PROV data. These serialisations are constructed using the PROV Toolbox [Mor12], a Java-based toolset provided and maintained by the Provenance community. We have also extended the wikipedia-provenance to include four further pieces of metadata for each revision description:

- An influence factor for the qualifiedRevision between two `wikiprov:articles`.

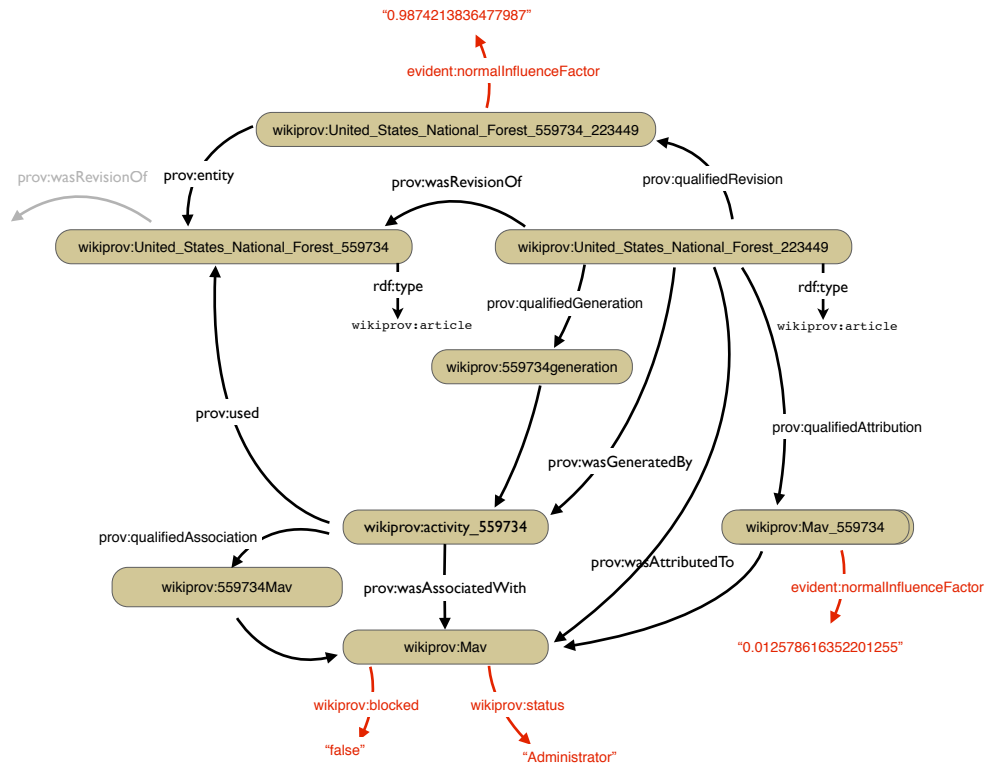


Figure 5.15: Wikipedia Provenance Example with Additional Metadata

- An influence factor for the qualifiedAttribution between the `wikiprov:article` and its `wikiprov:editor`.
- The authors type [Administrator, Registered, Anonymous] using the property `wikiprov:status`.
- Whether the author is blocked or not [true, false] using the property `wikiprov:isBlocked`.

Figure 5.15 provides a full example of the provenance graph for one revision highlighting the four new pieces of metadata⁴. The listing below illustrates the additional information for author type (line 2) and the author block status (line 4).

```

1 wikiprov:Mav wikiprov:userid "62"^^xsd:string ;
2   wikiprov:status "Administrator"^^xsd:string ;
3   wikiprov:memberOfGroup "sysop"^^xsd:string , "*"^^xsd:string , "user"^^
   xsd:string , "autoconfirmed"^^xsd:string ;
4   wikiprov:blocked "false"^^xsd:boolean .

```

⁴An RDF serialization the same revision is provided in Appendix G.

Influence Factor for Wikipedia revisions

Influence factor is included in our data for two qualified influences using the property `evident:normalInfluenceFactor`:

- As part of the `qualifiedAttribution` between the author and the revision.
- As part of the `qualifiedRevision` between the current and previous article revision.

We have based the influence factor on the number of words contributed to the revision. To calculate the influence factor we used a popular open source word-based diff tool `wdiff` [WDI12]. The tool calculates three values when comparing two revisions: the number of words in *common*, *changed*, and *inserted*. To represent the author's influence as a single quantified value we calculate the ratio between the number of words in common (*common*) with the total number of words in the previous revision (*previous*). The influence factors are calculated as follows:

- The influence factor for the author on the `qualifiedAttribution` is $1 - (common/previous)$.
- The influence factor for the `qualifiedRevision` is calculated as $(common/previous)$.

The listing below illustrates the inclusion of an influence factor for a qualified attribution:

```

1  wikipro:United_States_National_Forest_559734 prov:qualifiedAttribution wikipro
   :Mav_559734 .
3
3  wikipro:Mav_559734 a prov:Attribution ;
4      prov:agent wikipro:Mav ,
5      evident:normalInfluenceFactor "0.012578616352201255"^^xsd:double .

```

5.6.3 The Evident Generator

Figure 5.16 illustrates how the Evident generator has been implemented as an extension to the Evident framework presented in Chapter 4. The generator provides an implementation of the Quality Fragment generation procedure described in Algorithm 2 and the Combination Strategy described in section 5.5.3.

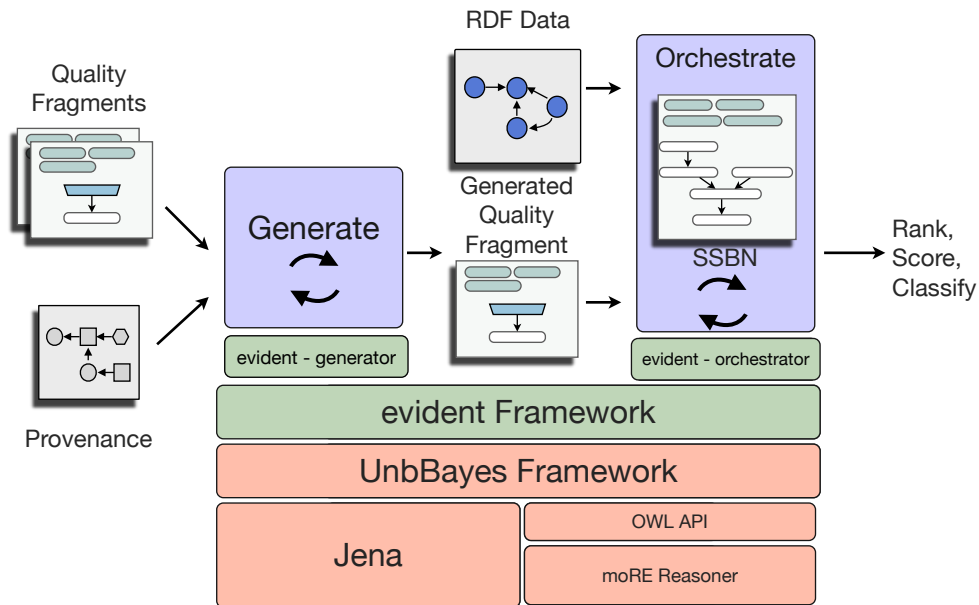


Figure 5.16: The Evident Generator Framework

The Evident orchestrator has also been extended to support the influence factor normalization procedure described in section 5.5.4 during SSBN generation.

To perform the generate procedure the Evident generator takes three inputs:

1. A URI for a PROV RDF file to be used as provenance data.
2. The URI of the entity in the PROV RDF to use as the example resource to generate a Fragment for e.g. `wikiprov:United_States_National_Forest_223449` as an example `wikiprov:article`.
3. A list of URIs for Quality Fragment files encoded in PR-OWL2, to be consulted for intrinsic metrics.

The result of the procedure is a Fragment encoded in PR-OWL2 that can be used with the orchestrator to evaluate the quality of other resources with a similar provenance representation to 2.

5.6.4 The Evident Vocabulary for Provenance-aware Quality Fragments.

We have extended the Evident vocabulary introduced in the previous chapter to support the requirements of our procedure. Specifically, we have introduced

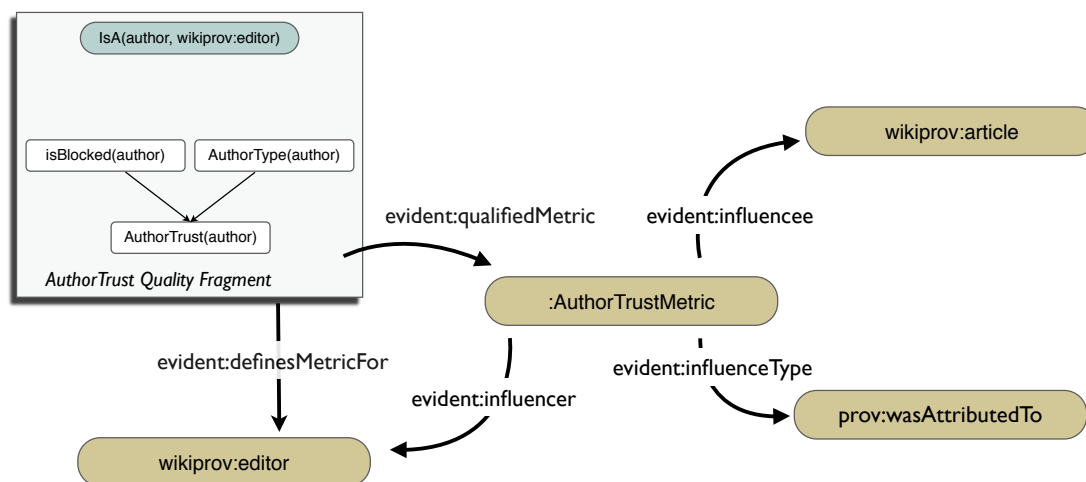


Figure 5.17: AuthorTrust Quality Fragment Described using evident Vocabulary.

terms to create *provenance-aware* Quality Fragments. The descriptions are used by the evident generator to discover Fragments that provide an assessment for a given data type (performed on line 20 of Algorithm 2).

The vocabulary allows the modeller to describe when the Quality Fragment can be used by restricting the specific type of provenance influence relationship, or the specific provenance influencee. This is demonstrated in figure 5.17. Using `evident:qualifiedMetric` the AuthorTrust Quality Fragment is restricted to when the influence that the `wikiproveditor` had was of type `prov:wasDerivedFrom` and when the influencee was of type `wikiproveditor`. These additional restrictions are inspected during the search for an existing Fragment.

5.6.5 The Generated Quality Fragment

Using the evident generator we have generated a Quality Fragment to assess the quality of a `wikiproveditor`. To generate the network we passed the following arguments to the Evident generate procedure:

- The wikipedia-provenance RDF PROV graph for United States National Forest.
- The Quality Fragment for Author Trust described in section 5.6.1.

- A reference to the Web resource `wikiproveditor:UnitedStatesNational-Forest_223449` as the example resource.

The Generated Network: `articleQ`

Figure 5.18 shows the structure of the *articleQ* Quality Fragment generated for the type `wiki:article`. This Fragment can be used to assess the quality of any `wiki:article` that has provenance data in the same structure.

The quality assessment value for a `wiki:article` is provided by the resident node *article_Q(article_1)*. The Quality Fragment has two input nodes that reuse two Fragments. The two input nodes have been included as a result of the two quantified influences for `wikiproveditor:UnitedStatesNationalForest-223449` described in the PROV graph in figure 5.15:

- The quantified attribution influence from the `wikiproveditor` and the availability of the intrinsic AuthorTrust Fragment has resulted in the *AuthorTrust(editor_1)* input node.
- The quantified revision influence from the previous revision's `wikiproveditor:article` and the availability of the newly generated Quality Fragment for `wikiproveditor:article` has resulted in the *article_Q(article_2)* input node.

The AuthorTrust quality fragment is used via the *AuthorTrust(editor_1)* input node, where *editor_1* is the editor who `wasAttributedTo article_1`. The Quality Fragment has an input node *article_Q(article_2)*, where *article_1 wasRevisionOf article_2*.

The *article_Q* Quality Fragment reuses itself via the *article_Q(article_2)* input node. This is because a `wiki:article` is influence by another `wiki:article` via the `wasRevisionOf` influence. This recursive reuse of *article_Q* is analogous to the iterative step in the Zeng network.

We can understand this ability for the Fragment to reuse itself from the $Q \leftarrow Q \cup q_{current}$ operation on line 17 of the generation procedure in Algorithm 2. By adding the newly generated Fragment *article_Q* to the set of existing Fragments *Q* it makes it available to be reused.

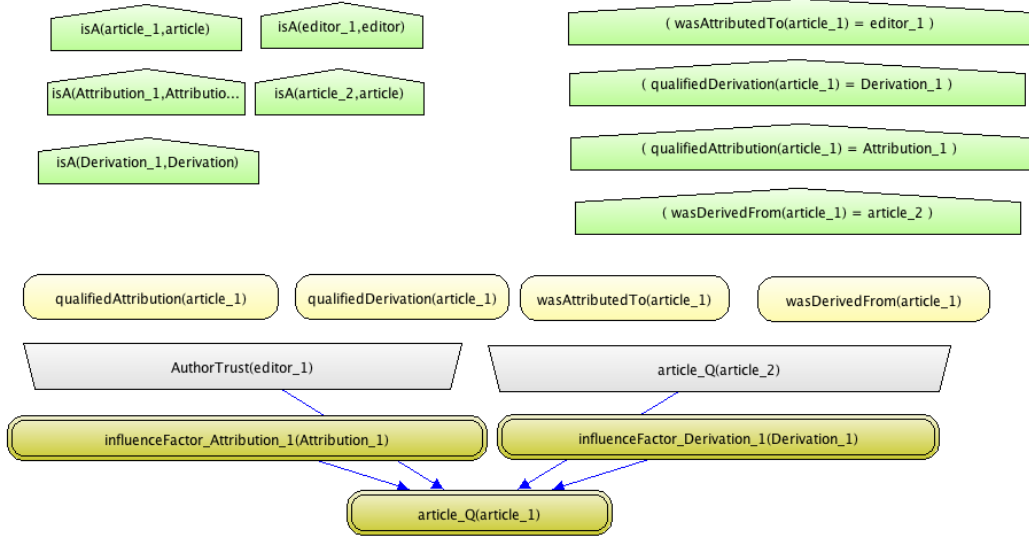


Figure 5.18: The Automatically Generated Quality Fragment for Wikipedia Provenance.

The Generated CPD for *article_Q*

The CPD for the *article_Q(article_1)* has been automatically generated according to the combination strategy in equation 5.1. The resident node combines influencing configurations that result from the *AuthorTrust(editor_1)* and *article_Q(article_2)* input nodes and their corresponding influence factors, *influenceFactor_Attribution_1* and *influenceFactor_Revision_1*. Listing 5.3 details the CPD script generated for *article_Q(article_1)* using the UnbBayes scripting language. The *if* statements check which input nodes have existing influencing configurations⁵.

These existence checks are performed at SSBN generation time in our modified Laskey algorithm to select the appropriate section of the CPD based upon the instantiated influencing configurations. For example if *article_1* has an influencing configuration defined for both the *article_Q* and *AuthorTrust* input nodes then the CPD for that node is defined as:

```
1 ( article_Q * influenceFactor_Derivation_1 ) + ( Sum(AuthorTrust) * Sum(
    influenceFactor_Attribution_1 ) ) * NormalDist(1,0)
```

However if the revision *article_1* has an influencing configuration for the *AuthorTrust(editor_1)* input node defined, but not *article_Q(article_2)* then the CPD

⁵This existence check has been implemented as an extension to the standard scripting language provided by the UnbBayes framework


```

1 if any article_5 have ( article_Q = Defined) [
2   if any editor_3 have ( AuthorTrust = Defined) [
3     ( article_Q * influenceFactor_Derivation_6 ) +
4     ( Sum(AuthorTrust) * Sum(influenceFactor_Attribution_4) ) * NormalDist
      (1,0) ]
5   else [
6     ( Sum(article_Q) * Sum(influenceFactor_Derivation_6) ) * NormalDist(1,0) ]
7   ]
8 else [
9   if any editor_3 have ( AuthorTrust = Defined) [
10    ( Sum(AuthorTrust) * Sum(influenceFactor_Attribution_4) ) * NormalDist(1,0)
11    ]
12   else [
13     NormalDist(1,0)
14   ]
15 ]

```

Listing 5.3: The CPD Script Generated for $article_Q(article_1)$

for that Quality Fragment instance would instead be defined as:

```

1 ( Sum(AuthorTrust) * Sum(influenceFactor_Attribution_1) ) * NormalDist(1,0)

```

This would be the case for the first revision of a Wikipedia article, where $article_1$ would not have an $article_2$ that satisfies the context node (*was RevisionOf(article_1) = article_2*).

5.6.6 Comparison of the Generated Network With the Zeng Network

We have used our generated Fragment to assess the PROV extractions of the 500 Wikipedia articles in the training set and the 100 Wikipedia article in the test set as described in section 5.6.2. Table 5.8 shows a summary of the results for the training set from the article_Q Quality Fragment compared to the results of Zeng. The table provides the average assessment value for the final revision of each article. Table 5.9 provides the full set of results for featured and cleanup articles in the training set. The assessment values from article_Q demonstrate a similar relative scoring to Zeng. The assessment value for featured articles is higher on average than normal articles and the assessment value for cleanup articles is lower. There is a narrower range between the three classes at 0.07 for our assessment compared to Zeng at 0.117. By using the same prior knowledge for Author Trust we can conclude that differences between the two networks results from two sources: (1) The combination strategy, or (2) the influence factor values used as evidence.

Zeng provides a detailed assessment for the Wikipedia article “United States

Training Set	Featured articles	Clean-up articles	Normal articles
Average tv_{final}	0.885	0.768	0.808
Average $article_Q(final)$	0.766	0.696	0.728
Difference	0.119	0.72	0.80

Table 5.8: Results from the article_Q Quality Fragment compared with the Zeng Network

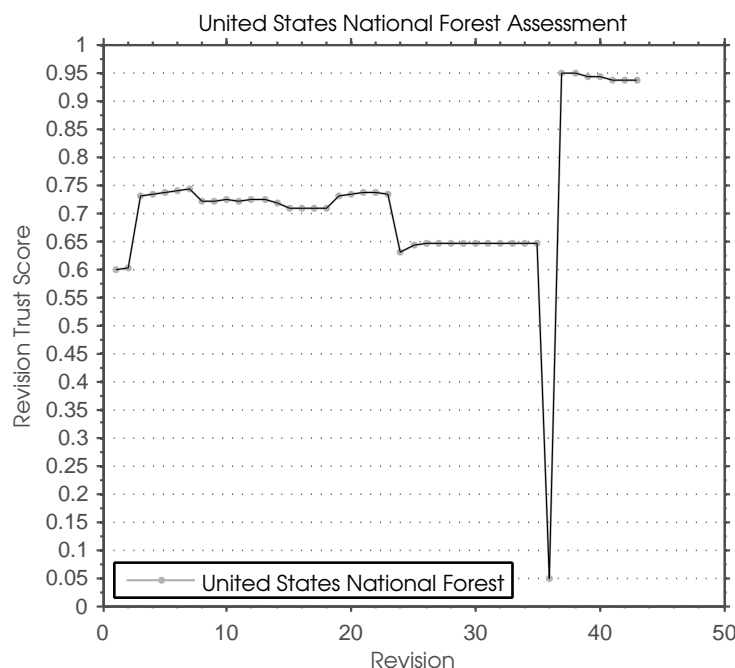


Figure 5.19: Assessment Values for Revisions of the United States National Forrest Wikipedia Article using Automatically generated Quality Fragments

National Forest”. The graph in Figure 5.20 is taken from Zeng and shows the trust values for the first 43 revisions of the article “United States National Forest” computed by the Dynamic Bayesian Network. Figure 5.19 shows the same 43 revisions assessed by our automatically generated network.

A comparison of the two graphs shows that our Fragment provides a similar estimation of the incremental changes in trustworthiness across revisions of the United States National Forest article. At major revisions both assessments capture the same changes in predicted quality. We observe however a number of discrepancies between our assessment and the original. For example, at the transition from revisions 18 and 19 Zeng describes a 10% insertion and deletion

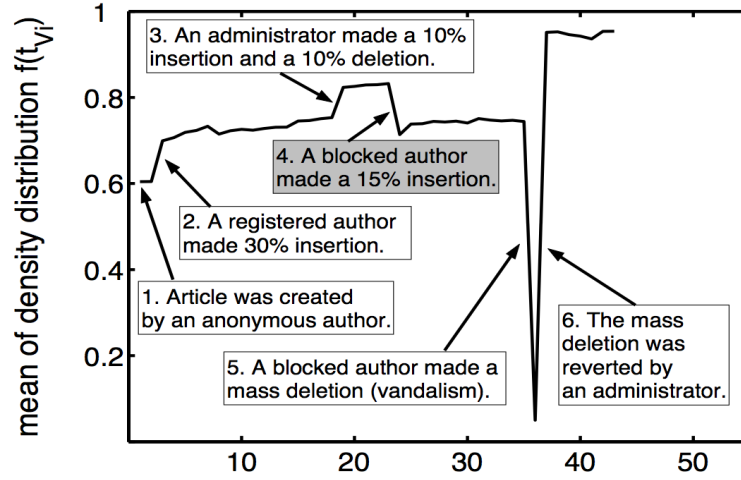


Figure 5.20: Assessment Values for Revisions of the United States National Forrest Wikipedia Article using Bayesian Network from Zeng et al.

at the transition between the two revisions which results in a significant increase in the assessment score. This increase is also reflected by our assessment, but not to the same extent. An inspection of revisions 18 and 19 does not reveal the same 10% insertion and 10% deletion when evaluated by `wdiff`. This provides some evidence that the specific implementation of `diff` used to calculate the differences between pages contributes to the observed differences.

For a more thorough comparison we have also created a classifier from the assessments of our training data. Like Zeng we used the values for the article revision when the article was marked as *featured* or *cleanup*. The average value for cleanup articles at this point is 0.672, slightly lower than the value for the final revision. The average value for featured articles when promoted is 0.796, slightly higher than the final version. Figure 5.21 compares the values for featured and cleanup articles. The plot highlights that values for cleanup articles cluster in values around 0.62, with some overlap between cleanup and featured articles around 0.69.

To establish a rule for classification we used the Weka Data Mining software suite [HFH⁺09]. We applied the `JTree` classifier to our training set which provided a learned classification rule of:

- $article_Q(i) > 0.690884$ is featured
- $article_Q(i) \leq 0.690884$ is cleanup

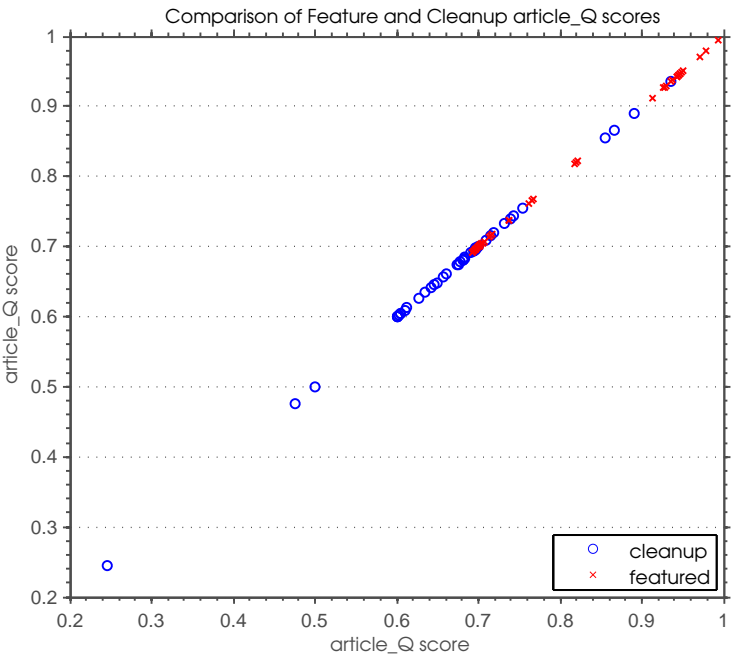


Figure 5.21: Comparison of Assessment Values for cleanup and featured in Training Set.

		Prediction			
		cleanup	featured		
actual value	50	30	20	cleanup	
	50	2	48	featured	
total		32	68		
Correct 78% Incorrect 22%					

Figure 5.22: Confusion Matrix for Test Set Classification

We have assessed the 100 PROV graphs from our test set (Table 5.10 provides a full set of results for the test set) and applied our classification rule

to each result. Figure 5.22 illustrates the classification as a confusion matrix. The result of our classification is an overall successful classification of 78%, 5 percentage points fewer than the Zeng network. When considering individual classes we have successfully predicted 96% of the featured articles, and 60% of the cleanup articles. Our network therefore predicted a high percentage of featured articles, and fewer cleanup articles. This result shows that our automatically generated network is successfully able to approximate the Zeng network and distinguish between quality classes. The ability to distinguish between classes falls short of the hand-crafted Zeng network. We believe however that this is a successful initial outcome with scope to improve the generated network by experimenting with alternative combination strategies.

Geography Featured	final score	promoted score	revision (promoted)	Cleanup	final score	demoted score	revision (demoted)
Aztalan State Park	0.914329213	0.767801815	37457311	All time Olympic Games medal table	0.696969697	0.696969697	35272301
Bath	0.767801815	0.923856003	37453127	BC	0.744282008	0.743243028	34018433
Belgium	0.923856003	0.821895364	37452642	Chen Chang siang	0.70211039	0.673747681	16938907
Byce Canyon National Park	0.697063199	0.929525183	37523761	Chinese music	0.49314704	0.473515278	23519486
Buckingham Palace	0.687634631	0.993682998	37536910	Crop circle	0.615309708	0.681060433	29388581
Cambodia	0.296805043	0.761418124	37453863	Cultural geography	0.719582912	0.719363378	24872035
Canberra	0.683721187	0.693721187	37493161	Do it yourself	0.64097322	0.648805474	15938155
Cathedral of Magdeburg	0.937009073	0.947059577	37457279	Doughterty Island	0.684596389	0.68408073	24934363
Chennai	0.8695837	0.942494729	37539208	Ecoticism	0.696969697	0.696969697	33021176
City status in the United Kingdom	0.817192666	0.942077619	37564282	Ecological modernization	0.645292039	0.644933666	36495090
Colditz Castle	0.61747327	0.693468352	37453894	European Geography Association	0.605373681	0.6	22902565
Craters of the Moon National Monument and Preserve	0.909972406	0.935077653	37453893	Everglades	0.696897439	0.612094771	22346494
Death Valley National Park	0.897202932	0.93781681	37457557	Expedition 1	0.690969682	0.690883849	28170673
Dorset	0.67977798	0.695857231	37457615	Flonistuy	0.600804251	0.600804251	35721413
Edfell	0.696699526	0.696849383	37457658	Frame of reference	0.649644259	0.609585678	23654294
Gangtok	0.698861436	0.704806957	37405105	Geography of Armenia	0.874368064	0.6751942	24377272
Geography of India	0.919070387	0.70707727	37405020	Geography of Asia	0.855253049	0.855253049	30678155
Geography of Ireland	0.691028345	0.699980053	37457685	Geography of Azerbaijan	0.885327926	0.626776313	24416461
Goa	0.471901646	0.696668328	37405303	Geography of Estonia	0.84985656	0.603774101	24415491
Gyeongju	0.898799091	0.944839361	37457765	Geography of Kenya	0.70122679	0.696949365	24026455
Hong Kong	0.696917141	0.696962199	37578131	Geography of Turkmenistan	0.923976442	0.866731146	18459285
India	0.942128531	0.926480805	37587846	Geography of the Netherlands	0.738630898	0.738697079	37021836
Isan	0.735161074	0.946846885	37457813	Global 200	0.69633975	0.697070626	13019603
Johannesburg	0.647432592	0.715247045	37457852	Grose Valley	0.800025175	0.660507773	28868731
Kalimpong	0.945154303	0.704186283	37495551	Harvey Point	0.63500303	0.63500303	27299749
Kerala	0.93182338	0.947898907	37457885	History of gardening	0.712873458	0.714807517	32247915
Lake Burley Griffin	0.682639782	0.697329349	37457924	Knotted cord	0.847647483	0.696139805	21985154
Mountgare	0.65845768	0.702129077	37458022	Kurdia	0.698338589	0.699414434	30871404
Mount Pinatubo	0.949216779	0.702130912	37558752	Kurdistan	0.500294369	0.500294369	37275504
Mumbai	0.766188588	0.766188588	37510628	List of Antarctic and subantarctic islands	0.732224262	0.732375444	33505663
National parks of England and Wales	0.693223384	0.692152093	37453114	List of Monterey metro stations	0.671369357	0.656304844	24715929
Niagara Falls	0.718063177	0.696418225	37512966	List of countries without armed forces	0.644276206	0.642486098	37000726
Oakland Cemetery Atlanta	0.698372865	0.698009164	37458226	List of environmental organizations	0.680658397	0.680658397	35742458
Palace of Westminster	0.878376595	0.712393518	37559902	Mahmud al Kashgari	0.62216655	0.602620803	15419090
Palazzo Pitti	0.738179018	0.737272673	37458579	Milton Santos	0.680800329	0.600260671	17530731
Piccadilly Circus	0.677525265	0.702353886	37460491	Mobile telephone numbering in India	0.607893254	0.708756491	21628744
Rondane National Park	0.707196674	0.927337778	37453625	Music history of Portugal	0.687668	0.69364034	33365531
San Jose California	0.92828379	0.701328616	37488641	Paifiti	0.683314149	0.69669697	35759583
Sarajevo	0.698053148	0.97826217	9729148	Panorama	0.696969697	0.890244207	28264830
Sheffield	0.821694307	0.820446081	3751948	Rain garden	0.871630662	0.696969697	34274767
Shrine of Remembrance	0.709324242	0.696969697	37467461	Rotating reference frame	0.245084473	0.245084473	36217397
Sikkim	0.526706285	0.736620645	37460639	Serbia	0.695604785	0.682262846	36276726
South Africa	0.970825	0.970825	37524907	Soit type	0.667321484	0.677692266	16264324
Suburbs of Johannesburg	0.717133956	0.717391778	37459206	Telephone numbers in the Republic of Ireland	0.669815172	0.600769601	28226446
Surtey	0.708082601	0.692056056	37459375	The Gambia	0.919498866	0.935666992	34891639
Yarralumla Australian Capital Territory	0.817419127	0.817920897	37458586	Time travel in fiction	0.634407498	0.696969697	21892551
Yellowstone National Park	0.696863383	0.699270806	37458852	Tirana	0.748309764	0.75450288	29184065
Yosemite National Park	0.695529666	0.912528833	37459942	Traffic wave	0.697867311	0.698710585	31769084
Zambesi	0.927227577	0.702286027	37555573	Urban geography	0.69586717	0.693476718	19924692
Zion National Park	0.914679085	0.944753761	37522136	Wallops Flight Facility	0.662054381	0.661430931	24310224

Table 5.9: *article-Q* Results for Featured and Cleanup Articles in Training Set

Biology	promoted score	revision	(promoted)	Cleanup	demoted score	revision	(demoted)
Featured							
Action potential	0.701500289		2106709	Acetogenesis	0.608536585		29661175
Antarctic krill	0.766865051		15734106	Advanced glycation end product	0.621617606		37275020
Asperger syndrome	0.993508362		22587461	Antioxidant	0.644125477		29990887
Asthma	0.784787016		22382627	Bacterial genetic nomenclature	0.696066166		33833129
Australian green tree frog	0.731010417		31004952	Bibliography of biology	0.651911892		32264079
Barbara McClintock	0.6		27573710	Bioelectromagnetics	0.696969697		30284521
Cat	0.703565922		21387779	Bionomics	0.609038264		22465515
Cerebellum	0.69337877		22208420	Chandra Wickramasinghe	0.678198759		32712652
Chagas disease	0.69391889		18353780	Collective intelligence	0.64798561		22499151
Chemical synapse	0.700124165		2085401	Crocodylia	0.671354543		25328311
Cladistics	0.812653047		2212374	Domesticated plants of Mesoamerica	0.6		28930840
Coconut crab	0.906407873		6588577	Dormancy	0.984131737		37247203
DNA repair	0.711641581		5497440	Emperor Penguin	0.913151436		32573940
Dinosaur	0.999256949		31719639	Evolutionary radiation	0.600775618		17134511
Evolution	0.697688392		9853289	Eye	0.686454566		27321453
Fauna of Australia	0.891138785		25183202	Fasting	0.981585502		37226063
Gene	0.740626171		2284060	Fat	0.729809105		23066522
Gray wolf	0.748431429		22023176	Flavonoid	0.674257124		24628959
Helicobacter pylori	0.746491607		6100137	Fluid balance	0.620647258		32554281
History of saffron	0.949160479		34632625	Foot	0.741214513		15694452
Homo floresiensis	0.844312088		7798119	Forgetting	0.612033481		15908752
Human	0.9748237		27043990	Human ecosystem	0.601661863		13192226
Kakapo	0.695510152		5230574	Iconic memory	0.695937323		16545709
Killer whale	0.698800069		5058609	Immunoelectrophoresis	0.629615385		27209082
Krill	0.71459153		18766488	Immunophenotyping	0.61989122		30779635
Lesch Nyhan syndrome	0.993577318		9497282	Inclusion bodies	0.69803285		24832639
Margined tortoise	0.702670152		7122255	Islamic views on evolution	0.948541369		36230402
Marine shrimp farming	0.647380343		20780714	Kidding Aside	0.709742186		32020186
Menstrual cycle	0.910773154		2213031	List of prehistoric mammals	0.677074003		25650570
Multiple sclerosis	0.940414968		25518156	Llama	0.94940813		37012582
Myxobolus cerebralis	0.701244644		14345478	Lydia Fairchild	0.626039958		28457950
Norman Borlaug	0.949923441		18861589	Macroevolution	0.875400275		34373924
Paracetamol	0.704397815		3523359	Macroevolution	0.875400275		34373924
Pneumonia	0.705864195		28727038	Megatrajectory	0.687981818		17128714
Prostate cancer	0.69920522		31631705	Metastability in the brain	0.604305879		23154829
Saffron	0.948407073		32929345	Microscopy	0.935967318		29672678
Short beaked echidna	0.713066		32150344	Milt	0.769715909		37086125
Tooth development	0.69696018		1760192	Mycofiltration	0.949015446		27668733
Tooth enamel	0.746272976		18916319	Neanderthal extinction hypotheses	0.698096063		35649187
Tuberculosis	0.791032579		2213026	Noogenesis	0.617073171		23267105
Platypus	0.987290939		37494535	Nuclear chain fiber	0.629395734		13369583
Mongrel	0.998098335		584832812	Nuclear fission product	0.690123269		32300162
Island fox	0.768130258		15276053	Obligate parasite	0.742464034		32460447
Sperm whale	0.703946028		1484601	Overnutrition	0.619737195		29883572
Blue whale	0.753621658		13914717	Personalized medicine	0.603608247		30917928
Humpback whale	0.724500176		4359835	Plant Patent Act of 1930	0.601978973		27062655
Tasmanian devil	0.743795511		19673899	Progressive contextualization	0.602336619		22636898
United States Navy Marine Mammal Program	0.948009501		25729220	Quantum evolution	0.685977973		22855977
Whale vocalization	0.920064104		580955763	Reciprocal altruism	0.687954885		34138881
Mdecins Sans Frontieres	0.796769697		36387966	Representative sequences	0.601564027		5486987

Table 5.10: *article_Q* Results for Featured and Cleanup Articles in Test Set

5.7 Comparison with Related Work

In Chapter 2 we discussed a number of previous IQ solutions that made use of provenance-based Quality Knowledge in their assessments. In our evaluation we have compared our approach in detail with one such approach, demonstrating how we can automatically generate the provenance-based aspect of the assessment.

Other work that combines intrinsic quality with provenance-based quality such as Golbeck et al. [GH06], Groth et al. [GMM⁺09], and Zaihraye et al. [ZDSM05] is also relevant. The most closely related work is that of Ceolin [CGvH⁺12] who also uses a combination of uncertainty reasoning and provenance information to make trust assessments in the Web of Data. We have previously discussed the benefits and limitations of subjective logics used by Ceolin, and similarly see the potential benefit of incorporating subject logics for a more detailed assessment result. This would however require an alternative combination strategy to take into account the alternative uncertainty representation.

Previous approaches to provenance-based quality assessment have focused on a specific assessment scenario, or specific intrinsic quality measures. We see the approach presented here as complimentary to this previous work, and a move towards a more general approach for provenance-based quality assessment. With our procedure we have proposed a method of bootstrapping the building of similar provenance-based assessments, by encoding the intrinsic measures of quality as Fragments, and inferring the provenance based aspects of the assessment from provenance data.

We could for example perform an assessment such as the one presented by Golbeck in the FilmTrust system by creating a Fragment that encodes the TidalTrust metric, and infer the provenance-based Quality Knowledge through our procedure from provenance information relating to the attribution of Film reviews to authors.

Our uniform representation using MEBNs also afford us the ability to comparatively evaluate alternative approaches to modelling the same type of intrinsic assessment. For example with multiple agent trustworthiness Fragments we could use the same provenance graph, and substitute the Fragments to compare results. This extends the analytical capabilities provided by our approach to modelling predictive Quality Knowledge.

We are not the first to recognize the need to annotate provenance with further quantitative information to support the computational tractability of quality assessment. In their work using provenance to assess the timeliness of Web Data Hartig et al. [HZ09] proposes a type of annotation called *impact values*. In contrast to our influence factor, the authors use the term impact values to refer *any* type of metadata that informs a quality assessment. What is considered an impact value is therefore contextual, and tied to the particular type of quality assessment being performed. For example an impact value might be the *creation time* of a resource for a timeliness assessment, or a data creator's *credibility* for a believability assessment. Our influence factor is instead a general mechanism for capturing a quantitative influence from one resource to another. We therefore see influence factor and impact values being complimentary, where influence factor can be characterized as a certain class of impact value, depending on the quality assessment.

5.8 Conclusions and Future Work for Quality Fragment Generation

In this chapter we have described a procedure that automatically creates reusable quality components, in the form of MEBN-based Quality Fragments, by analysing provenance data and existing Fragments. This work was motivated by a common intuition observed in many of the existing provenance based assessments, where the quality of a resource is informed by the quality of those that produced it. We have shown that a quality component automatically generated by our procedure can provide a predictive IQ assessment that is a close approximation to a Bayesian Network solution created by hand to encode this type of provenance-based assessment.

We have also demonstrated the utility of `influenceFactor` an additional vocabulary feature for provenance description that serves to quantify the qualified influences in the PROV specification. We have shown that we can use this new feature along with existing provenance data to evaluate the quality and trustworthiness of data on the Web.

Our automatically generated Fragment has successfully demonstrated the ability to predict quality classes. Comparison with the hand-crafted network has shown that it is close in overall accuracy. We believe that it demonstrates

a successful initial step towards automatically generating Bayesian Network-based Fragments from provenance data. A key contribution of the work is in establishing the three necessary components to build MEBN-based Fragments from Provenance, these are:

- A structure generation procedure.
- A combination strategy.
- An impact factor normalization strategy.

This procedure is also a further step towards our vision of an ecosystem of reusable quality components, where engineers and users are provided the ability to use, combine, benchmark and develop reusable quality components for the Web of Data. A Data Scientist using a machine learning tool such as Weka to build a classifier is afforded the ability to explore different classifiers and parameters to meet their needs. We similarly envision a Quality Knowledge Engineer experimenting with alternative combination strategies and influence factor normalization strategies to develop reusable quality components.

Beyond our case-study the ProvBench data provides further examples that we can use to evaluate how well we can generalize our approach. We can use the eScience workflow provenance traces provided by the WF4ever project to generate Quality Fragments. In the spirit of the project's preservation goal we could use the provenance traces to build Fragments that detect workflow decay and assess the likelihood of Workflow still running. These assessments would necessarily rely on some prior knowledge about the reliability of the Web Services, such as up-time statistics from a long term monitoring of Web Services from service such as BioCatalogue [BTN⁺10].

We see similar potential for the evaluation of distributed SPARQL queries. With a number of approaches being developed to encode the provenance of SPARQL queries [DAA12] [GKCF13], given similar prior knowledge about endpoint up-time or graph availability, we could dynamically build Fragments to assess the likelihood that we can continue to run these queries.

Further case-studies would also allow us to explore the use of **influence-Factor** to quantify influence, and identify its potential scope. We are interested in investigating further the properties of influence factor, for example the properties and limits of its transitivity. We have also discussed only one approach to normalising influence factor and look to investigate alternative approaches.

Our current procedure does not support a number of the extended PROV features. A useful initial extension would be to consider PROV collections. Scientific information on the Web of Data is commonly represented as aggregations of Web resources. It would therefore be useful to understand if we can automatically infer how the IQ assessment of individual parts of a collection can contribute to the assessment of the whole.

5.9 Chapter Summary

In this chapter we have presented a novel procedure to build Multi-Entity Bayesian Network-based Quality Fragments using provenance data. We have described the intuition behind our procedure, based upon a desire to generalise existing provenance-based assessments. We have grounded our work in the PROV provenance model, and have defined influence factor as an additional aspect of how provenance required to automatically build Quality Fragments provenance. We have describe the three components of our approach; a structure generation procedure, a combination strategy, and an influence factor normalization strategy. To evaluate our procedure we have replicated an existing hand-built Bayesian Network using our Evident framework, and shown that we can successfully approximate its results.

In the next and final chapter we draw together conclusions for our work in modelling and assessing Quality Knowledge in the Web of Data, and propose an agenda for future work.

Chapter 6

Conclusions

“*And so another, even bigger computer had to be built to find out what the actual question was.*”

- *Douglas Adams, The Hitch-Hiker’s Guide to the Galaxy.*

The Web of Linked Data has emerged as a platform through which the sciences can publish, share, discover and ultimately reuse data in a machine readable manner. The openness of this Web of Data has led to data of varied quality and trustworthiness, presenting scientists with the challenge that the quality of data they wish to use is *unknown*.

The research we have described in this thesis has been an investigation in response to this challenge, developing approaches that support the automated quality assessment of scientific data on the Web of Data. In particular we have sought to establish mechanisms to build reusable components that can be shared in the Web of Data and used by others to mitigate this IQ challenge. In the next section we summarize the research contributions that have resulted from this investigation. Figure 6.1 supports this discussion and illustrates the structure of the thesis that we have presented.

6.1 Summary of Research Contributions

We began this thesis by posing a motivating question:

What is the likely quality of the entry for Ethane in DBpedia?

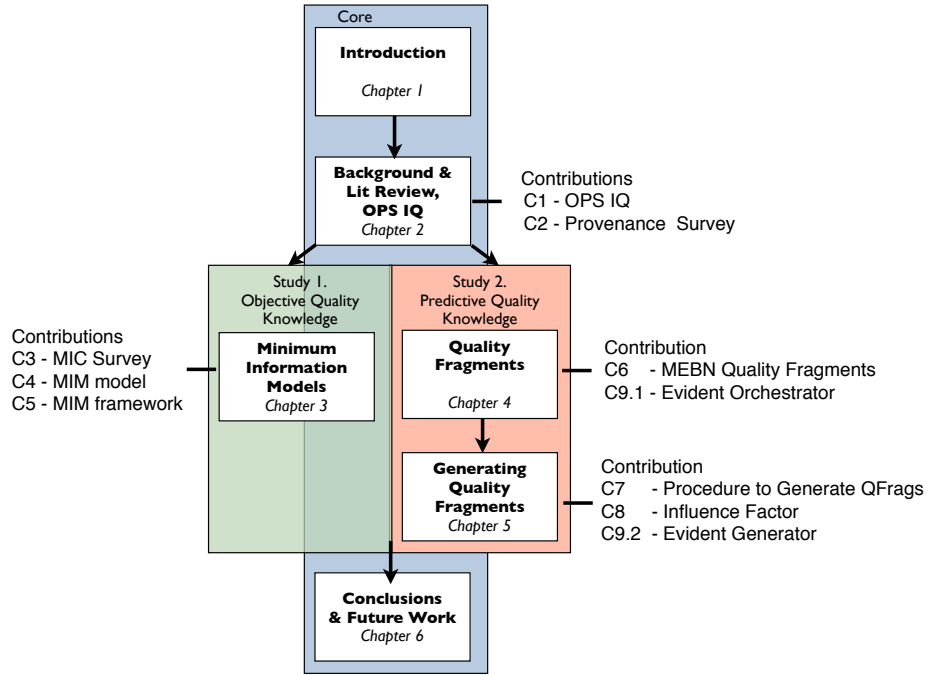


Figure 6.1: Structure of the Thesis Presented

To develop IQ assessments capable of answering our motivating question, our investigations have had two focuses:

- Develop a methodology for IQ that reflects the challenges and opportunities posed by sharing scientific data in the Web of Data.
- Develop techniques and infrastructure using our methodology that support the scientific user in managing and overcoming the problems that are posed by data of unknown quality.

To do this, we have grounded our investigations in the outcomes of the previous work of Missier [Mis08] who investigated the modelling and computation of quality in eScience. In particular, we have adopted and developed three concepts posed by Missier: *Quality Knowledge*, *reusable quality components* the *Information Quality Life-cycle*. The overarching hypothesis (H1) that we have explored is that by using the IQ Life-cycle as a supporting framework we can develop IQ solutions in the Web of Data that support quality based decisions.

In the course of this thesis we have presented two approaches to assessing IQ that are capable of answering our motivating question and in turn go towards supporting our overarching hypothesis:

- **chemmim:** Using our MIM framework and chemmim assessment the Ethane chembox minimally satisfies the chemmim checklist, but maximally satisfies the Identifiers requirement set in that checklist. This suggests that the Ethane chembox, like many of the chemistry articles in Wikipedia, is a good source of chemical identifiers.
- **article_Q:** The result of our generated Quality Fragment for Wikipedia articles provides a score of 0.76633 for the Ethane article. Based upon our classification this suggests that in its current state, the article for Ethane is more likely the quality of a featured article than a cleanup article.

These two approaches were developed in response to our IQ classification, Objective Predictive Subjective IQ.

Objective Predictive Subjective IQ

To apply the IQ Life-cycle to the Web of Data we examined the concept of Quality Knowledge and argued that there exist three distinct aspects that can be defined in terms of the information that support their assessment.

- Objective - Informed by an objectively defined standard.
- Predictive - Informed by prior knowledge that can relate features of the data to its likely quality.
- Subjective - Informed by the user's subjective requirements for quality.

In support of our second hypothesis (H2) we have demonstrated in this thesis that by using OPS IQ we can develop IQ techniques and infrastructure that can successfully evaluate the quality of scientific information on the Web of Data. Where previous work has tailored IQ approaches to the subjective needs of the user, we have tailored our approaches to our assessment-oriented classification, focusing on two of the aspects.

Assessing Objective Quality Knowledge with Quality Standards: Observing the role that standards play in Web-based science and the objective elements of scientific Quality Knowledge. The objective nature of the Quality Knowledge utilized means that we achieve an assessment that is broadly relevant. The ability to perform a large-scale standards-based assessment of

provides feedback not just for users, but each of the stakeholders involved: data consumers, data providers and the standards creators themselves.

Assessing Predictive Quality Knowledge with MEBN-based Quality Fragments: We have shown that Multi-Entity Bayesian Networks, a template-based Bayesian Network approach are a suitable mechanism for modelling predictive Quality Knowledge in the Web of Data. MEBNs provide a general and uniform approach to modelling Quality Knowledge that is a first step towards a library of reusable quality components. Given a universal representation these components can be more readily combined, compared and benchmarked.

We believe we have demonstrated that our assessment-oriented classification provides a useful grounding for developing IQ solutions that reflect the scientific users approach to IQ in the Web of Data. In the following we summarize the main contributions to result from this classification.

The Minimum Information Model Vocabulary and Framework

The development of the Minimum Information Model Vocabulary and Framework has focused on modelling Minimum Information Checklists (MICs), a particular type of Quality Knowledge. Through an analysis of existing examples of MICs we have established the salient features of a checklist. The resulting MIM vocabulary is a Quality Knowledge Encoding that can be used to share MICs as RDF documents in the Web of Data, and that can be used to evaluate the completeness of data. We demonstrated an approach to separating the concerns of Quality Knowledge Encoding and Alignment through the use of a rule-based approach, which further broadens the scope of the assessment.

Our successful evaluation of the chembox data provides support for our hypothesis (H3) that a checklist-based approach can be successfully applied to evaluate the quality of Linked Data. Specifically we have demonstrated this by evaluating the data along the dimension of completeness.

Our evaluation of the MIM framework has also highlighted interactions between the Information Quality Life-cycle and the process of generating a Linked Data extraction. We have shown that there is mutual benefit to be gained from understanding these interactions to improve both Quality Knowledge Engineering and Linked Data Extraction.

Multi-Entity Bayesian Network-based Quality Fragments

A more general approach to modelling predictive Quality Knowledge explored the use of Multi-Entity Bayesian Networks as an encoding for Quality Fragments. In particular, we have demonstrated the following strengths of such an approach:

- Their grounding in first order logic makes them particularly suited to the Web of Data for the purposes of aligning with existing RDF data.
- Their template-based modelling reflects the modular nature of quality metrics, and allows them to be flexible for re-use in the face of the inconsistent and varied data in the Web of Data.

In support of hypothesis H4 we have shown that we can effectively model a range of metrics including four in Chapter 4 (GAQ, Group GAQ, Product GAQ, Mean GAQ), and two in Chapter 5 (Author Trust and Article Quality), and that we can successfully replicate exiting metrics in the case of the GAQ metrics. We have also successfully show that the predictive aspects of Bayesian Networks makes them robust in the face of inconsistent metadata in the Web of Data, allowing them to continue to provide best-effort assessments that can be used to make quality-based decisions where previous encodings could not.

A Procedure to build Quality Fragments from Provenance

We have exercised our MEBN-based approach further by using it as a component in the final piece of work presented in this thesis. The motivation for this work was twofold:

- The prevalent use of provenance in the assessment of predictive Quality Knowledge.
- The increasing availability of provenance metadata in the Web of Data.

This presented an opportunity to bootstrap the Quality Knowledge Engineering process. In support of our final hypothesis (H5) we have successfully developed a procedure that can generate a MEBN-based Quality Fragment, and shown that we can reuse these Fragments to support IQ assessment by successfully approximating a hand-crafted Bayesian Network based IQ assessment. In

particular we have established the information and procedures required to automatically build these Fragments from provenance, these are:

- A structure generation procedure informed by the influencing elements of a provenance graph.
- A combination strategy to combine quality scores for those influencing elements.
- An impact factor normalisation strategy to weight those scores according to their influence.

This work is a further step towards developing an ecosystem of Fragments, procedures and tooling that can support the engineer to develop and benchmark quality components for use in the Web of Data.

Influence Factor

In the process of developing our procedure, we have identified a gap in the current PROV model of provenance with respect to the metadata required for our IQ assessment. To fill this gap, we have proposed *influenceFactor* as an extension to the PROV model to represent a quantified measure of influence between two elements of a provenance graph. We have demonstrated the application and utility of influence factor in supporting IQ assessment.

We have shown for each of our approaches that the quality components created are reusable across a particular class of data, and are flexible to the heterogenous and “messy” nature of the Web of Data.

6.2 Future Work

In each chapter of the thesis we have addressed future work. Here we set an agenda for future work with respect to the broader task of managing and assessing IQ on the Web of Data.

Measuring the *Quality* and *Reusability* of Reusable Quality Components

There is a broad spectrum of potential re-use of quality components. We have outlined four desirable characteristics of reuse, and observed a number of features for example relating to alignment and encoding that impact the potential for reuse. Understanding the features and inhibitors of reuse would enable the community to improve the reusability of existing quality components, and inform the development of new quality components.

There is a long and rich literature in component-based software development that has attempted to identify and quantify metrics to measure reuse of software components [FK05]. Metrics developed address dimensions such as understandability, adaptability, and portability. These measurements can be performed on both black-box components with limited knowledge of their internal workings [WYF03], as well as through a full and detailed analysis of the component and its implementation [FT96]. Further to this methodologies such as the ISO Systems and software Quality Requirements and Evaluation (SQuaRE) (ISO/IEC 25010:2011) approach [ISO11] provide an international standard and guidance for the evaluation of software quality. Using this body of research as a grounding we see the potential to establish mechanisms to evaluate the quality and reusability of Web-based quality components.

Tooling and Infrastructure to Benchmark Reusable Quality Components

In this thesis we have proposed a number of approaches that a Quality Knowledge Engineer may use to address IQ in the Web of Data. As previously suggested, like a Data Scientist using a machine learning tool such as Weka to build a classifier is afforded the ability to explore different classifiers and parameters to meet their needs, we similarly envision a Quality Knowledge Engineer experimenting with alternative approaches to develop reusable quality components. We therefore see a need for tooling and infrastructure that can not only support the engineer in building quality components, but also benchmark and compare alternative implementations, strategies, data etc. Missier established the Qurator workbench to support the user in developing Quality Views, we might therefore look to build upon this existing infrastructure. This infrastructure can be developed in parallel with the previously suggested work

to establish methodologies to evaluate quality components.

HCI Aspects of the IQ Life-cycle

Much of the current research in Web-based IQ assessment has demonstrated some capability in predicting or evaluating in the quality of information. What is less understood are many of the Human Computer Interaction (HCI) aspects of the IQ assessment problem. Valuable future work for IQ research on the Web of Data (for scientific data, and information in general) would be to investigate this further. These HCI aspects relate to both the engineering process and exploitation.

Quality Knowledge Engineering. We have established an approach to encoding predictive Quality Knowledge using a model of uncertainty. In our particular case we have used Multi-Entity Bayesian Networks. To advance our solution we are faced with the challenge of how to successfully elicit the Quality Knowledge from the expert domain user, and encode it in our chosen uncertainty representation.

Capturing a domain expert's knowledge in a Bayesian Network is a non-trivial task. We might ask whether an alternative uncertainty representation such as Dempster-Shafer, or subjective logic would be better, and what the trade off might be with respect to expressiveness. We believe there is scope to apply work from the field of uncertainty modelling that provide a mechanisms for the evaluation of elicitation schemes for probabilistic models [Wan07]. With this we can design usability experiments that evaluate the ease with which experts can capture their domain knowledge using different representations of uncertainty, and different tooling.

Quality Knowledge Exploitation One of the greatest challenges we foresee in the adoption of any IQ frameworks is that of explanation. Explanation plays a critical part in knowledge engineering and data manipulation that involves some human interrogator [Hor11]. This is also true of IQ assessment frameworks [Biz07]. Valuable future work for both the MIM framework and Evident for example would be to establish approaches to providing human readable explanations for assessment results.

The use of metrics leads to the users or subjects of those metrics to question why and how a result was derived [AdAP10]. An number of existing IQ solutions that we have discussed provide some form of explanation for their

results [Biz07] [MPdS04] [GR02]. To chose a suitable explanation when developing an IQ solution we see the need to understand the impact that type of explanation will have on the perceived quality and trustworthiness of the data. Previous work in the field of HCI for example has developed approaches to better understand this, demonstrating that the perceived credibility information can be altered based upon *how* quality-based metadata is presented to users [SCKP08] [PWS09]. Beyond just the development of IQ solutions to evaluate quality, we therefore see the need for complimentary work to support the effective application of these solutions.

Bibliography

- [ABK⁺07] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. Dbpedia: A Nucleus for A Web of Open Data. In *The semantic web*, pages 722–735. Springer, 2007.
- [ACHZ09] K. Alexander, R. Cyganiak, M. Hausenblas, and J. Zhao. Describing Linked Datasets. In *Proceedings of the Linked Data on the Web (LDOW) Workshop*, 2009.
- [Ada84] J. B. Adams. Probabilistic Reasoning and Certainty Factors. *Rule-Based Expert Systems*, pages 263–271, 1984.
- [AdAP10] B. Adler, L. de Alfaro, and I. Pye. Detecting Wikipedia Vandalism Using WikiTrust. *Notebook Papers of CLEF*, 1:22–23, 2010.
- [AEK00] A. Ansari, S. Essegaier, and R. Kohli. Internet Recommendation Systems. *Journal of Marketing research*, 37(3):363–375, 2000.
- [AG07] D. Artz and Y. Gil. A Survey of Trust in Computer Science and The Semantic Web. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(2):58–71, 2007.
- [AT99] J. E. Alexander and M. A. Tate. *Web Wisdom; How to Evaluate and Create Information Quality on The Web*. CRC Press, May 1999.
- [BAW⁺05] A. Bairoch, R. Apweiler, C. H. Wu, W. C. Barker, B. Boeckmann, S. Ferro, E. Gasteiger, H. Huang, R. Lopez, M. Magrane, M. J. Martin, D. A. Natale, C. O’Donovan, N. Redaschi, and L.-S. L. Yeh. The Universal Protein Resource (UniProt). *Nucleic acids research*, 33(Database issue):D154–9, January 2005.

- [BBR⁺13] S. Bechhofer, I. Buchan, D. D. Roure, P. Missier, J. Ainsworth, J. Bhagat, P. Couch, D. Cruickshank, M. Delderfield, I. Dunlop, M. Gamble, D. Michaelides, S. Owen, D. Newman, S. Sufi, and C. Goble. Why Linked Data Is Not Enough for Scientists. *Future Generation Computer Systems*, 29(2):599 – 611, 2013.
- [BC09] C. Bizer and R. Cyganiak. Quality-driven Information Filtering using the WIQA Policy Framework. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(1):1–10, January 2009.
- [BCFM09] C. Batini, C. Cappiello, C. Francalanci, and a. Maurino. Methodologies for Data Quality Assessment and Improvement. *ACM Computing Surveys*, 41(3):1–52, July 2009.
- [BCG⁺12] K. Belhajjame, O. Corcho, D. Garijo, J. Zhao, P. Missier, D. Newman, R. Palma, S. Bechhofer, et al. Workflow-centric Research Objects: First Class Citizens in Scholarly Discourse. In *Proceedings of the ESWC2012 Workshop on the Future of Scholarly Communication in the Semantic Web*, 2012.
- [BCGM05] C. Bizer, R. Cyganiak, t. Gauss, and O. Maresch. The Triql. P Browser: Filtering Information Using Context-, Content-and Rating-based Trust Policies. In *Proceedings of the Semantic Web and Policy Workshop at the 4th International Semantic Web Conference*, 2005.
- [BDH⁺09] D. Binns, E. Dimmer, R. Huntley, D. Barrell, C. O’Donovan, and R. Apweiler. QuickGO: A Web-based Tool for Gene Ontology Searching. *Bioinformatics*, 25(22):3045–3046, 2009.
- [BDRG⁺10] S. Bechhofer, D. De Roure, M. Gamble, C. Goble, and I. Buchan. Research Objects: Towards Exchange and Reuse of Digital Knowledge. *The Future of the Web for Collaborative Science*, 2010.
- [BEG⁺13] C. Brenninkmeijer, C. Evelo, C. Goble, A. Gray, A. Waagmeester, and E. Willighagen. Dataset Descriptions for The Open Pharmacological Space. <http://www.openphacts.org/specs/datadesc/>, September 2013. [accessed 17/11/2013].

- [BG11] A. Bellenger and S. Gatepaille. Uncertainty in ontologies: Dempster-shafer Theory for Data Fusion Applications. *arXiv preprint arXiv:1106.3876*, 2011.
- [BGG⁺13] S. Bail, B. Glimm, R. Goncalves, E. Jimenez Ruiz, Y. Kazakov, N. Matentzoglou, and B. Parsia, editors. *The Owl Reasoner Evaluation Workshop*, volume 1015. <http://ceur-ws.org>, 2013. [accessed 20/07/2013].
- [BGtUC12] J. Bolleman, S. Gehant, and the UniProt Consortium. Catching Inconsistencies with the Semantic Web: a Biocuration Case Study. In *Proceedings of the 5th International Workshop on Semantic Web Applications and tools for Life Sciences*, volume 952, Paris, France, November 2012. <http://ceur-ws.org/>.
- [BHBL09] C. Bizer, T. Heath, and T. Berners-Lee. Linked Data—The Story So Far. *International Journal on Semantic Web and Information Systems*, 5(3):1–22, 2009.
- [BHPS11] P. A. Bonatti, A. Hogan, A. Polleres, and L. Sauro. Robust and Scalable Linked Data Reasoning Incorporating Provenance and Trust Annotations. *Web Semantics: Science, Services and Agents on the World Wide Web*, 9(2):165–201, 2011.
- [Biz07] C. Bizer. *Quality-driven Information Filtering in the Context of Web-based Information Systems*. PhD thesis, Freie Universität Berlin, 2007.
- [Biz13] C. Bizer. Interlinking Scientific Data on A Global Scale. *Data Science Journal*, 12(0):GRDI6–GRDI12, 2013.
- [BKWC01] P. Buneman, S. Khanna, and T. Wang-Chiew. Why and Where: A Characterization of Data Provenance. In *Database Theory—ICDT 2001*, pages 316–330. Springer, 2001.
- [BL97] T. Berners-Lee. Design Issues - Consistent User Interface. <http://www.w3.org/DesignIssues/UI.html>, September 1997. [accessed 20/11/2013].

- [BL06] T. Berners-Lee. Design Issues - Linked Data. <http://www.w3.org/DesignIssues/LinkedData.html>, July 2006. [accessed 14/10/2013].
- [BLK⁺09] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann. Dbpedia - A Crystallization Point for The Web of Data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(3):154–165, September 2009.
- [BLWC11] J.-c. Bradley, a. Lang, A. Williams, and E. Curtin. ONS Open Melting Point Collection. *Nature Precedings*, pages 1–699, August 2011.
- [BM10] D. Brickley and L. Miller. Foaf Vocabulary Specification 0.98. <http://xmlns.com/foaf/spec/20100809.html>, August 2010. [accessed 05/10/2013].
- [BMC⁺04] P. Biron, A. Malhotra, W. W. W. Consortium, et al. Xml Schema Part 2: Datatypes. *World Wide Web Consortium Recommendation REC-xmlschema-2-20041028*, 2004.
- [BMW⁺08] T. J. Buza, F. M. McCarthy, N. Wang, S. M. Bridges, and S. C. Burgess. Gene ontology Annotation Quality Analysis in Model Eukaryotes. *Nucleic acids research*, 36(2):e12, February 2008.
- [BNt⁺08] F. Belleau, M.-A. Nolin, N. tourigny, P. Rigault, and J. Morissette. Bio2RDF: Towards A Mashup to Build Bioinformatics Knowledge Systems. *Journal of biomedical informatics*, 41(5):706–16, October 2008.
- [Bou12] R. Boulton. Science as an Open Enterprise. *London: Royal Society*, page 104pp, 2012.
- [Bra97] S. Bradner. IETF RFC 2119:Key Words for use in RFCs to Indicate Requirement Levels. <http://www.ietf.org/rfc/rfc2119.txt>, 1997. [accessed 17/07/2012].
- [BS06] C. Batini and M. Scannapieco. *Data Quality: Concepts, Methodologies and Techniques*. Data-Centric Systems and Applications. Springer Berlin Heidelberg, 2006.

- [BS10] C. Bizer and a. Schultz. The R2R Framework: Publishing and Discovering Mappings on The Web. *ISWC Workshop on Consuming Linked Data*, 2010.
- [BSHW06] O. Benjelloun, A. D. Sarma, A. Halevy, and J. Widom. ULDBs: Databases With Uncertainty and Lineage. In *Proceedings of the 32nd international conference on Very large data bases*, pages 953–964. VLDB Endowment, 2006.
- [BTN⁺10] J. Bhagat, F. Tanoh, E. Nzuobontane, T. Laurent, J. Orlowski, M. Roos, K. Wolstencroft, S. Aleksejevs, R. Stevens, S. Pettifer, et al. Biocatalogue: A Universal Catalogue of Web Services for The Life Sciences. *Nucleic acids research*, 38(suppl 2):W689–W694, 2010.
- [Buz] T. J. Buza. GA2GAQ.pl: The GO Annotation Quality Score Perl Script. <http://www.agbase.msstate.edu/cgi-bin/tools/GA2GAQ.cgi>. [accessed 16/09/2013].
- [BZMGPS13] K. Belhajjame, J. Zhao, J. Manuel Gomez-Perez, and S. Sahoo, editors. *Provench Workshop*. ACM, Proceedings of the Joint EDBT/ICDT 2013 Workshops, 2013.
- [CBHS05] J. J. Carroll, C. Bizer, P. Hayes, and P. Stickler. Named Graphs, Provenance and Trust. In *Proceedings of the 14th international conference on World Wide Web*, pages 613–622. ACM, 2005.
- [CCL⁺10] P. C. Costa, K.-C. Chang, K. Laskey, t. Levitt, and W. Sun. High-level Fusion: Issues in Developing a Formal Theory. In *Information Fusion (FUSION), 2010 13th Conference on*, pages 1–8. IEEE, 2010.
- [CCSP07] A. Caro, C. Calero, H. A. Sahraoui, and M. Piattini. A Bayesian Network to Represent A Data Quality Model. *International Journal of Information Quality*, 1(3):272–294, 2007.
- [CCT09] J. Cheney, L. Chiticariu, and W.-C. Tan. *Provenance in Databases: Why, How, and Where*, volume 1. Now Publishers Inc, 2009.

- [CCtA⁺12] A. Callahan, J. Cruz-toledo, P. Ansell, D. Klassen, G. Tumarello, and M. Dumontier. Improved Dataset Coverage and Interoperability With Bio2rdf Release 2. In *SWAT4LS*, 2012.
- [CCTAD13] A. Callahan, J. Cruz-Toledo, P. Ansell, and M. Dumontier. Bio2RDF Release 2: Improved Coverage, Interoperability and Provenance of Life Science Linked Data. In *The Semantic Web: Semantics and Big Data*, pages 200–212. Springer, 2013.
- [CDD⁺04] J. J. Carroll, I. Dickinson, C. Dollin, D. Reynolds, a. Seaborne, and K. Wilkinson. Jena: Implementing the Semantic Web Recommendations. In *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, pages 74–83. ACM, 2004.
- [CDJ⁺10] B. Chen, X. Dong, D. Jiao, H. Wang, Q. Zhu, Y. Ding, and D. J. Wild. Chem2Bio2RDF: A Semantic Framework for Linking and Data Mining Chemogenomic and Systems Chemical Biology Data. *BMC bioinformatics*, 11:255, January 2010.
- [Ceol] D. Ceolin. Annotation trust vocabulary. <http://www.few.vu.nl/~dceolin/annotationTrust.rdf>. [accessed 01/02/2013].
- [CGVH10] D. Ceolin, P. Groth, and W. R. Van Hage. Calculating The Trust of Event Descriptions Using Provenance. *Proceedings of the SWPM*, 2010.
- [CGvH⁺12] D. Ceolin, P. T. Groth, W. R. van Hage, A. Nottamkandath, and W. Fokkink. Trust Evaluation Through User Reputation and Provenance Analysis. In *URSW*, pages 15–26, 2012.
- [Cha05] A. D. Chapman. *Principles of Data Quality*. GBIF, 2005.
- [Che06] T. Chesney. An Empirical Examination of Wikipedia’s Credibility. *First Monday*, 11(11), 2006.
- [CHI] Chipset - Epidemiology toolkit. <https://www.informatics.manchester.ac.uk/mhealthecosystem/projects/current/Pages/CHIPSET.aspx>, [accessed 14/11/2013].

- [cit13] Citeulike Online Reference Manager. <http://www.citeulike.org>, 2013. [accessed 12/10/2013].
- [CJ11] R. Cyganiak and A. Jentzsch. The Linking Open Data Cloud Diagram. lod-cloud.net, September 2011. [accessed 20/12/2013].
- [Cla11] A. Clark. The Real Reason for Junk Chemical Data. <http://cheminf20.org/2011/05/17/the-real-reason-for-junk-chemical-data/>, April 2011. [accessed 10/05/2012].
- [CLC⁺10] R. Carvalho, K. Laskey, P. Costa, M. Ladeira, L. Santos, and S. Matsumoto. Unbbayes: Modeling Uncertainty for Plausible Reasoning in The Semantic Web. *Semantic Web, IN-TECH Publishing, ISBN*, pages 978–953, 2010.
- [CLC13] R. N. Carvalho, K. B. Laskey, and P. C. Costa. Pr-owl 2.0—bridging The Gap to Owl Semantics. In *Uncertainty Reasoning for the Semantic Web II*, pages 1–18. Springer, 2013.
- [CMB⁺04] E. Camon, M. Magrane, D. Barrell, V. Lee, E. Dimmer, J. Maslen, D. Binns, N. Harte, R. Lopez, and R. Apweiler. The Gene Ontology Annotation (GOA) Database: Sharing Knowledge in UniProt With Gene ontology. *Nucleic acids research*, 32(Database issue):D262–6, January 2004.
- [CMM13] J. Cheney, P. Missier, and L. Moreau. Constraints of The Prov Data Model. 2013. [accessed 10/06/2013].
- [CNF12a] D. Ceolin, A. Nottamkandath, and W. Fokkink. Automated Evaluation of Annotators for Museum Collections Using Subjective Logic. In *Trust Management VI*, pages 232–239. Springer, 2012.
- [CNF12b] D. Ceolin, A. Nottamkandath, and W. Fokkink. Subjective Logic Extensions for The Semantic Web. In *URSW*, pages 27–38, 2012.
- [CP12] Y.-W. Cheah and B. Plale. Provenance Analysis: towards Quality Provenance. In *E-Science (e-Science), 2012 IEEE 8th International Conference on*, pages 1–8. IEEE, 2012.

- [CVHF10] D. Ceolin, W. R. Van Hage, and W. Fokkink. A Trust Model to Estimate The Quality of Annotations Using The Web. *WebSci10: extending the frontiers of society on-line*, 2010.
- [DAA12] C. V. Damásio, A. Analyti, and G. Antoniou. Provenance for SPARQL Queries. In *The Semantic Web–ISWC 2012*, pages 625–640. Springer, 2012.
- [DAKPA11] L. De Alfaro, A. Kulshreshtha, I. Pye, and B. T. Adler. Reputation Systems for Open Collaboration. *Communications of the ACM*, 54(8):81–87, 2011.
- [Dat] Data Observation Network for Earth (dataone). <http://www.dataone.org>, [accessed 23/3/2010].
- [DC11] Y. Ding and B. Cronin. Popular and/or Prestigious? Measures of Scholarly Esteem. *Information processing & management*, 47(1):80–96, 2011.
- [DGS09] D. De Roure, C. Goble, and R. Stevens. The Design and Realisation of The myExperiment Virtual Research Environment for Social Sharing of Workflows. *Future Generation Computer Systems*, 25(5):561–567, May 2009.
- [DLBK08] C. Dai, D. Lin, E. Bertino, and M. Kantarcioglu. An Approach to Evaluate Data Trustworthiness used on Data Provenance. In *Secure Data Management*, pages 82–98. Springer, 2008.
- [DPP06] Z. Ding, Y. Peng, and R. Pan. BayesOWL: Uncertainty Modeling in Semantic Web Ontologies. In *Soft Computing in ontologies and Semantic Web*, pages 3–29. Springer, 2006.
- [EB161] “Chemistry”, *Encyclopedia Britannica*, volume 5, page 374. 1961.
- [Edw04] P. N. Edwards. Beyond The Ivory tower. “a Vast Machine”: Standards As Social Technology. *Science (New York, N.Y.)*, 304(5672):827–8, May 2004.
- [EG98] B. Ewing and P. Green. Base-calling of Automated Sequencer Traces Using phred. Ii. Error Probabilities. *Genome research*, 8(3):186–194, 1998.

- [Eki11] S. Ekins. Collaboration Could Give Us a Gold Standard Database of Drugs. <https://web.archive.org/web/20131227104632/http://www.collabchem.com/2011/04/29/collaboration-could-give-us-a-gold-standard-database-of-drugs/>, April 2011. [accessed 08/11/2013].
- [EM02] M. J. Eppler and P. Muenzenmayer. Measuring Information Quality in The Web Context. In *Proceedings of the 7th International Conference on Information Quality (ICIQ-02)*, pages 187–196, 2002.
- [F⁺98] FGDC et al. Geospatial Positioning Accuracy Standards, Part 3: National Standard for Spatial Data Accuracy. *Washington, DC, Federal Geographic Data Committee, 25p*, 1998.
- [FH10] C. Furber and M. Hepp. Using Semantic Web Resources for Data Quality Management. *Knowledge Engineering and Management by the Masses*, 6317:211–225, 2010.
- [FH11] C. Fürber and M. Hepp. towards A Vocabulary for Data Quality Management in Semantic Web Architectures. In *Proceedings of the 1st International Workshop on Linked Web Data Management*, pages 1–8. ACM, 2011.
- [FK05] W. B. Frakes and K. Kang. Software Reuse Research: Status and Future. *Software Engineering, IEEE Transactions on*, 31(7):529–536, 2005.
- [FLNP00] N. Friedman, M. Linial, I. Nachman, and D. Pe’er. Using Bayesian Networks to Analyze Expression Data. *Journal of computational biology*, 7(3-4):601–620, 2000.
- [FMF⁺12] A. D. Fant, E. Muratov, D. Fourches, D. Sharpe, A. J. Williams, and A. Tropsha. On The Accuracy of Chemical Structures Found on The Internet. [figshare.http://dx.doi.org/10.6084/m9.figshare.662308](http://dx.doi.org/10.6084/m9.figshare.662308), figshare, 2012. [accessed 10/10/2013].
- [FSC⁺09] D. Field, S.-A. Sansone, A. Collis, T. Booth, P. Dukes, S. K.

- Gregurick, K. Kennedy, P. Kolar, E. Kolker, M. Maxon, S. Millard, A.-M. Mugabushaka, N. Perrin, J. E. Remacle, K. Remington, P. Rocca-Serra, C. F. Taylor, M. Thorley, B. Tiwari, and J. Wilbanks. Megascience. 'Omics Data Sharing. *Science (New York, N.Y.)*, 326(5950):234–6, October 2009.
- [FT96] W. Frakes and C. Terry. Software Reuse: Metrics and Models. *ACM Computing Surveys (CSUR)*, 28(2):415–435, 1996.
- [GA07] Y. Gil and D. Artz. Towards Content Trust of Web Resources. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(4):227–239, 2007.
- [Gam12] M. Gamble. wikipedia-provenance extended GitHub repository. <https://github.com/matthewgamble/wikipedia-provenance>, 2012. [accessed 20/12/2013].
- [GCdAS11] B. M. Good, E. L. Clarke, L. de Alfaro, and a. I. Su. The Gene Wiki In 2011: Community Intelligence Applied to Human Gene Annotation. *Nucleic Acids Research*, 2011.
- [GCG⁺10] Y. Gil, J. Cheney, P. Groth, O. Hartig, S. Miles, L. Moreau, P. P. Da Silva, S. Coppens, D. Garijo, J. Gomez, et al. Provenance Xg Final Report. <http://www.w3.org/2005/Incubator/prov/XGR-prov-20101214/>, December 2010. [accessed 11/10/2013].
- [GCL13] A. Gerber, T. W. Cole, and D. Lowery. Using Sparql to Validate Open Annotation Rdf Graphs. In *RDF Validation Workshop: Practical Assurances for Quality RDF Data*, September 2013.
- [GE02] P. Green and B. Ewing. Phred command line tool: Documentation. <http://www.phrap.org/phredphrap/phred.html>, 2002. [accessed 03/07/2013].
- [GG11a] M. Gamble and C. Goble. Quality, Trust, and Utility of Scientific Data on The Web: Towards A Joint Model. *Proceedings of the ACM WebSci'11*, pages 14–17, 2011.

- [GG11b] Y. Gil and P. Groth. Using Provenance in the Semantic Web. *Web Semantics: Science, Services and Agents on the World Wide Web*, 9(2):147–148, 2011.
- [GGCM12] P. Groth, Y. Gil, J. Cheney, and S. Miles. Requirements for Provenance on The Web. *International Journal of Digital Curation*, 7(1):39–56, 2012.
- [GGL⁺12] A. J. Gray, P. Groth, A. Loizou, S. Askjaer, C. Brenninkmeijer, K. Burger, C. Chichester, C. T. Evelo, C. Goble, L. Harland, et al. Applying Linked Data Approaches to Pharmacology: Architectural Decisions and Implementation. *Semantic Web Journal*, 2012.
- [GGV10] P. Groth, a. Gibson, and J. Velterop. The Anatomy of A Nanopublication. *Information Services and Use*, 30(1):51–56, 2010.
- [GH06] J. Golbeck and J. Hendler. Filmtrust: Movie Recommendations Using Trust in Web-based Social Networks. In *Proceedings of the IEEE Consumer communications and networking conference*, volume 96. Citeseer, 2006.
- [Gil05] J. Giles. Internet Encyclopaedias Go Head to Head. *Nature*, 438(7070):900–901, 2005.
- [Gio07] R. Giordano. The Scientist: Secretive, Selfish Or Reticent? A Social Network Analysis. In *E-Social Science conference*, Ann Arbor, MI, 2007.
- [GKCF13] F. Geerts, G. Karvounarakis, V. Christophides, and I. Fundulaki. Algebraic Structures for Capturing The Provenance of SPARQL Queries. In *Proceedings of the 16th International Conference on Database Theory, ICDT '13*, pages 153–164, New York, NY, USA, 2013. ACM.
- [GM09] P. Groth and L. Moreau. Recording Process Documentation for Provenance. *Parallel and Distributed Systems, IEEE Transactions on*, 20(9):1246–1259, 2009.

- [GM13] P. Groth and L. Moreau. An Overview of The Prov Family of Documents. <http://www.w3.org/TR/prov-overview/>, April 2013. [accessed 20/09/2013].
- [GMM06] P. Groth, S. Miles, and S. Munroe. Principles of High Quality Documentation for Provenance: A Philosophical Discussion. In *Provenance and Annotation of Data*, pages 278–286. Springer, 2006.
- [GMM⁺09] P. Groth, S. Miles, S. Modgil, N. Oren, M. Luck, and Y. Gil. Determining The Trustworthiness of New Electronic Contracts. In *Engineering Societies in the Agents World X*, pages 132–147. Springer, 2009.
- [Gob02] C. Goble. Position Statement: Musings on Provenance, Workflow and (Semantic Web) Annotations for Bioinformatics. In *Workshop on Data Derivation and Provenance, Chicago*, 2002.
- [GOF] The Gene Ontology FTP Archive. <ftp.geneontology.org:/pub/go/>. [accessed 16/09/2013].
- [Gol05] J. Golbeck. *Computing and Applying Trust In Web Based Social Networks*. Phd, University of Maryland, 2005.
- [Gol06] J. Golbeck. Trust on the World Wide Web: A Survey. *Foundations and Trends in Web Science*, 1(2):131–197, 2006.
- [Gol08] J. Golbeck. Weaving A Web of Trust. *Science*, 321(5896):1640–1641, 2008.
- [GR02] Y. Gil and V. Ratnakar. Trusting Information Sources one Citizen at a Time. In *The Semantic Web—ISWC 2002*, pages 162–176. Springer, 2002.
- [Har09a] O. Hartig. Provenance Information In The Web of Data. In *In Proceedings of the Linked Data on the Web Workshop*, 2009.
- [Har09b] O. Hartig. Querying Trust in Rdf Data With tSPARQL. In *The Semantic Web: Research and Applications*, pages 5–20. Springer, 2009.

- [Har10] O. Hartig. towards A Data-centric Notion of Trust in The Semantic Web. In *2nd Workshop on Trust and Privacy on the Social and Semantic Web SPOT2010, Heraklion (Greece)*, volume 110, 2010.
- [HDG⁺06] M. Horridge, N. Drummond, J. Goodwin, A. L. Rector, R. Stevens, and H. Wang. The Manchester OWL Syntax. In *OWLed*, volume 216, 2006.
- [HFH⁺09] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The Weka Data Mining Software: An Update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18, 2009.
- [HHP⁺10] A. Hogan, A. Harth, A. Passant, S. Decker, and A. Polleres. Weaving the Pedantic Web. In *Proceedings of the Linked Data on the Web (LDOW) Workshop 2010*. CEUR, 2010.
- [HIOG⁺08] J. W. Huss III, C. Orozco, J. Goodale, C. Wu, S. Batalov, T. J. Vickers, F. Valafar, and a. I. Su. A Gene Wiki for Community Annotation of Gene Function. *PLoS biology*, 6(7):e175, 2008.
- [HLOD10] L. L. Haines, J. Light, D. O’Malley, and F. A. Delwiche. Information-seeking Behavior of Basic Science Researchers: Implications for Library Services. *Journal of the Medical Library Association: JMLA*, 98(1):73, 2010.
- [HLVA07] B. M. Hemminger, D. Lu, K. Vaughan, and S. J. Adams. Information Seeking Behavior of Academic Scientists. *Journal of the American Society for Information Science and Technology*, 58(14):2205–2225, 2007.
- [Hor11] M. Horridge. *Justification Based Explanation In ontologies*. PhD thesis, the University of Manchester, 2011.
- [HPK08] D. Hull, S. R. Pettifer, and D. B. Kell. Defrosting the Digital Library: Bibliographic Tools for The Next Generation Web. *PLoS Comput Biol*, 4(10):e1000204, 10 2008.
- [HR08] V. Henning and J. Reichelt. Mendeley-a Last. Fm for Research? In *eScience, 2008. eScience’08. IEEE Fourth International Conference on*, pages 327–328. IEEE, 2008.

- [HSW⁺11] R. Huang, N. Southall, Y. Wang, A. Yasgar, P. Shinn, A. Jadhav, D.-T. Nguyen, and C. P. Austin. The NCGC Pharmaceutical Collection: A Comprehensive Resource of Clinically Approved Drugs Enabling Repurposing and Chemical Genomics. *Science Translational Medicine*, 3(80):80ps16, 2011.
- [HT03] A. J. Hey and A. Trefethen. *The Data Deluge: An e-science Perspective*. Wiley and Sons, 2003.
- [HTT09] T. Hey, S. Tansley, and K. Tolle, editors. *The Fourth Paradigm: Data-Intensive Scientific Discovery*. Microsoft Research, Redmond, Washington, 2009.
- [HUH⁺12] A. Hogan, J. Umbrich, a. Harth, R. Cyganiak, A. Polleres, and S. Decker. An Empirical Survey of Linked Data Conformance. *Web Semantics: Science, Services and Agents on the World Wide Web*, 14:14–44, 2012.
- [HWZW13] C. Haupt, a. Waagmeester, M. Zimmermann, and E. Wilhagen. Guidelines for Exposing Data as RDF in Open PHACTS. <http://www.openphacts.org/specs/2013/WD-rdfguide-20131007/>, October 2013. [accessed 27/12/2013].
- [HZ09] O. Hartig and J. Zhao. Using Web Data Provenance for Quality Assessment. In *Proceedings of the First International Workshop on the role of Semantic Web in Provenance Management at ISWC2009*, Washington D.C., 2009.
- [HZ10] O. Hartig and J. Zhao. Publishing and Consuming Provenance Metadata on The Web of Linked Data. In *Provenance and Annotation of Data and Processes*, pages 78–90. Springer, 2010.
- [ISO08] E. ISO. 9001: 2008. *Quality management systems—Requirements (ISO)*, 9001, 2008.
- [ISO11] I. ISO. Iec 25010: 2011: Systems and Software Engineering—Systems and Software Quality Requirements and Evaluation (SQUARE)—system and Software Quality Models. *International Organization for Standardization*, 2011.

- [IUP04] IUPAC Provisional Recommendations 4–7. Preferred iupac names., IUPAC, 2004.
- [JIB07] A. Jøsang, R. Ismail, and C. Boyd. A Survey of Trust and Reputation Systems for Online Service Provision. *Decision support systems*, 43(2):618–644, 2007.
- [JL94] F. V. Jensen and J. Liang. Drhugin A System for Value of Information In Bayesian Networks. 1994.
- [JN07] F. V. Jensen and T. D. Nielsen. *Bayesian Networks and Decision Graphs*. Springer, 2007.
- [Jøs97] A. Jøsang. Artificial Reasoning with Subjective Logic. In *Proceedings of the Second Australian Workshop on Commonsense Reasoning*, volume 48. Perth, 1997.
- [Jøs08] A. Jøsang. Conditional Reasoning with Subjective Logic. *Journal of Multiple-Valued Logic and Soft Computing*, 15(1):5–38, 2008.
- [JQJ98] M. A. Jeusfeld, C. Quix, and M. Jarke. Design and Analysis of Quality Information for Data Warehouses. In *Proceedings of the 17th International Conference on Conceptual Modeling*, number 1, 1998.
- [JSC13] G. Jiang, H. Solbrig, and C. Chute. A Semantic Web-based Framework for Quality Assurance of Electronic Medical Records Data for Secondary Use. <https://www.w3.org/2012/12/rdf-val/>, September 2013. [accessed 01/11/2013].
- [Jur74] J. M. Juran. *Quality Control Handbook*. Mcgraw-Hill (Tx), 3rd edition, 1974.
- [KB05] S. A. Knight and J. Burn. Developing A Framework for Assessing Information Quality on the World Wide Web. *Informing Science*, 8, 2005.
- [KF09] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. The MIT Press, 2009.

- [KFS⁺10] C. Kettner, D. Field, S.-A. Sansone, C. Taylor, J. Aerts, N. Binns, a. Blake, C. M. Britten, A. de Marco, J. Fostel, P. Gaudet, A. González-Beltrán, N. Hardy, J. Hellemans, H. Hermjakob, N. Juty, J. Leebens-Mack, E. Maguire, S. Neumann, S. Orchard, H. Parkinson, W. Piel, S. Ranganathan, P. Rocca-Serra, A. Santarsiero, D. Shotton, P. Sterk, a. Untergasser, and P. L. Whetzel. Meeting Report From The Second “Minimum Information for Biological and Biomedical Investigations” (MIBBI) Workshop. *Standards in genomic sciences*, 3(3):259–66, January 2010.
- [KFW08] K. Kelton, K. R. Fleischmann, and W. A. Wallace. Trust In Digital Information. *Journal of the American Society for Information Science and Technology*, 59(3):363–374, February 2008.
- [Knu11a] H. Knublauch. Spin - Modeling Vocabulary. W3C Member Submission. <http://www.w3.org/Submission/spin-modeling/>, February 2011. [accessed 11/02/2013].
- [Knu11b] H. Knublauch. The TopBraid SPIN API. <http://topbraid.org/spin/api/>, May 2011. [accessed 05/09/2011].
- [Kol99] D. Koller. Probabilistic Relational Models. In *Inductive logic programming*, pages 3–13. Springer, 1999.
- [KP97] D. Koller and A. Pfeffer. Object-oriented Bayesian Networks. In *Proceedings of the Thirteenth conference on Uncertainty in artificial intelligence*, pages 302–313. Morgan Kaufmann Publishers Inc., 1997.
- [KSW02] B. K. Kahn, D. M. Strong, and R. Y. Wang. Information Quality Benchmarks: Product and Service Performance. *Communications of the ACM*, 45(4ve):184–192, April 2002.
- [KZAL13] D. Kontokostas, A. Zaveri, S. Auer, and J. Lehmann. Triplecheckmate: A Tool for Crowdsourcing The Quality Assessment of Linked Data. In *Knowledge Engineering and the Semantic Web*, pages 265–272. Springer, 2013.

- [Las08] K. B. Laskey. MEBN: A Language for First-order Bayesian Knowledge Bases. *Artificial Intelligence*, 172(2):140–178, 2008.
- [LMS⁺05] P. Lord, A. Macdonald, R. Sinnott, D. Ecklund, M. Westhead, and a. Jones. Large-scale Data Sharing In The Life Sciences: Data Standards, Incentives, Barriers and Funding Models. Technical report, 2005.
- [LOD] Linking Open Data Project. <http://www.w3.org/wiki/SweoIG/Taskforces/CommunityProjects/LinkingOpenData>. [accessed 10/10/2013].
- [LOD13] LODRefine Data Intergration tool. <http://code.zemanta.com/sparkica/>, 2013. [accessed 11/10/2013].
- [LSB⁺08] J. a. Lee, J. Spidlen, K. Boyce, J. Cai, N. Crosbie, M. Dalphin, J. Furlong, M. Gasparetto, M. Goldberg, E. M. Goralczyk, B. Hyun, K. Jansen, t. Kollmann, M. Kong, R. Leif, S. McWeeney, T. D. Moloshok, W. Moore, G. Nolan, J. Nolan, J. Nikolich-Zugich, D. Parrish, B. Purcell, Y. Qian, B. Selvaraj, C. Smith, O. Tchuvatkina, A. Wertheimer, P. Wilkinson, C. Wilson, J. Wood, R. Zigon, R. H. Scheuermann, and R. R. Brinkman. MiFlowCyt: The Minimum Information About A Flow Cytometry Experiment. *Cytometry. Part A : the journal of the International Society for Analytical Cytology*, 73(10):926–30, October 2008.
- [LSKW02] Y. W. Lee, D. M. Strong, B. K. Kahn, and R. Y. Wang. AIMQ: A Methodology for Information Quality Assessment. *Information & Management*, 40(2):133 – 146, 2002.
- [LSM⁺13] T. Lebo, S. Sahoo, D. McGuinness, K. Belhajjame, J. Cheney, D. Corsar, D. Garijo, S. Soiland-Reyes, S. Zednik, and J. Zhao. Prov-O: The Prov Ontology. <http://www.w3.org/TR/prov-o/>, April 2013. [accessed 03/09/2013].
- [LVdSJ⁺07] C. Lagoze, H. Van de Sompel, P. Johnston, M. L. Nelson, R. Sanderson, and S. Warner. Open Archives Initiative Object

- Reuse and Exchange (OAI-ORE). Technical report, Technical report, Open Archives Initiative, 2007.
- [Mar94] S. P. Marsh. *Formalising Trust as a Computational Concept*. PhD thesis, University of Stirling, April 1994.
- [MBD⁺12] M. S. Marshall, R. Boyce, H. F. Deus, J. Zhao, E. L. Willighagen, M. Samwald, E. Pichler, J. Hajagos, E. Prud'hommeaux, and S. Stephens. Emerging Practices for Mapping and Linking Life Sciences Data Using RDF — A Case Series. *Web Semantics: Science, Services and Agents on the World Wide Web*, 14:2–13, July 2012.
- [MBKZ⁺11] E. Meyer, M. Bulger, A. Kyriakidou-Zacharoudiou, L. Power, P. Williams, W. Venters, M. Terras, and S. Wyatt. Collaborative Yet Independent: Information Practices in The Physical Sciences. *Research Information Network (RIN) Report Series, IOP Publishing*, 2011.
- [MC13] P. Missier and Z. Chen. Extracting PROV Provenance Traces from Wikipedia History Pages. In *Proceedings of the Joint EDBT/ICDT 2013 Workshops*, pages 327–330. ACM, 2013.
- [MCF⁺11] L. Moreau, B. Clifford, J. Freire, J. Futrelle, Y. Gil, P. Groth, N. Kwasnikowska, S. Miles, P. Missier, J. Myers, et al. The Open Provenance Model Core Specification (v1. 1). *Future Generation Computer Systems*, 27(6):743–756, 2011.
- [MG11] L. Moreau and P. Groth. Provenance Interchange Working Group Charter. <http://www.w3.org/2011/01/prov-wg-charter#about>, January 2011. [accessed 07/11/13].
- [MGBM07] S. Miles, P. Groth, M. Branco, and L. Moreau. The Requirements of Using Provenance In e-Science Experiments. *Journal of Grid Computing*, 5(1):1–25, 2007.
- [MH05] A. Martinez and J. Hammer. Making Quality Count in Biological Data Sources. In *Proceedings of the 2nd international workshop on Information quality in information systems - IQIS '05*, page 16, New York, New York, USA, 2005. ACM Press.

- [MIA06] Miare: Minimum Information About An RNAi Experiment. <http://miare.sourceforge.net/MIAREReportingGuidelines>, 2006. [accessed 17/07/2012].
- [Mis08] P. Missier. *Modelling and Computing The Quality of Information In e-science*. PhD thesis, The University of Manchester, 2008.
- [Mis12] P. Missier. wikipedia-provenance GitHub repository. <https://github.com/PaoloMissier/wikipedia-provenance>, 2012. [accessed 07/12/2013].
- [MMB12] P. N. Mendes, H. Mühleisen, and C. Bizer. Sieve: Linked Data Quality Assessment and Fusion. In *Proceedings of the 2012 Joint EDBT/ICDT Workshops*, pages 116–123. ACM, 2012.
- [MNJ05] P. Mitra, N. F. Noy, and A. R. Jaiswal. Omen: A Probabilistic Ontology Mapping tool. In *The Semantic Web—ISWC 2005*, pages 537–547. Springer, 2005.
- [Mor10] L. Moreau. The Foundations for Provenance on The Web. *Foundations and Trends in Web Science*, 2(2–3):99–241, 2010.
- [Mor12] L. Moreau. The prov toolbox github repository. <https://github.com/lucmoreau/ProvToolbox/>, 2012. [accessed 20/12/2013].
- [MPdS04] D. L. McGuinness and P. Pinheiro da Silva. Explaining Answers from the Semantic Web: The Inference Web Approach. *Web Semantics: Science, Services and Agents on the World Wide Web*, 1(4):397–413, 2004.
- [MPPG09] V. Momtchev, D. Peychev, t. Primov, and G. Georgiev. Expanding The Pathway and Interaction Knowledge In Linked Life Data. *ontotextcom*, 2009.
- [MR08] P. Murray-Rust. Chemistry for Everyone. *Nature*, 451(7179):648–51, February 2008.
- [MRV00] G. A. Mihaila, L. Raschid, and M.-E. Vidal. Using Quality of Data Metadata for Source Selection and Ranking. In *WebDB (Informal Proceedings)*, pages 93–98, 2000.

- [MVH⁺04] D. L. McGuinness, F. Van Harmelen, et al. OWL Web Ontology Language Overview. *W3C recommendation*, 10(2004-03):10, 2004.
- [MZdS⁺06] D. L. McGuinness, H. Zeng, P. P. da Silva, L. Ding, D. Narayanan, and M. Bhaowal. Investigations Into Trust for Collaborative Information Repositories: A Wikipedia Case Study. In *MTW*, 2006.
- [Nat08a] Big Data Special. *Nature*, 455(7209), September 2008.
- [Nat08b] Community Cleverness Required. *Nature*, 455(7209):1, September 2008.
- [Nau02] F. Naumann. *Quality-driven Query Answering for Integrated Information Systems*, volume 2261 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2002.
- [NHL⁺10] X. Niu, B. M. Hemminger, C. Lown, S. Adams, C. Brown, A. Level, M. McLure, A. Powers, M. R. Tennant, and T. Cataldo. National Study of Information Seeking Behavior of Academic Researchers In The United States. *Journal of the American Society for Information Science and Technology*, 61(5):869–890, 2010.
- [Nik00] D. Nikovski. Constructing Bayesian Networks for Medical Diagnosis from Incomplete and Partially Correct Statistics. *Knowledge and Data Engineering, IEEE Transactions on*, 12(4):509–516, 2000.
- [NR00] F. Naumann and C. Rolker. Assessment Methods for Information Quality Criteria. In B. D. Klein and D. F. Rossin, editors, *IQ*, pages 148–162. MIT, 2000.
- [NRHW06] N. Noy, A. Rector, P. Hayes, and C. Welty. Defining N-ary Relations on The Semantic Web. *W3C Working Group Note*, 12:4, 2006.
- [OALB⁺11] S. Orchard, B. Al-Lazikani, S. Bryant, D. Clark, E. Calder, I. Dix, O. Engkvist, M. forster, A. Gaulton, M. Gilson, R. Glen, M. Grigorov, K. Hammond-Kosack, L. Harland, a. Hopkins, C. Larminie, N. Lynch, R. K. Mann, P. Murray-Rust, E. Lo

- Piparo, C. Southan, C. Steinbeck, D. Wishart, H. Hermjakob, J. Overington, and J. Thornton. Minimum Information About a Bioactive Entity (MIABE). *Nature reviews. Drug discovery*, 10(9):661–9, September 2011.
- [OH08] K. O’Hara and W. Hall. Trust on The Web: Some Web Science Research Challenges. *UoC Papers: E-Journal on the Knowledge Society*, (7), 2008.
- [OOO⁺02] T. Oprea, M. Olah, L. Ostopovici, R. Rad, and M. Mracec. On The Propagation of Errors In The QSAR Literature. *Euro QSAR*, 2002.
- [Ope12] Openwetware. http://openwetware.org/wiki/Main_Page, [accessed 01/03/2012].
- [PAG06] J. Pérez, M. Arenas, and C. Gutierrez. Semantics and Complexity of Sparql. In *The Semantic Web-ISWC 2006*, pages 30–43. Springer, 2006.
- [PGT12] J. Priem, P. Groth, and D. Taraborelli. The Altmetrics Collection. *PLoS ONE*, 7(11):e48753, 11 2012.
- [PLW02] L. L. Pipino, Y. W. Lee, and R. Y. Wang. Data Quality Assessment. *Communications of the ACM*, 45(4ve):211–218, April 2002.
- [PME⁺08] A. Preece, P. Missier, S. Embury, B. Jin, and M. Greenwood. An ontology-based approach to handling information quality in e-science. *Concurrency and computation: practice and experience*, 20(3):253–264, 2008.
- [PPH12] J. Priem, H. A. Piwowar, and B. M. Hemminger. Altmetrics in the Wild: Using Social Media to Explore Scholarly Impact. *arXiv preprint arXiv:1203.4745*, 2012.
- [Pru13a] E. Prud’hommeaux. RDF Validation Workshop: Practical Assurances for Quality RDF Data. <https://www.w3.org/2012/12/rdf-val/>, September 2013. [accessed 01/10/2011].

- [Pru13b] E. Prud'hommeaux. Shape Expressions Primer. <http://www.w3.org/2013/ShEx/Primer>, September 2013. [accessed 27/12/2013].
- [PSS⁺05] H. Parkinson, U. Sarkans, M. Shojatalab, N. Abeygunawardena, S. Contrino, R. Coulson, A. Farne, G. Garcia Lara, E. Holloway, M. Kapushesky, P. Lilja, G. Mukherjee, A. Oezcimen, T. Rayner, P. Rocca-Serra, A. Sharma, S. Sansone, and A. Brazma. Array-Express – A Public Repository for Microarray Gene Expression Data At The Ebi. *Nucl. Acids Res.*, 33(suppl.1):D553–555, January 2005.
- [PUSC12] H. Pérez-Urbina, E. Sirin, and K. Clark. Validating RDF with Owl Integrity Constraints. <http://docs.stardog.com/icv/icv-specification.html>, 2012. [accessed 27/11/2013].
- [PW10] H. E. Pence and A. Williams. Chempider: An online Chemical Information Resource. *Journal of Chemical Education*, 87(11):1123–1124, November 2010.
- [PWS09] P. Pirolli, E. Wollny, and B. Suh. So You Know You're Getting The Best Possible Information: A Tool that Increases Wikipedia Credibility. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1505–1508. ACM, 2009.
- [Qui] QuickGO: Datasets. <http://www.ebi.ac.uk/QuickGO/Dataset.html>. [accessed 20/07/2012].
- [Red01] T. C. Redman. *Data Quality: The Field Guide*. Digital Press, 2001.
- [RGH12] A. A. Romero, B. C. Grau, and I. Horrocks. More: Modular Combination of Owl Reasoners for ontology Classification. In *The Semantic Web–ISWC 2012*, pages 1–16. Springer, 2012.
- [RJS11] O. Reichman, M. B. Jones, and M. P. Schildhauer. Challenges and Opportunities of Open Data In Ecology. *Science(Washington)*, 331(6018):703–705, 2011.
- [RLHS13] A. G. Ryman, A. J. Le Hors, and S. Speicher. OSLC Resource Shape A Language for Defining Constraints on Linked Data. In

RDF Validation Workshop: Practical Assurances for Quality RDF Data, September 2013.

- [RSBM⁺10] P. Rocca-Serra, M. Brandizi, E. Maguire, N. Sklyar, C. Taylor, K. Begley, D. Field, S. Harris, W. Hide, O. Hofmann, S. Neumann, P. Sterk, W. tong, and S.-A. Sansone. ISA Software Suite: Supporting Standards-compliant Experimental Annotation and Enabling Curation at the Community Level. *Bioinformatics (Oxford, England)*, 26(18):2354–6, September 2010.
- [SAD12] N. Skunca, A. Altenhoff, and C. Dessimoz. Quality of Computationally Inferred Gene ontology Annotations. *PLoS Computational Biology*, 8(5):e1002533, 2012.
- [SAR⁺07] B. Smith, M. Ashburner, C. Rosse, J. Bard, W. Bug, W. Ceusters, L. J. Goldberg, K. Eilbeck, A. Ireland, C. J. Mungall, N. Leontis, P. Rocca-Serra, A. Ruttenberg, S.-A. Sansone, R. H. Scheuermann, N. Shah, P. L. Whetzel, and S. Lewis. The OBO Foundry: Coordinated Evolution of Ontologies to Support Biomedical Data Integration. *Nature biotechnology*, 25(11):1251–5, November 2007.
- [SB13] S. Simister and D. Brickley. Simple Application-specific Constraints for Rdf Models. In *RDF Validation Workshop: Practical Assurances for Quality RDF Data*, Cambridge, Massachusetts, September 2013.
- [SC10] W. Sun and K. Chang. Direct Message Passing for Hybrid Bayesian Networks and Performance Analysis. In *SPIE Defense, Security, and Sensing*, pages 76970S–76970S. International Society for Optics and Photonics, 2010.
- [SCC97] K. a. Spackman, K. E. Campbell, and R. a. Côté. SNOMED Rt: A Reference Terminology for Health Care. *Proceedings : a conference of the American Medical Informatics Association, AMIA Fall Symposium*, pages 640–4, January 1997.
- [Sch06] T. J. Scheff. *Goffman Unbound!: A New Paradigm for Social Science*. Paradigm Publishers Boulder, 2006.

- [Sci11] Special Issue: Dealing With Data. *Science*, 331(6018):639–806, February 2011.
- [SCKP08] B. Suh, E. H. Chi, A. Kittur, and B. A. Pendleton. Lifting The Veil: Improving Accountability and Social Transparency in Wikipedia with Wikidashboard. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1037–1040. ACM, 2008.
- [SGTS07] B. Stvilia, L. Gasser, M. B. Twidale, and L. C. Smith. A Framework for Information Quality Assessment. *Journal of the American Society for Information Science and Technology*, 58(12):1720–1733, 2007.
- [Sha76] G. Shafer. *A Mathematical Theory of Evidence*, volume 1. Princeton university press Princeton, 1976.
- [Sho10] D. Shotton. Introduction The Semantic Publishing and Referencing (SPAR) Ontologies. October 14, 2010. <http://opencitations.wordpress.com/2010/10/14/introducing-the-semantic-publishingand-referencing-spar-ontologies>, 2010. [accessed 20/09/2013].
- [sin09] Sindice: The Semantic Web Index. Internet Archive. <http://web.archive.org/web/20090217221221/http://sindice.com/>, February 2009. [accessed 01/02/2013].
- [SK12] S. Schlobach and C. a. Knoblock. Dealing With The Messiness of The Web of Data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 14:1, July 2012.
- [SMH08] R. Shearer, B. Motik, and I. Horrocks. Hermit: A Highly-efficient Owl Reasoner. In *OWLED*, volume 432, 2008.
- [SMI⁺11] a. Schultz, a. Matteini, R. Isele, C. Bizer, and C. Becker. LDIF-linked Data Integration Framework. *COLD*, 782, 2011.
- [SPG05a] Y. L. Simmhan, B. Plale, and D. Gannon. A Survey of Data Provenance In e-science. *ACM Sigmod Record*, 34(3):31–36, 2005.

- [SPG05b] Y. L. Simmhan, B. Plale, and D. Gannon. A Survey of Data Provenance Techniques. *Computer Science Department, Indiana University, Bloomington IN*, 47405, 2005.
- [SRSB⁺08] S.-A. Sansone, P. Rocca-Serra, M. Brandizi, A. Brazma, D. Field, J. Fostel, a. G. Garrow, J. Gilbert, F. Goodsaid, N. Hardy, P. Jones, A. Lister, M. Miller, N. Morrison, T. Rayner, N. Sklyar, C. Taylor, W. tong, G. Warner, and S. Wiemann. The First RSBI (ISA-tab) Workshop: "Can A Simple Format Work for Complex Studies?". *Omics : a journal of integrative biology*, 12(2):143–9, June 2008.
- [SRSF⁺12] S.-a. Sansone, P. Rocca-Serra, D. Field, E. Maguire, C. Taylor, O. Hofmann, H. Fang, S. Neumann, W. tong, L. Amaral-Zettler, K. Begley, T. Booth, L. Bougueleret, G. Burns, B. Chapman, T. Clark, L.-a. Coleman, J. Copeland, S. Das, A. de Daruvar, P. de Matos, I. Dix, S. Edmunds, C. T. Evelo, M. J. forster, P. Gaudet, J. Gilbert, C. Goble, J. L. Griffin, D. Jacob, J. Kleinjans, L. Harland, K. Haug, H. Hermjakob, S. J. H. Sui, A. Laederach, S. Liang, S. Marshall, A. McGrath, E. Merrill, D. Reilly, M. Roux, C. E. Shamu, C. A. Shang, C. Steinbeck, A. Trefethen, B. Williams-Jones, K. Wolstencroft, I. Xenarios, and W. Hide. Toward Interoperable Bioscience Data. *Nature Genetics*, 44(2):121–126, January 2012.
- [Sta] Stardog: The RDF Database. <http://www.stardog.com/>. [accessed 17/05/2013].
- [Sti50] G. J. Stigler. The Development of Utility Theory. I. *Journal of Political Economy*, 58(4):pp. 307–327, 1950.
- [Su09] a. I. Su. The Gene Wiki – Portal. http://en.wikipedia.org/wiki/Portal:Gene_Wiki, May 2009. [accessed 10/05/2012].
- [SVM⁺04] M. Scannapieco, A. Virgillito, C. Marchetti, M. Mecella, and R. Baldoni. The Architecture: A Platform for Exchanging and Improving Data Quality In Cooperative Information Systems1. *Information Systems*, 29(7):551–582, October 2004.

- [TFS⁺08] C. F. Taylor, D. Field, S.-A. Sansone, J. Aerts, R. Apweiler, M. Ashburner, C. a. Ball, P.-A. Binz, M. Bogue, T. Booth, A. Brazma, R. R. Brinkman, A. Michael Clark, E. W. Deutsch, O. Fiehn, J. Fostel, P. Ghazal, F. Gibson, T. Gray, G. Grimes, J. M. Hancock, N. W. Hardy, H. Hermjakob, R. K. Julian, M. Kane, C. Kettner, C. Kinsinger, E. Kolker, M. Kuiper, N. Le Novère, J. Leebens-Mack, S. E. Lewis, P. Lord, A.-M. Mallon, N. Marthandan, H. Masuya, R. McNally, A. Mehrle, N. Morrison, S. Orchard, J. Quackenbush, J. M. Reecy, D. G. Robertson, P. Rocca-Serra, H. Rodriguez, H. Rosenfelder, J. Santoyo-Lopez, R. H. Scheuermann, D. Schober, B. Smith, J. Snape, C. J. Stoeckert, K. Tipton, P. Sterk, a. Untergasser, J. Vandesompele, and S. Wiemann. Promoting Coherent Minimum Reporting Guidelines for Biological and Biomedical Investigations: The MIBBI Project. *Nature biotechnology*, 26(8):889–96, August 2008.
- [Top] TopBraid Composer. http://www.topquadrant.com/products/TB_Composer.html. [accessed 27/12/2013].
- [TSBM10] J. Tao, E. Sirin, J. Bao, and D. L. McGuinness. Integrity Constraints in Owl. In *AAAI*, 2010.
- [VA07] L.-H. Vu and K. Aberer. A Probabilistic Framework for Decentralized Management of Trust and Quality. In *Cooperative Information Agents XI*, pages 328–342. Springer, 2007.
- [VBGK09] J. Volz, C. Bizer, M. Gaedke, and G. Kobilarov. Discovering and Maintaining Links on the Web of Data. In *The Semantic Web-ISWC 2009*, pages 650–665. Springer, 2009.
- [VN12] R. Van Noorden. Chemistry’s Web of Data Expands. *Nature*, 483(7391):524–524, 2012.
- [VNM07] J. Von Neumann and O. Morgenstern. *Theory of Games and Economic Behavior (Commemorative Edition)*. Princeton university press, 2007.

- [Wai] The Waisda Project. <http://www.waisda.nl>. [accessed 14/12/2013].
- [Wan98] R. Wang. A Product Perspective on Total Data Quality Management. *Communications of the ACM*, 41(2):58–65, 1998.
- [Wan07] H. Wang. *Building Bayesian Networks: Elicitation, Evaluation, and Learning*. PhD thesis, University of Pittsburgh, 2007.
- [WCG⁺06] Y. Wang, V. Cahill, E. Gray, C. Harris, and L. Liao. Bayesian Network Based Trust Management. In *Autonomic and Trusted Computing*, pages 246–257. Springer, 2006.
- [WDI12] Gnu Wdiff. <http://www.gnu.org/software/wdiff/>, 2012. [accessed 20/11/2012].
- [WE11] A. J. Williams and S. Ekins. A Quality Alert and Call for Improved Curation of Public Chemistry Databases. *Drug Discovery today*, 16(17-18):747–750, July 2011.
- [WET12] A. J. Williams, S. Ekins, and V. Tkachenko. Towards A Gold Standard: Regarding Quality In Public Domain Chemistry Databases and Approaches to Improving The Situation. *Drug discovery today*, 17(13):685–701, 2012.
- [WHG⁺12] A. J. Williams, L. Harland, P. Groth, S. Pettifer, C. Chichester, E. L. Willighagen, C. T. Evelo, N. Blomberg, G. Ecker, C. Goble, and B. Mons. Open PHACTS: Semantic Interoperability for Drug Discovery. *Drug discovery today*, 00(00), June 2012.
- [Wid04] J. Widom. Trio: A System for Integrated Management of Data, Accuracy, and Lineage. *Technical Report*, 2004.
- [wik06] Wikipedia: Feature Articles (January 2006). http://en.wikipedia.org/w/index.php?title=Wikipedia:Featured_articles&oldid=37587127, January 2006. [accessed 02/02/2013].
- [Wil08] A. J. Williams. A Perspective of Publicly Accessible/Open-access Chemistry Databases. *Drug discovery today*, 13(11):495–501, 2008.

- [WKLW98] S. Weibel, J. Kunze, C. Lagoze, and M. Wolf. Dublin Core Metadata for Resource Discovery. *Internet Engineering Task force RFC*, 2413:222, 1998.
- [WMSE10] W.-C. Wong, S. Maurer-Stroh, and F. Eisenhaber. More Than 1,001 Problems With Protein Domain Databases: Transmembrane Regions, Signal Peptides and the Issue of Sequence Homology. *PLoS computational biology*, 6(7):e1000867, 2010.
- [WOH⁺11] K. Wolstencroft, S. Owen, M. Horridge, O. Krebs, W. Mueller, J. L. Snoep, F. du Preez, and C. Goble. RightField: Embedding Ontology Annotation in Spreadsheets. *Bioinformatics (Oxford, England)*, pages 1–2, May 2011.
- [WPC12] Wikiproject Chemicals Assessment. http://en.wikipedia.org/wiki/Wikipedia:WikiProject_Chemicals, 2012. [accessed 10/07/2012].
- [WS96] R. Y. Wang and D. M. Strong. Beyond Accuracy : What Data Quality Means to Data Consumers. *Journal of Management Information Systems*, 12(4):5–34, 1996.
- [Wu13] S. Wu. A Model for Assessing The Quality of Gene Ontology. pages 953–956. iSchools, 02 2013.
- [WV05] Y. Wang and J. Vassileva. Bayesian Network Trust Model In Peer-to-peer Networks. In *Agents and Peer-to-Peer Computing*, pages 23–34. Springer, 2005.
- [WYF03] H. Washizaki, H. Yamamoto, and Y. Fukazawa. A Metrics Suite for Measuring Reusability of Software Components. In *Software Metrics Symposium, 2003. Proceedings. Ninth International*, pages 211–223, 2003.
- [YC05] Y. Yang and J. Calmet. Ontobayes: An ontology-driven Uncertainty Model. In *Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on*, volume 1, pages 457–463. IEEE, 2005.

- [YGL11] X. Yang, Y. Guo, and Y. Liu. Bayesian-inference Based Recommendation In online Social Networks. In *INFOCOM, 2011 Proceedings IEEE*, pages 551–555. IEEE, 2011.
- [ZAD⁺06] H. Zeng, M. A. Alhossaini, L. Ding, R. Fikes, and D. L. McGuinness. Computing Trust from Revision History. Technical report, Stanford University Knowledge Systems Lab., 2006.
- [ZAFM06] H. Zeng, M. A. Alhossaini, R. Fikes, and D. L. McGuinness. Mining Revision History to Assess Trustworthiness of Article Fragments. In *Collaborative Computing: Networking, Applications and Worksharing, 2006. CollaborateCom 2006. International Conference on*, pages 1–10. IEEE, 2006.
- [ZDSM05] I. Zaihrayeu, P. P. Da Silva, and D. L. McGuinness. IWTrust: Improving User Trust In Answers From The Web. In *Trust Management*, pages 384–392. Springer, 2005.
- [Zim08] A. S. Zimmerman. New Knowledge From Old Data: The Role of Standards in the Sharing and Reuse of Ecological Data. *Science, Technology & Human Values*, 33(5):631–652, September 2008.
- [ZKGG13] J. Zhao, G. Klyne, M. Gamble, and C. Goble. A Checklist-based Approach for Quality Assessment of Scientific Information. In *Proceedings of the Third Linked Science Workshop co-located at the International Semantic Web Conference, Sydney, Australia*, 2013.
- [ZL04] C.-N. Ziegler and G. Lausen. Spreading Activation Models for Trust Propagation. In *e-Technology, e-Commerce and e-Service, 2004. EEE'04. 2004 IEEE International Conference on*, pages 83–97. IEEE, 2004.
- [ZMG08] T. Zesch, C. Müller, and I. Gurevych. Extracting Lexical Semantic Knowledge From Wikipedia and Wiktionary. In *Proceedings of the 6th International Conference on Language Resources and Evaluation*, Marrakech, Morocco, May 2008.

- [ZRM⁺12] A. Zaveri, A. Rula, a. Maurino, R. Pietrobon, J. Lehmann, and S. Auer. Quality Assessment Methodologies for Linked Open Data. In Press, December 2012.
- [ZWG⁺04] J. Zhao, C. Wroe, C. Goble, R. Stevens, D. Quan, and M. Greenwood. Using Semantic Web Technologies for Representing e-science Provenance. In *The Semantic Web-ISWC 2004*, pages 92–106. Springer, 2004.

Minimum Information Checklist Analysis

275

Checklist	NAME	DOMAIN according to bioharing.org	ORGANIZATION	Re- quire- ments	Re- quire- ment Sets	Re- quire- ment Levels	Use of Vo- cabularies/ Data Restrictions	Cardi- nality Restric- tions	Num- ber of Au- thors	Fine	Med	Coarse	Notes
GUID and LSID Applicability Statements	Globally Unique Identifier and Life Science Identifier Applicability Statement	identifier	TDWG members	-	-	-	-	-	-	-	-	-	Not Rel- e- vant
GIATE	Guidelines for Information About Therapy Experiments	cancer therapy	Antibody Society; GIATE working group	78	29	Y	Y	Y	?	51	21	6	
LAB	LABoratory	clinical laboratory raw data; pharmacogenomics	CDISC members	91	12	Y	Y	Y	?	84	3	4	
MIACA	Minimal Information About a Cellular Assay	cell biology; cell line; assay; phenotype	The Cell Based Assay Standards Consortium	138	16	Y	Y	Y	?	106	16	16	
MINSEQE	Minimal Information about a high throuput SEQencing Experiment	nucleotide sequencing;gene expression	FGED society; MINSEQE working group	6	1	N	N	N	?	0	0	5	
MIAPA	Minimal Information About a Phylogenetic Analysis	phylogenetics		-	-	-	-	-	28	-	-	-	DRAFT
MIATA	Minimal Information About T Cell Assays	T cell immune monitoring; T cell epitope recognition assay	MIATA consortium	5	1	N	N	N	10	0	0	5	
MIMPP	Minimal Information for Mouse Phenotyping Procedures	mouse phenotyping; protocol	Mouse Phenotype Database Integration consortium	-	-	-	-	-	-	-	-	-	DRAFT
MINEMO	Minimal Information for Neural ElectroMagnetic Ontologies	neuroscience;event related potentials;electro- encephalogram	NEMO consortium	65	10	Y	Y	N	10	65	0	0	
MIPFE	Minimal Information for Protein Functional Evaluation	biochemistry	MIPFE working group	83	28	N	N	N	3+	83	0	0	DRAFT
MIQAS	Minimal Information for QTLs and Association Studies	association study;genotyping;genotype	MIQAS working group	75	9	Y	Y	Y	?	64	6	5	
MIRIAM	Minimal Information Required In the Annotation of biochemical Models	model description;biochemical reaction networks; curation; kinetic simulation	BioModels community	13	4	N	Y	N	16	2	3	8	
MINIMESS	Minimal Metagenome Sequence Analysis Standard	metage- nomics;metatranscriptomics		7	1	N	Y	N	3+	0	0	3	
MIAME/Tox	Minimum Information about a array-based toxicogenomics experiment	toxicogenomics; DNA microarray; gene expression; chemical compound;toxicity	RSBI working group; grown into the; ISA community	88	16	N	Y	N	12	32	35	21	

Check- list	NAME	DOMAIN according to bioharing.org	ORGANIZATION	Re- quire- ments	Re- quire- ment Sets	Re- quire- ment Levels	Use of Vocabularies/ Data Restrictions	Cardi- nality Restric- tions	Num- ber of Au- thors	Fine	Med	Coarse	Notes
MI- ABE	Minimum Information About a Bioactive Entity	bioactive entities	HUPO- PSI; MIABE working group	45	7	Y	Y	N	26	6	7	19	
MIA- BiE	Minimum Information About a Biofilm Experiment	biofilm	-	-	-	-	-	-	-	-	-	-	DRAFT
MICEE	Minimum Information about a Cardiac Electrophysiology Experiment	electrophysiology	MICEE working group	64	5	N	Y	N	60	30	26	8	
MIGS- MIMS	Minimum Information about a Genome Sequence	genome	GSC consortium; MIGS/MIMS working group	33	3	Y	Y	N	63	26	5	2	
MI- Gen	Minimum Information About a Genotyping Experiment	genotyping assay; association study; genotype	MIGen working group	61	26	Y	Y	Y	13	30	16	15	
MI- MARKS	Minimum Information about a MARKer gene Sequence	biodiversity;microbial communities;marker gene	GSC consortium; MIMARKS working group	15	5	Y	y	N	98	13	2	0	
MI- AME	Minimum Information About a Microarray Experiment	DNA microarray	FGED society; MIAME working group	6	1	N	Y	N	24	0	0	6	
MI- AME/- Plant	Minimum Information About a Microarray Experiment involving Plants	plant; DNA microarray; gene expression	FGED society; MIAME plant working group	199	65	N	Y	Y	-	185	9	5	
MIMIX	Minimum Information about a Molecular Interaction Experiment	molecular interaction	HUPO- PSI; molecular interactions working group	14	5	N	Y	N	39	14	0	0	
MINI	Minimum Information about a Neuroscience Investigation	electrophysiol- ogy;neuroscience	CARMEN consortium	52	7	Y	N	N	17	23	26	3	
MI- AME/Nut	Minimum Information about a Nutrigenomics experiment	nutrition; DNA microarray; gene expression	RSBI working group; grown into the;ISA community	242	49	N	Y	N	-	192	32	10	
MIA- PepAE	Minimum Information About a Peptide Array Experiment	proteomics, microarray		64	17	Y	Y	N	2	42	15	7	
MIA- PAR	Minimum Information about a Protein Affinity Reagent	protein affinity reagent; antibody; antigen;affinity	HUPO- PSI; molecular interactions working group	53	8	N	n	N	34	53	0	0	
MIAPE- MS	Minimum Information About a Proteomics Experiment	proteomics;gel electrophoresis; mass spectrometry; chromatography	HUPO-PSI initiative	34	12	N	Y	Y	21	25	3	6	

Check- list	NAME	DOMAIN according to bioharing.org	ORGANIZATION	Re- quire- ments	Re- quire- ment Sets	Re- quire- ment Levels	Use of Vo- cabularies/ Data Restrictions	Cardi- nality Re- stric- tions	Num- ber of Au- thors	Fine	Med	Coarse	Notes
MI- ARE	Minimum Information About a RNAi Experiment	RNA interference; gene expression; siRNA; shRNA	MIARE informatics working group	63	12	Y	Y	N	61	37	9	17	
MI- ASE	Minimum Information About a Simulation Experiment	simula- tion; modeling; biochemistry; physiology	BioModels.net	10	3	N	n	N	28	0	0	10	
MI- EN- AME/EN	Minimum Information about an Environmental transcriptomic experiment	environmental condition; transcriptomics; DNA microarray; comparative genomics	EG working group; grown into the; ISA community	52	26	Y	Y	Y	-	37	12	13	
MfMRI	Minimum Information about an fMRI Study	functional magnetic resonance imaging	fMRI methods working group	81	17	N	Y	N	6	13	45	23	
MI- ABIS	Minimum Information About Biobank data Sharing	biobanks	BBMRI consortium	52	2	N	Y	Y	7	46	3	4	
MI- ASPPE	Minimum Information About Sample Preparation for a Phosphoproteomics Experiment	protein; phosphorylation; sample preparation	CSIC/UAB Proteomics Laboratory	129	34	N	Y	Y	?	41	60	28	
MI- Flow- Cyt	Minimum Information for a Flow Cytometry Experiment	flow cytometry	Flow Cytometry consortium	69	4	Y	Y	Y	33	29	10	30	
MIQE	Minimum Information for Publication of Quantitative Real-Time PCR Experiments	quantitative PCR	RDML consortium; MIQE working group	42	5	Y	Y	N	12	9	27	9	
MIDE	Minimum Information required for a DMET Experiment	pharmacogenomics	Bioinformatics unit, CPGR (Centre for Proteomic and Genomic Research)	-	-	-	-	-	-	-	-	-	To Be Pub- lished
MIS- FISHIE	Minimum Information Specification For In Situ Hybridization and Immunohistochemistry Experiments	gene expression; localization; immunohistochemistry; in-situ hybridization	FGED society; MISFISHIE working group	27	6	Y	Y	N	40	6	11	10	
ORION	Outbreak Reports and Intervention studies Of Nosocomial infection	infection control	ORION group	22	5	Y	Y	N	12	2	3	17	
PRISMA	Preferred Reporting Items for Systematic reviews and Meta-Analyses	evaluation of intervention study	PRISMA members	27	7	N	N	N	4+	0	0	27	
RE- FLECT	Reporting guidELines For randomized controlLed trials for livEstock and food safeTy	livestock trials	REFLECT members	22	1	N	N	N	20	0	0	22	
FO- RUM		toxicology		-	-	-	-	- - -	-	-	-		DRAFT

Appendix B

The MIM Vocabulary

```
1 @prefix :      <http://purl.org/net/mim/ns#> .
2 @prefix dc:    <http://purl.org/dc/elements/1.1/> .
3 @prefix owl: <http://www.w3.org/2002/07/owl#> .
4 @prefix owl2xml: <http://www.w3.org/2006/12/owl2-xml#> .
5 @prefix rdf:    <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
6 @prefix rdfs:   <http://www.w3.org/2000/01/rdf-schema#> .
7 @prefix xsd:    <http://www.w3.org/2001/XMLSchema#> .
8
9 dc:creator
10     rdf:type owl:AnnotationProperty .
11
12 <http://purl.org/net/mim/ns#
13     rdf:type owl:Ontology ;
14     dc:creator "Matthew Gamble" .
15
16 :Condition
17     rdf:type owl:Class ;
18     rdfs:subClassOf owl:Thing .
19
20 :DataReport
21     rdf:type owl:Class ;
22     rdfs:subClassOf :Report ;
23     owl:equivalentClass
24         [ rdf:type owl:Restriction ;
25           owl:onProperty :reports ;
26           owl:someValuesFrom :DataRequirement
27         ] .
28
29 :DataRequirement
30     rdf:type owl:Class ;
31     rdfs:subClassOf :Requirement .
32
33 :MIM rdf:type owl:Class ;
34     rdfs:comment "Model identifies the top level Minimum Information Model
35                 that is being defined."@en ;
36     rdfs:subClassOf :ReportSet ;
```

```

36     owl:equivalentClass
37         [ rdf:type owl:Restriction ;
38           owl:onProperty :hasRequirement ;
39           owl:someValuesFrom :Requirement
40         ] .
42
42 :ObjectReport
43     rdf:type owl:Class ;
44     rdfs:subClassOf :Report ;
45     owl:equivalentClass
46         [ rdf:type owl:Restriction ;
47           owl:onProperty :reports ;
48           owl:someValuesFrom :ObjectRequirement
49         ] .
51
51 :ObjectRequirement
52     rdf:type owl:Class ;
53     rdfs:subClassOf :Requirement .
55
55 :Report
56     rdf:type owl:Class ;
57     rdfs:comment "A Report identifies the instance of an attempt to report a
58         requirement in a minimum information model. A report must indicate the
59         requirement that it reports." ;
60     owl:equivalentClass
61         [] .
62
62 :ReportSet
63     rdf:type owl:Class ;
64     owl:equivalentClass
65         [ rdf:type owl:restriction ;
66           owl:hasSelf "true"^^xsd:boolean ;
67           owl:onProperty :contains
68         ] .
69
69 :Requirement
70     rdf:type owl:Class ;
71     rdfs:comment "A requirement is an atomic reporting requirement for a
72         minimum information model."@en .
73
73 :RequirementSet
74     rdf:type owl:Class .
76
76 :Restriction
77     rdf:type owl:Class ;
78     rdfs:subClassOf owl:Thing .
80
80 :Rule
81     rdf:type owl:Class .
83
83 :additionallySatisfies
84     rdf:type owl:ObjectProperty ;
85     rdfs:subPropertyOf :satisfies .
87

```



```

87 :adequatelySatisfies
88     rdf:type owl:ObjectProperty ;
89     rdfs:subPropertyOf :satisfies .
91
91 :cardinality
92     rdf:type owl:DatatypeProperty .
94
94 :contains
95     rdf:type owl:ObjectProperty ;
96     rdfs:subPropertyOf owl:topObjectProperty .
98
98 :containsDataReport
99     rdf:type owl:ObjectProperty ;
100    rdfs:domain :Report ;
101    rdfs:range :DataReport ;
102    rdfs:subPropertyOf :containsReport .
104
104 :containsReport
105     rdf:type owl:ObjectProperty ;
106     rdfs:domain :ReportSet ;
107     rdfs:range :Report ;
108     rdfs:subPropertyOf :contains .
110
110 :containsReportSet
111     rdf:type owl:ObjectProperty ;
112     rdfs:domain :ReportSet ;
113     rdfs:range :ReportSet ;
114     rdfs:subPropertyOf :contains .
116
116 :directInstanceOf
117     rdf:type owl:ObjectProperty ;
118     rdfs:domain :Restriction ;
119     rdfs:range rdfs:Class ;
120     rdfs:subPropertyOf owl:topObjectProperty .
122
122 :directSubclassOf
123     rdf:type owl:ObjectProperty ;
124     rdfs:domain :Restriction ;
125     rdfs:range rdfs:Class ;
126     rdfs:subPropertyOf owl:topObjectProperty .
128
128 :exactCardinality
129     rdf:type owl:DatatypeProperty ;
130     rdfs:domain :Restriction ;
131     rdfs:range xsd:positiveInteger ;
132     rdfs:subPropertyOf :cardinality .
134
134 :hasMustRequirement
135     rdf:type owl:ObjectProperty ;
136     rdfs:subPropertyOf :hasRequirement .
138
138 :hasOptionalRequirement
139     rdf:type owl:ObjectProperty ;
140     rdfs:subPropertyOf :hasRequirement .

```

```

142
142 :hasRequirement
143     rdf:type owl:ObjectProperty ;
144     rdfs:domain :RequirementSet ;
145     rdfs:range :Requirement , :RequirementSet ;
146     rdfs:subPropertyOf :contains .
148
148 :hasRestriction
149     rdf:type owl:ObjectProperty ;
150     rdfs:domain :RequirementSet , :Requirement ;
151     rdfs:range :Restriction ;
152     rdfs:subPropertyOf owl:topObjectProperty .
154
154 :hasShouldRequirement
155     rdf:type owl:ObjectProperty ;
156     rdfs:subPropertyOf :hasRequirement .
158
158 :identifies
159     rdf:type owl:ObjectProperty ;
160     rdfs:domain :Rule ;
161     rdfs:range :Requirement ;
162     rdfs:subPropertyOf owl:topObjectProperty ;
163     owl:inverseOf :isIdentifiedBy .
165
165 :instanceOf
166     rdf:type owl:ObjectProperty ;
167     rdfs:domain :Restriction ;
168     rdfs:range rdfs:Class ;
169     rdfs:subPropertyOf owl:topObjectProperty .
171
171 :isConditionalUpon
172     rdf:type owl:ObjectProperty ;
173     rdfs:domain :Restriction ;
174     rdfs:range :Condition ;
175     rdfs:subPropertyOf owl:topObjectProperty .
177
177 :isIdentifiedBy
178     rdf:type owl:ObjectProperty ;
179     rdfs:domain :Requirement ;
180     rdfs:range :Rule ;
181     rdfs:subPropertyOf owl:topObjectProperty .
183
183 :isReportedBy
184     rdf:type owl:ObjectProperty ;
185     rdfs:subPropertyOf owl:topObjectProperty ;
186     owl:inverseOf :reports .
188
188 :isSatisfiedBy
189     rdf:type owl:ObjectProperty ;
190     rdfs:subPropertyOf owl:topObjectProperty .
192
192 :maxCardinality
193     rdf:type owl:DatatypeProperty ;
194     rdfs:domain :Restriction ;

```

```

195         rdfs:range xsd:positiveInteger ;
196         rdfs:subPropertyOf :cardinality .
197
198 :maximallySatisfies
199         rdf:type owl:ObjectProperty ;
200         rdfs:subPropertyOf :satisfies .
201
202 :minCardinality
203         rdf:type owl:DatatypeProperty ;
204         rdfs:domain :Restriction ;
205         rdfs:range xsd:positiveInteger ;
206         rdfs:subPropertyOf :cardinality .
207
208 :minimallySatisfies
209         rdf:type owl:ObjectProperty ;
210         rdfs:subPropertyOf :satisfies .
211
212 :onRequirement
213         rdf:type owl:ObjectProperty ;
214         rdfs:domain :Restriction ;
215         rdfs:range :Requirement , :RequirementSet ;
216         rdfs:subPropertyOf owl:topObjectProperty .
217
218 :onSelf
219         rdf:type owl:DatatypeProperty ;
220         rdfs:domain :Restriction ;
221         rdfs:range xsd:boolean .
222
223 :partOf
224         rdf:type owl:ObjectProperty ;
225         rdfs:domain :Requirement ;
226         rdfs:range :RequirementSet ;
227         rdfs:subPropertyOf owl:topObjectProperty .
228
229 :reports
230         rdf:type owl:ObjectProperty ;
231         rdfs:domain :ReportSet , :Report ;
232         rdfs:range :RequirementSet , :Requirement ;
233         rdfs:subPropertyOf owl:topObjectProperty .
234
235 :satisfies
236         rdf:type owl:ObjectProperty ;
237         rdfs:subPropertyOf owl:topObjectProperty .
238
239 :subclassOf
240         rdf:type owl:ObjectProperty ;
241         rdfs:domain :Restriction ;
242         rdfs:range rdfs:Class ;
243         rdfs:subPropertyOf owl:topObjectProperty .
244
245 :type
246         rdf:type owl:ObjectProperty ;
247         rdfs:domain :Restriction ;
248         rdfs:range rdfs:Datatype ;

```

```
249     rdfs:subPropertyOf owl:topObjectProperty .
251
251 :withValue
252     rdf:type owl:DatatypeProperty ;
253     rdfs:domain :DataReport .
255
255 owl:Thing
256     rdf:type owl:Class .
258
258 owl:topObjectProperty
259     rdf:type owl:ObjectProperty .
```

Appendix C

The chemmim Checklist

```
1 @prefix :      <http://sierra-nevada.cs.man.ac.uk/mim/chembox/chembox-mim#> .
2 @prefix dc:    <http://purl.org/dc/elements/1.1/> .
3 @prefix dcterms: <http://purl.org/dc/terms/> .
4 @prefix foaf:  <http://xmlns.com/foaf/0.1/> .
5 @prefix mim:   <http://purl.org/net/mim/ns#> .
6 @prefix owl: <http://www.w3.org/2002/07/owl#> .
7 @prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
8 @prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#> .
9 @prefix xsd:   <http://www.w3.org/2001/XMLSchema#> .
11
11 <http://sierra-nevada.cs.man.ac.uk/mim/chembox/chemmim>
12   rdf:type owl:Ontology ;
13   owl:imports <http://purl.org/net/mim/ns#> .
15
15 :MIC rdf:type mim:MIC ;
16     mim:hasMustRequirement
17       :Properties , :Identifiers ;
18     mim:hasOptionalRequirement
19       :Synonym ;
20     mim:hasRestriction
21       [ mim:exactCardinality
22         1 ;
23         mim:onRequirement :Properties , :Identifiers
24       ] ;
25     mim:hasShouldRequirement
26       :IUPACName , :Image .
28
28 :MeltingPoint
29   rdf:type mim:RequirementSet ;
30   mim:hasMustRequirement
31     :MeltingPointUnits , :MeltingPointValue ;
32   mim:hasRestriction
33     [ mim:exactCardinality
34       1 ;
35       mim:onRequirement :MeltingPointUnits , :MeltingPointValue
36     ] .
38
```

```
38 :ChemSpider
39     rdf:type mim:DataRequirement ;
40     mim:hasRestriction
41         [ mim:onSelf "true"^^xsd:boolean ;
42           mim:type xsd:integer
43         ] .
44
45 :Identifiers
46     rdf:type mim:RequirementSet ;
47     mim:hasMustRequirement
48         :InChI , :SMILES ;
49     mim:hasShouldRequirement
50         :ChemSpider , :PubChem .
51
52 :InChI
53     rdf:type mim:DataRequirement ;
54     mim:hasRestriction
55         [ mim:onSelf "true"^^xsd:boolean ;
56           mim:type xsd:string
57         ] .
58
59 :MolarMass
60     rdf:type mim:DataRequirement .
61
62 :MolecularFormula
63     rdf:type mim:DataRequirement ;
64     mim:hasRestriction
65         [ mim:onSelf "true"^^xsd:boolean ;
66           mim:type xsd:string
67         ] .
68
69 :Properties
70     rdf:type mim:RequirementSet ;
71     mim:hasMustRequirement
72         :MolecularFormula ;
73     mim:hasOptionalRequirement
74         :Solubility ;
75     mim:hasRestriction
76         [ mim:exactCardinality
77           1 ;
78           mim:onRequirement :MolecularFormula
79         ] ;
80     mim:hasShouldRequirement
81         :MeltingPoint , :MolarMass .
82
83 :PubChem
84     rdf:type mim:DataRequirement ;
85     mim:hasRestriction
86         [ mim:onSelf "true"^^xsd:boolean ;
87           mim:type xsd:integer
88         ] .
89
90 :SMILES
91     rdf:type mim:DataRequirement ;
```

```
92         mim:hasRestriction
93             [ mim:onSelf "true"^^xsd:boolean ;
94               mim:type xsd:string
95             ] .
96
97 :Solubility
98     rdf:type mim:DataRequirement .
99
100 :Synonym
101     rdf:type mim:DataRequirement .
102
103 :Image
104     rdf:type mim:DataRequirement .
105
106 :IUPACName
107     rdf:type mim:DataRequirement .
108
109 :MeltingPointUnits
110     rdf:type mim:DataRequirement .
111
112 :MeltingPointValue
113     rdf:type mim:DataRequirement .
```

Appendix D

The mimspin MIM Validation Rules

```
1 # baseURI: http://purl.org/net/mim/mimspin
2 # imports: http://purl.org/net/mim/ns
3 # imports: http://spinrdf.org/spin
4 # imports: http://spinrdf.org/spl
5
6
7 @prefix :      <http://purl.org/net/mim/mimspin#> .
8 @prefix fn:    <http://www.w3.org/2005/xpath-functions#> .
9 @prefix foaf:  <http://xmlns.com/foaf/0.1/> .
10 @prefix mim:   <http://purl.org/net/mim/ns#> .
11 @prefix mimspin: <http://purl.org/net/mim/mimspin#> .
12 @prefix owl: <http://www.w3.org/2002/07/owl#> .
13 @prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
14 @prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#> .
15 @prefix sp:    <http://spinrdf.org/sp#> .
16 @prefix spin:  <http://spinrdf.org/spin#> .
17 @prefix spl:   <http://spinrdf.org/spl#> .
18 @prefix xsd:   <http://www.w3.org/2001/XMLSchema#> .
19
20 <http://purl.org/net/mim/mimspin>
21   rdf:type owl:Ontology ;
22   owl:imports <http://spinrdf.org/spin> , <http://purl.org/net/mim/ns> , <
23     http://spinrdf.org/spl> ;
24   owl:versionInfo "Created with TopBraid Composer"^^xsd:string .
25
26 mimspin:ChecklistSatisfaction
27   rdf:type spin:ConstructTemplate ;
28   rdfs:subClassOf mimspin:ConstructTemplates ;
29   spin:abstract "true"^^xsd:boolean .
30
31 mimspin:ConstructTemplates
32   rdf:type spin:ConstructTemplate ;
33   rdfs:subClassOf spin:ConstructTemplates ;
34   spin:abstract "true"^^xsd:boolean .
```



```

34 mимspin:Contains
35     rdf:type spin:Template ;
36     rdfs:subClassOf mимspin:ConstructTemplates ;
37     spin:body
38         [ rdf:type sp:Construct ;
39           sp:text """CONSTRUCT {
40             ?x mим:contains ?y .
41         }
42     WHERE {
43         ?x (mим:containsReport|mим:containsReportSet)|mим:containsDataReport ?y .
44     }"""^^xsd:string
45         ] ;
46     spin:labelTemplate "MIMSPIN: Contains inference"^^xsd:string .
47
48 mимspin:ContainsReport
49     rdf:type spin:Template ;
50     rdfs:subClassOf mимspin:ConstructTemplates ;
51     spin:body
52         [ rdf:type sp:Construct ;
53           sp:text """CONSTRUCT {
54             ?x mим:containsReport ?y .
55         }
56     WHERE {
57         ?x mим:containsDataReport ?y .
58     }"""^^xsd:string
59         ] ;
60     spin:labelTemplate "MIMSPIN: Contains Report inference"^^xsd:string .
61
62 mимspin:DataReport
63     rdf:type spin:Template ;
64     rdfs:subClassOf mимspin:ConstructTemplates ;
65     spin:body
66         [ rdf:type sp:Construct ;
67           sp:text """CONSTRUCT {
68             ?x a mим:DataReport .
69         }
70     WHERE {
71         ?x mим:reports ?y .
72         ?x mим:withValue ?value .
73     }"""^^xsd:string
74         ] ;
75     spin:labelTemplate "MIMSPIN: DataReport inference"^^xsd:string .
76
77 mимspin:DataRequirementSatisfaction
78     rdf:type spin:ConstructTemplate ;
79     rdfs:comment "Requirement Satisfaction"^^xsd:string ;
80     rdfs:subClassOf mимspin:RequirementSatisfaction ;
81     spin:body
82         [ rdf:type sp:Construct ;
83           sp:text """# DataReport satisfies DataRequirement
84     CONSTRUCT {
85         ?z mим:satisfies ?y .
86     }

```

```

87 WHERE {
88     ?x a mim:ReportSet .
89     ?y a mim:DataRequirement .
90     ?x mim:containsDataReport ?z .
91     ?z mim:reports ?y .
92     ?z mim:withValue ?v .
93     BIND (mimspin:violatesDatatypeRestriction(?y, ?v) AS ?result) .
94     FILTER (!?result) .
95 }""^^xsd:string
96     ] ;
97     spin:labelTemplate "MiMSPIN : DataRequirementSatisfaction"^^xsd:string .
98
99 mimspin:Functions
100     rdf:type spin:Function ;
101     rdfs:subClassOf spin:Functions ;
102     spin:abstract "true"^^xsd:boolean .
103
104 mimspin:ObjectRequirementSatisfaction
105     rdf:type spin:ConstructTemplate ;
106     rdfs:comment "Requirement Satisfaction"^^xsd:string ;
107     rdfs:subClassOf mimspin:RequirementSatisfaction ;
108     spin:body
109         [ rdf:type sp:Construct ;
110           sp:text ""# ObjectReport satisfies ObjectRequirement : FIX type
111             check!
112         ]
113     ]
114 WHERE {
115     ?x a mim:Report .
116     ?y a mim:ObjectRequirement .
117     ?x mim:reports ?y .
118     FILTER (!mimspin:violatesObjectInstanceOfRestriction(?y, ?x)) .
119 }""^^xsd:string
120     ] ;
121     spin:labelTemplate "MiMSPIN : ObjectRequirementSatisfaction"^^xsd:string .
122
123 mimspin:Report
124     rdf:type spin:Template ;
125     rdfs:subClassOf mimspin:ConstructTemplates ;
126     spin:body
127         [ rdf:type sp:Construct ;
128           sp:text ""CONSTRUCT {
129             ?x a mim:Report .
130           }
131         ]
132     ]
133 WHERE {
134     ?x mim:reports ?y .
135 }""^^xsd:string
136     ] ;
137     spin:labelTemplate "MiMSPIN: Report inference"^^xsd:string .
138
139 mimspin:ReportSet
140     rdf:type spin:Template ;
141     rdfs:subClassOf mimspin:ConstructTemplates ;

```

```

140         spin:body
141         [ rdf:type sp:Construct ;
142           sp:text """CONSTRUCT {
143             ?x a mim:ReportSet .
144           }
145         WHERE {
146             ?x mim:contains ?y .
147         }"""^^xsd:string
148         ] ;
149         spin:labelTemplate "MIMSPIN: ReportSet inference"^^xsd:string .
151
151     mimspin:ReportSetContainsSelf
152         rdf:type spin:Template ;
153         rdfs:subClassOf mimspin:ConstructTemplates ;
154         spin:body
155         [ rdf:type sp:Construct ;
156           sp:text """CONSTRUCT {
157             ?this mim:contains ?this .
158           }
159         WHERE {
160             ?this a mim:ReportSet .
161         }"""^^xsd:string
162         ] ;
163         spin:labelTemplate "MIMSPIN: ReportSet-contains-Self"^^xsd:string .
165
165     mimspin:RequirementSatisfaction
166         rdf:type spin:ConstructTemplate ;
167         rdfs:subClassOf mimspin:ChecklistSatisfaction .
169
169     mimspin:RequirementSet
170         rdf:type spin:Template ;
171         rdfs:subClassOf mimspin:ConstructTemplates ;
172         spin:body
173         [ rdf:type sp:Construct ;
174           sp:text """CONSTRUCT {
175             ?x a mim:RequirementSet .
176           }
177         WHERE {
178             ?x a mim:MIC .
179         }"""^^xsd:string
180         ] ;
181         spin:labelTemplate "MIMSPIN: RequirementSet inference"^^xsd:string .
183
183     mimspin:RequirementSetSatisfaction
184         rdf:type spin:ConstructTemplate ;
185         rdfs:subClassOf mimspin:ChecklistSatisfaction ;
186         spin:abstract "true"^^xsd:boolean .
188
188     mimspin:Satisfies
189         rdf:type spin:Template ;
190         rdfs:subClassOf mimspin:ConstructTemplates ;
191         spin:body
192         [ rdf:type sp:Construct ;
193           sp:text """CONSTRUCT {

```

```

194     ?x mim:satisfies ?y .
195 }
196 WHERE {
197     ?x ((mim:minimallySatisfies|mim:additionallySatisfies)|mim:
198         adequatelySatisfies)|mim:maximallySatisfies ?y .
199 }""^^xsd:string
200     ] ;
201     spin:labelTemplate "MIMSPIN: Contains inference"^^xsd:string .
202
203 mimspin:additionallySatisfies
204     rdf:type spin:ConstructTemplate ;
205     rdfs:comment "Additionally Satisfies"^^xsd:string ;
206     rdfs:subClassOf mimspin:RequirementSetSatisfaction ;
207     spin:body
208         [ rdf:type sp:Construct ;
209           sp:text ""CONSTRUCT {
210             ?arg1 mim:additionallySatisfies ?arg2 .
211           }
212         ]
213 WHERE {
214     ?arg1 a mim:ReportSet .
215     ?arg2 a mim:RequirementSet .
216     ?arg1 mim:minimallySatisfies ?arg2 .
217     OPTIONAL {
218         ?arg2 mim:hasOptionalRequirement ?req .
219         ?arg1 mim:satisfies ?req .
220     } .
221     OPTIONAL {
222         ?arg2 mim:hasShouldRequirement ?req .
223         ?arg1 mim:satisfies ?req .
224     } .
225     FILTER bound(?req) .
226 }""^^xsd:string
227     ] ;
228     spin:labelTemplate "MIMSPIN : AdditionallySatisfies"^^xsd:string .
229
230 mimspin:adequatelySatisfies
231     rdf:type spin:ConstructTemplate ;
232     rdfs:comment "Adequately Satisfies"^^xsd:string ;
233     rdfs:subClassOf mimspin:RequirementSetSatisfaction ;
234     spin:body
235         [ rdf:type sp:Construct ;
236           sp:text ""CONSTRUCT {
237             ?arg1 mim:adequatelySatisfies ?arg2 .
238             ?x mim:should ?should .
239             ?x mim:adequate ?adequate .
240           }
241         ]
242 WHERE {
243     ?arg1 a mim:ReportSet .
244     ?arg2 a mim:RequirementSet .
245     ?arg1 mim:minimallySatisfies ?arg2 .
246     BIND ((!mimspin:hasUnsatisfiedShouldRequirement(?arg2, ?arg1)) AS ?should) .
247     FILTER (?should) .
248     BIND ((!mimspin:missingAdequatelySatisfyingChild(?arg2, ?arg1, mim:
249         adequatelySatisfies)) AS ?adequate) .

```

```

246     FILTER (?adequate) .
247 }""^^xsd:string
248     ] .
249
250 mимspin:hasUnsatisfied
251     rdf:type spin:Function ;
252     rdfs:subClassOf mимspin:Functions ;
253     spin:abstract "true"^^xsd:boolean .
254
255 mимspin:hasUnsatisfiedMustRequirement
256     rdf:type spin:Function ;
257     rdfs:subClassOf mимspin:hasUnsatisfied ;
258     spin:body
259         [ rdf:type sp:Ask ;
260           sp:text ""# is there an immediate must requirement in ReqSet
261             that is violated the by RepSet ?
262
263 ASK WHERE {
264     ?arg1 mим:hasMustRequirement ?Req .
265     BIND (mимspin:reportCount(?arg2, ?Req) AS ?rc) .
266     BIND (mимspin:satisfyingReportCount(?arg2, ?Req) AS ?sc) .
267     BIND (mимspin:violatesMaxCardinality(?Req, ?arg1, ?rc) AS ?max) .
268     BIND (mимspin:violatesMinCardinality(?Req, ?arg1, ?rc) AS ?min) .
269     BIND (mимspin:violatesExactCardinality(?Req, ?arg1, ?rc) AS ?exact) .
270     BIND (mимspin:violatesMaxCardinality(?Req, ?arg1, ?sc) AS ?smax) .
271     BIND (mимspin:violatesMinCardinality(?Req, ?arg1, ?sc) AS ?smin) .
272     BIND (mимspin:violatesExactCardinality(?Req, ?arg1, ?sc) AS ?sexact) .
273     BIND (mимspin:violatesDefaultCardinality(?Req, ?arg1, ?sc) AS ?default) .
274     FILTER (((((?max || ?min) || ?exact) || ?smax) || ?smin) || ?sexact) || ?
275         default) .
276 }""^^xsd:string
277     ] ;
278     spin:constraint
279         [ rdf:type spl:Argument ;
280           rdfs:comment "The ReportSet to be tested."^^xsd:string ;
281           spl:predicate sp:arg1
282         ] ;
283     spin:constraint
284         [ rdf:type spl:Argument ;
285           rdfs:comment "The MIM to be tested."^^xsd:string ;
286           spl:predicate sp:arg2
287         ] ;
288     spin:returnType xsd:integer .
289
290 mимspin:hasUnsatisfiedMustRequirement_1
291     rdf:type spin:Function ;
292     rdfs:subClassOf mимspin:hasUnsatisfied ;
293     spin:body
294         [ rdf:type sp:Ask ;
295           sp:text ""# is there an immediate must requirement in ReqSet
296             that is violated the by RepSet ?
297
298 ASK WHERE {
299     ?arg1 mим:hasMustRequirement ?Req .
300     BIND (:reportCount(?arg2, ?Req) AS ?rc) .
301     BIND (mимspin:satisfyingReportCount(?arg2, ?Req) AS ?sc) .

```

```

297 BIND (mimspin:violatesMaxCardinality(?Req, ?arg1, ?rc) AS ?max) .
298 BIND (mimspin:violatesMinCardinality(?Req, ?arg1, ?rc) AS ?min) .
299 BIND (mimspin:violatesExactCardinality(?Req, ?arg1, ?rc) AS ?exact) .
300 BIND (mimspin:violatesMaxCardinality(?Req, ?arg1, ?sc) AS ?smax) .
301 BIND (mimspin:violatesMinCardinality(?Req, ?arg1, ?sc) AS ?smin) .
302 BIND (mimspin:violatesExactCardinality(?Req, ?arg1, ?sc) AS ?sexact) .
303 BIND (mimspin:violatesDefaultCardinality(?Req, ?arg1, ?sc) AS ?default) .
304 FILTER ( (((((?max || ?min) || ?exact) || ?smax) || ?smin) || ?sexact) || ?
    default) .
305 }""^^xsd:string
306 ] ;
307 spin:constraint
308 [ rdf:type spl:Argument ;
309   rdfs:comment "The MIM to be tested."^^xsd:string ;
310   spl:predicate sp:arg2
311 ] ;
312 spin:constraint
313 [ rdf:type spl:Argument ;
314   rdfs:comment "The ReportSet to be tested."^^xsd:string ;
315   spl:predicate sp:arg1
316 ] ;
317 spin:returnType xsd:integer .
319
319 mimspin:hasUnsatisfiedOptionalRequirement
320   rdf:type spin:Function ;
321   rdfs:subClassOf mimspin:hasUnsatisfied ;
322   spin:body
323     [ rdf:type sp:Ask ;
324       sp:text ""ASK
325 WHERE
326   { ?arg1 mim:hasOptionalRequirement ?Req
327     BIND(:reportCount(?arg2, ?Req) AS ?rc)
328     BIND(:satisfyingReportCount(?arg2, ?Req) AS ?sc)
329     BIND(:violatesMaxCardinality(?Req, ?arg1, ?rc) AS ?max)
330     BIND(:violatesMinCardinality(?Req, ?arg1, ?rc) AS ?min)
331     BIND(:violatesExactCardinality(?Req, ?arg1, ?rc) AS ?exact)
332     BIND(:violatesMaxCardinality(?Req, ?arg1, ?sc) AS ?smax)
333     BIND(:violatesMinCardinality(?Req, ?arg1, ?sc) AS ?smin)
334     BIND(:violatesExactCardinality(?Req, ?arg1, ?sc) AS ?sexact)
335     BIND(:violatesDefaultCardinality(?Req, ?arg1, ?sc) AS ?default)
336     FILTER ( ( ( ( ( ?max || ?min ) || ?exact ) || ?smax ) || ?smin ) || ?
        sexact ) || ?default )
337 }""^^xsd:string
338 ] ;
339 spin:constraint
340 [ rdf:type spl:Argument ;
341   rdfs:comment "The MIM to be tested."^^xsd:string ;
342   spl:predicate sp:arg2
343 ] ;
344 spin:constraint
345 [ rdf:type spl:Argument ;
346   rdfs:comment "The ReportSet to be tested."^^xsd:string ;
347   spl:predicate sp:arg1
348 ] ;

```

```

349         spin:returnType xsd:integer .
351
352 mimsSpin:hasUnsatisfiedRequirement
353     rdf:type spin:Function ;
354     rdfs:subClassOf mimsSpin:hasUnsatisfied ;
355     spin:body
356         [ rdf:type sp:Ask ;
357           sp:text ""ASK
358 WHERE
359     { ?arg1 mim:contains ?Req
360       BIND(:reportCount(?arg2, ?Req) AS ?rc)
361       BIND(:satisfyingReportCount(?arg2, ?Req) AS ?sc)
362       BIND(:violatesMaxCardinality(?Req, ?arg1, ?rc) AS ?max)
363       BIND(:violatesMinCardinality(?Req, ?arg1, ?rc) AS ?min)
364       BIND(:violatesExactCardinality(?Req, ?arg1, ?rc) AS ?exact)
365       BIND(:violatesMaxCardinality(?Req, ?arg1, ?sc) AS ?smax)
366       BIND(:violatesMinCardinality(?Req, ?arg1, ?sc) AS ?smin)
367       BIND(:violatesExactCardinality(?Req, ?arg1, ?sc) AS ?sexact)
368       BIND(:violatesDefaultCardinality(?Req, ?arg1, ?sc) AS ?default)
369       FILTER ( ( ( ( ( ?max || ?min ) || ?exact ) || ?smax ) || ?smin ) || ?
370               sexact ) || ?default )
371     }""^^xsd:string
372     ] ;
373     spin:constraint
374         [ rdf:type spl:Argument ;
375           rdfs:comment "The ReportSet to be tested."^^xsd:string ;
376           spl:predicate sp:arg1
377         ] ;
378     spin:constraint
379         [ rdf:type spl:Argument ;
380           rdfs:comment "The MIM to be tested."^^xsd:string ;
381           spl:predicate sp:arg2
382         ] ;
383     spin:returnType xsd:integer .
384
385 mimsSpin:hasUnsatisfiedShouldRequirement
386     rdf:type spin:Function ;
387     rdfs:subClassOf mimsSpin:hasUnsatisfied ;
388     spin:body
389         [ rdf:type sp:Ask ;
390           sp:text ""
391 ASK
392 WHERE
393     { ?arg1 mim:hasShouldRequirement ?Req
394       BIND(:reportCount(?arg2, ?Req) AS ?rc)
395       BIND(:satisfyingReportCount(?arg2, ?Req) AS ?sc)
396       BIND(:violatesMaxCardinality(?Req, ?arg1, ?rc) AS ?max)
397       BIND(:violatesMinCardinality(?Req, ?arg1, ?rc) AS ?min)
398       BIND(:violatesExactCardinality(?Req, ?arg1, ?rc) AS ?exact)
399       BIND(:violatesMaxCardinality(?Req, ?arg1, ?sc) AS ?smax)
400       BIND(:violatesMinCardinality(?Req, ?arg1, ?sc) AS ?smin)
401       BIND(:violatesExactCardinality(?Req, ?arg1, ?sc) AS ?sexact)
402       BIND(:violatesDefaultCardinality(?Req, ?arg1, ?sc) AS ?default)

```

```

401     FILTER ( ( ( ( ( ?max || ?min ) || ?exact ) || ?smax ) || ?smin ) || ?
          sexact ) || ?default )
402   }
403   ""^^xsd:string
404   ] ;
405   spin:constraint
406     [ rdf:type spl:Argument ;
407       rdfs:comment "The MIM to be tested."^^xsd:string ;
408       spl:predicate sp:arg2
409     ] ;
410   spin:constraint
411     [ rdf:type spl:Argument ;
412       rdfs:comment "The ReportSet to be tested."^^xsd:string ;
413       spl:predicate sp:arg1
414     ] ;
415   spin:returnType xsd:integer .
417
417 mimspin:maximallySastisfies
418   rdf:type spin:ConstructTemplate ;
419   rdfs:comment "Maximally Satisfies"^^xsd:string ;
420   rdfs:subClassOf mimspin:RequirementSetSatisfaction ;
421   spin:body
422     [ rdf:type sp:Construct ;
423       sp:text ""CONSTRUCT {
424         ?x mim:maximallySatisfies ?y .
425         ?x mim:must ?must .
426         ?x mim:should ?should .
427         ?x mim:optional ?optional .
428       }
429     WHERE {
430       ?x a mim:ReportSet .
431       ?y a mim:RequirementSet .
432       ?x mim:reports ?y .
433       BIND (!mimspin:hasUnsatisfiedMustRequirement(?y, ?x)) AS ?must) .
434       BIND (!mimspin:hasUnsatisfiedShouldRequirement(?y, ?x)) AS ?should) .
435       BIND (!mimspin:hasUnsatisfiedOptionalRequirement(?y, ?x)) AS ?optional) .
436       FILTER ((?must && ?should) && ?optional) .
437       BIND (!mimspin:missingAdequatelySatisfyingChild(?y, ?x, mim:
         maximallySatisfies)) AS ?adequate) .
438       FILTER (?adequate) .
439     }""^^xsd:string
440     ] .
442
442 mimspin:minimallySastisfies
443   rdf:type spin:ConstructTemplate ;
444   rdfs:comment "Minimally Satisfies"^^xsd:string ;
445   rdfs:subClassOf mimspin:RequirementSetSatisfaction ;
446   spin:body
447     [ rdf:type sp:Construct ;
448       sp:text ""CONSTRUCT {
449         ?reportSet mim:minimallySatisfies ?requirementSet .
450       }
451     WHERE {
452       ?reportSet a mim:ReportSet .

```



```

453     ?requirementSet a mim:RequirementSet .
454     ?reportSet mim:reports ?requirementSet .
455     BIND ((!mimspin:hasUnsatisfiedMustRequirement(?requirementSet , ?reportSet))
           AS ?result) .
456     FILTER (?result) .
457 }""^^xsd:string
458     ] .
460
460 mimspin:missingAdequatelySatisfyingChild
461     rdf:type spin:Function ;
462     rdfs:subClassOf mimspin:Functions ;
463     spin:body
464         [ rdf:type sp:Ask ;
465           sp:text """"# is there an immediate requirement in ReqSet that
                     isnt {?arg3} satisfied the by RepSet ?
466 ASK WHERE {
467     ?arg1 a mim:RequirementSet .
468     ?arg2 a mim:ReportSet .
469     ?arg2 mim:reports ?arg1 .
470     ?arg2 mim:contains ?SubRepSet .
471     ?SubRepSet a mim:ReportSet .
472     ?arg1 ((mim:hasMustRequirement|mim:hasShouldRequirement)|mim:
             hasOptionalRequirement)|mim:hasRequirement ?SubReqSet .
473     ?SubReqSet a mim:RequirementSet .
474     ?SubRepSet mim:reports ?SubReqSet .
475     FILTER (?SubReqSet != ?arg1) .
476     MINUS {
477         ?SubRepSet ?arg3 ?SubReqSet .
478     } .
479 }""^^xsd:string
480     ] ;
481     spin:constraint
482         [ rdf:type spl:Argument ;
483           spl:predicate sp:arg3
484         ] ;
485     spin:constraint
486         [ rdf:type spl:Argument ;
487           rdfs:comment "The ReportSet to be tested."^^xsd:string ;
488           spl:predicate sp:arg1
489         ] ;
490     spin:constraint
491         [ rdf:type spl:Argument ;
492           rdfs:comment "The MIM to be tested."^^xsd:string ;
493           spl:predicate sp:arg2
494         ] ;
495     spin:returnType xsd:integer .
497
497 mimspin:reportCount
498     rdf:type spin:Function ;
499     rdfs:subClassOf mimspin:Functions ;
500     spin:body
501         [ rdf:type sp:Select ;
502           sp:text """"SELECT COUNT(DISTINCT ?report)
503 WHERE {

```

```

504     OPTIONAL {
505         ?arg1 mim:contains ?report .
506         ?report mim:reports ?arg2 .
507     } .
508     OPTIONAL {
509         ?arg1 (mim:contains)* ?z .
510         ?z mim:reports ?arg2 .
511         ?z mim:withValue ?report .
512     } .
513 }""^^xsd:string
514     ] ;
515     spin:constraint
516         [ rdf:type spl:Argument ;
517           rdfs:comment "The Requirement to be tested."^^xsd:string ;
518           spl:predicate sp:arg2
519         ] ;
520     spin:constraint
521         [ rdf:type spl:Argument ;
522           rdfs:comment "The ReportSet to be tested."^^xsd:string ;
523           spl:predicate sp:arg1
524         ] ;
525     spin:returnType xsd:integer .
527 mimspin:satisfyingReportCount
528     rdf:type spin:Function ;
529     rdfs:subClassOf mimspin:Functions ;
530     spin:body
531         [ rdf:type sp:Select ;
532           sp:text ""# ReportSet reports Requirement count
533 SELECT COUNT(DISTINCT ?report)
534 WHERE {
535     ?arg1 a mim:ReportSet .
536     OPTIONAL {
537         ?arg1 mim:contains ?report .
538         ?report mim:satisfies ?arg2 .
539     } .
540     OPTIONAL {
541         ?arg1 (mim:contains)* ?report .
542         ?report mim:satisfies ?arg2 .
543         ?report mim:withValue ?v .
544     } .
545 }""^^xsd:string
546     ] ;
547     spin:constraint
548         [ rdf:type spl:Argument ;
549           rdfs:comment "The Requirement to be tested."^^xsd:string ;
550           spl:predicate sp:arg2
551         ] ;
552     spin:constraint
553         [ rdf:type spl:Argument ;
554           rdfs:comment "The ReportSet to be tested."^^xsd:string ;
555           spl:predicate sp:arg1
556         ] ;
557     spin:returnType xsd:integer .

```

```

559
559 mimspin:violatesCardinality
560     rdf:type spin:Function ;
561     rdfs:subClassOf mimspin:Functions ;
562     spin:abstract "true"^^xsd:boolean .
564
564 mimspin:violatesDatatypeRestriction
565     rdf:type spin:Function ;
566     rdfs:subClassOf mimspin:violatesRestriction ;
567     spin:body
568         [ rdf:type sp:Ask ;
569           sp:text ""ASK WHERE {
570 ?arg1 mim:hasRestriction ?r .
571 ?r mim:type ?t .
572 BIND ((datatype(?arg2) != ?t) AS ?result) .
573 FILTER (?result) .
574 }""^^xsd:string
575         ] ;
576     spin:constraint
577         [ rdf:type spl:Argument ;
578           rdfs:comment "The Requirement to be tested."^^xsd:string ;
579           spl:predicate sp:arg2
580         ] ;
581     spin:constraint
582         [ rdf:type spl:Argument ;
583           rdfs:comment "The ReportSet to be tested."^^xsd:string ;
584           spl:predicate sp:arg1
585         ] ;
586     spin:returnType xsd:integer .
588
588 mimspin:violatesDefaultCardinality
589     rdf:type spin:Function ;
590     rdfs:subClassOf mimspin:violatesCardinality ;
591     spin:body
592         [ rdf:type sp:Ask ;
593           sp:text ""ASK WHERE {
594 ?arg2 a mim:RequirementSet .
595 OPTIONAL {
596     ?arg2 mim:hasRestriction ?Restr .
597     ?Restr mim:onRequirement ?arg1 .
598 } .
599 MINUS {
600     ?Restr mim:maxCardinality ?i .
601 } .
602 MINUS {
603     ?Restr mim:minCardinality ?i .
604 } .
605 MINUS {
606     ?Restr mim:exactCardinality ?i .
607 } .
608 FILTER (?arg3 < 1) .
609 }""^^xsd:string
610         ] ;
611     spin:constraint

```

```

612         [ rdf:type spl:Argument ;
613           rdfs:comment "The Requirement being checked."^^xsd:string ;
614           spl:predicate sp:arg3
615         ] ;
616     spin:constraint
617         [ rdf:type spl:Argument ;
618           rdfs:comment "The number of times this ReportSet reports the
619             Requirement"^^xsd:string ;
620           spl:predicate sp:arg2
621         ] ;
622     spin:constraint
623         [ rdf:type spl:Argument ;
624           rdfs:comment "The ReportSet being checked."^^xsd:string ;
625           spl:predicate sp:arg1
626         ] ;
627     spin:returnType xsd:boolean .
628
629 mimspin:violatesExactCardinality
630     rdf:type spin:Function ;
631     rdfs:subClassOf mimspin:violatesCardinality ;
632     spin:body
633         [ rdf:type sp:Ask ;
634           sp:text """ASK WHERE {
635             ?arg2 a mim:RequirementSet .
636             ?arg2 mim:hasRestriction ?Restr .
637             ?Restr mim:onRequirement ?arg1 .
638             ?Restr mim:exactCardinality ?exact .
639             FILTER (?arg3 != ?exact) .
640           }"""^^xsd:string
641         ] ;
642     spin:constraint
643         [ rdf:type spl:Argument ;
644           rdfs:comment "The Requirement being checked."^^xsd:string ;
645           spl:predicate sp:arg3
646         ] ;
647     spin:constraint
648         [ rdf:type spl:Argument ;
649           rdfs:comment "The number of times this ReportSet reports the
650             Requirement"^^xsd:string ;
651           spl:predicate sp:arg2
652         ] ;
653     spin:constraint
654         [ rdf:type spl:Argument ;
655           rdfs:comment "The ReportSet being checked."^^xsd:string ;
656           spl:predicate sp:arg1
657         ] ;
658     spin:returnType xsd:boolean .
659
660 mimspin:violatesMaxCardinality
661     rdf:type spin:Function ;
662     rdfs:subClassOf mimspin:violatesCardinality ;
663     spin:body
664         [ rdf:type sp:Ask ;
665           sp:text """ASK WHERE {

```

```

664     ?arg2 a mim:RequirementSet .
665     ?arg2 mim:hasRestriction ?Restr .
666     ?Restr mim:onRequirement ?arg1 .
667     ?Restr mim:maxCardinality ?max .
668     FILTER (?arg3 > ?max) .
669 }""^^xsd:string
670     ] ;
671     spin:constraint
672         [ rdf:type spl:Argument ;
673           rdfs:comment "The number of times this ReportSet reports the
674             Requirement"^^xsd:string ;
675           spl:predicate sp:arg2
676         ] ;
677     spin:constraint
678         [ rdf:type spl:Argument ;
679           rdfs:comment "The Requirement being checked."^^xsd:string ;
680           spl:predicate sp:arg3
681         ] ;
682     spin:constraint
683         [ rdf:type spl:Argument ;
684           rdfs:comment "The ReportSet being checked."^^xsd:string ;
685           spl:predicate sp:arg1
686         ] ;
687     spin:returnType xsd:boolean .
688
689 mimspin:violatesMinCardinality
690     rdf:type spin:Function ;
691     rdfs:subClassOf mimspin:violatesCardinality ;
692     spin:body
693         [ rdf:type sp:Ask ;
694           sp:text ""ASK WHERE {
695             ?arg2 a mim:RequirementSet .
696             ?arg2 mim:hasRestriction ?Restr .
697             ?Restr mim:onRequirement ?arg1 .
698             ?Restr mim:minCardinality ?min .
699             FILTER (?arg3 < ?min) .
700           }""^^xsd:string
701         ] ;
702     spin:constraint
703         [ rdf:type spl:Argument ;
704           rdfs:comment "The ReportSet being checked."^^xsd:string ;
705           spl:predicate sp:arg1
706         ] ;
707     spin:constraint
708         [ rdf:type spl:Argument ;
709           rdfs:comment "The Requirement being checked."^^xsd:string ;
710           spl:predicate sp:arg3
711         ] ;
712     spin:constraint
713         [ rdf:type spl:Argument ;
714           rdfs:comment "The number of times this ReportSet reports the
715             Requirement"^^xsd:string ;
716           spl:predicate sp:arg2
717         ] ;

```

```

716     spin:returnType xsd:boolean .
718
718 mimspin:violatesObjectInstanceOfRestriction
719     rdf:type spin:Function ;
720     rdfs:subClassOf mimspin:violatesRestriction ;
721     spin:body
722         [ rdf:type sp:Ask ;
723           sp:text ""ASK WHERE {
724             ?arg1 mim:hasRestriction ?Restr .
725             ?Restr mim:instanceOf ?i .
726             BIND ((!<http://spinrdf.org/spl#instanceOf(?arg2, ?i)) AS ?result) .
727             FILTER (?result) .
728           }""^^xsd:string
729         ] ;
730     spin:constraint
731         [ rdf:type spl:Argument ;
732           rdfs:comment "The ReportSet being checked."^^xsd:string ;
733           spl:predicate sp:arg1
734         ] ;
735     spin:constraint
736         [ rdf:type spl:Argument ;
737           rdfs:comment "The Requirement being checked."^^xsd:string ;
738           spl:predicate sp:arg3
739         ] ;
740     spin:constraint
741         [ rdf:type spl:Argument ;
742           rdfs:comment "The number of times this ReportSet reports the
743             Requirement"^^xsd:string ;
744           spl:predicate sp:arg2
745         ] ;
746     spin:returnType xsd:boolean .
747
747 mimspin:violatesRestriction
748     rdf:type spin:Function ;
749     rdfs:subClassOf mimspin:Functions ;
750     spin:abstract "true"^^xsd:boolean .
751
752 spin:onceRule
753     rdf:type spin:RuleProperty ;
754     rdfs:subPropertyOf spin:rule ;
755     spin:rulePropertyMaxIterationCount
756         1 .
757
758 owl:Thing
759     spin:rule
760         [ rdf:type mimspin:DataReport
761         ] ;
762     spin:rule
763         [ rdf:type mimspin:DataRequirementSatisfaction
764         ] ;
765     spin:rule
766         [ rdf:type mimspin:ObjectRequirementSatisfaction
767         ] ;
768     spin:rule

```

```

769         [ rdf:type mimspin:RequirementSet
770         ] ;
771     spin:rule
772         [ rdf:type mimspin:Report
773         ] ;
774     spin:rule
775         [ rdf:type mimspin:adequatelySastisfies
776         ] ;
777     spin:rule
778         [ rdf:type mimspin:ReportSet
779         ] ;
780     spin:rule
781         [ rdf:type mimspin:ContainsReport
782         ] ;
783     spin:rule
784         [ rdf:type mimspin:Contains
785         ] ;
786     spin:rule
787         [ rdf:type mimspin:additionallySastisfies
788         ] ;
789     spin:rule
790         [ rdf:type mimspin:ReportSetContainsSelf
791         ] ;
792     spin:rule
793         [ rdf:type mimspin:minimallySastisfies
794         ] ;
795     spin:rule
796         [ rdf:type mimspin:maximallySastisfies
797         ] ;
798     spin:rule
799         [ rdf:type mimspin:Satisfies
800         ] .
802
802 []    sp:object
803         [ sp:varName "Req"^^xsd:string
804         ] ;
805     sp:predicate mim:contains ;
806     sp:subject spin:_arg1 .
808
808 []    sp:object _:b1 ;
809     sp:predicate mim:contains ;
810     sp:subject spin:_arg2 .
812
812 _:b1  sp:varName "Rep"^^xsd:string .
814
814 []    sp:object _:b2 ;
815     sp:predicate mim:contains ;
816     sp:subject _:b1 .
818
818 _:b2  sp:varName "DRep"^^xsd:string .
820
820 []    sp:object mim:DataReport ;
821     sp:predicate rdf:type ;
822     sp:subject _:b2 .

```

```

824
824 []      sp:object
825           [ sp:varName "Req"^^xsd:string
826               ] ;
827     sp:predicate mim:contains ;
828     sp:subject spin:_arg1 .
830
830 []      sp:object _:b3 ;
831     sp:predicate mim:contains ;
832     sp:subject spin:_arg2 .
834
834 _:b3    sp:varName "Rep"^^xsd:string .
836
836 []      sp:object _:b4 ;
837     sp:predicate mim:contains ;
838     sp:subject _:b3 .
840
840 _:b4    sp:varName "DRep"^^xsd:string .
842
842 []      sp:object mim:DataReport ;
843     sp:predicate rdf:type ;
844     sp:subject _:b4 .
846
846 []      sp:object
847           [ sp:varName "Req"^^xsd:string
848               ] ;
849     sp:predicate mim:contains ;
850     sp:subject spin:_arg1 .
852
852 []      sp:object _:b5 ;
853     sp:predicate mim:contains ;
854     sp:subject spin:_arg2 .
856
856 _:b5    sp:varName "Rep"^^xsd:string .
858
858 []      sp:object _:b6 ;
859     sp:predicate mim:contains ;
860     sp:subject _:b5 .
862
862 _:b6    sp:varName "DRep"^^xsd:string .
864
864 []      sp:object mim:DataReport ;
865     sp:predicate rdf:type ;
866     sp:subject _:b6 .

```


Appendix E

The Evident Vocabulary

```
1 @prefix : <http://purl.org/net/evident#> .
2 @prefix owl: <http://www.w3.org/2002/07/owl#> .
3 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
4 @prefix xml: <http://www.w3.org/XML/1998/namespace> .
5 @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
6 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
7 @base <http://purl.org/net/evident> .
8
9 <http://purl.org/net/evident> rdf:type owl:Ontology ;
10
11     owl:imports <http://www.pr-owl.org/pr-owl2.owl> ,
12                 <http://www.w3.org/ns/prov-o#> .
13
14 #####
15 #
16 #   Object Properties
17 #
18 #####
19
20
21
22
23 ### http://purl.org/net/evident#qualifiedMetric
24
25 :qualifiedMetric rdf:type owl:ObjectProperty ;
26
27     rdfs:range :Metric ;
28
29     rdfs:domain <http://www.pr-owl.org/pr-owl2.owl#ResidentNode> ;
30
31     rdfs:subPropertyOf owl:topObjectProperty .
32
33
34
35
36 #####
```

```

37 #
38 #   Data properties
39 #
40 #####
41
42
43
43 ###   http://purl.org/net/evident#continuousInfluenceFactor
45
45 :continuousInfluenceFactor rdf:type owl:DatatypeProperty ;
47
47         rdfs:subPropertyOf :influenceFactor ;
49
49         rdfs:domain <http://www.w3.org/ns/prov#Influence> .
53
53
53
53 ###   http://purl.org/net/evident#definesMetricFor
55
55 :definesMetricFor rdf:type owl:DatatypeProperty ;
57
57         rdfs:domain <http://www.pr-owl.org/pr-owl2.owl#ResidentNode> ;
59
59         rdfs:range xsd:anyURI .
63
63
63
63 ###   http://purl.org/net/evident#discreteInfluenceFactor
65
65 :discreteInfluenceFactor rdf:type owl:DatatypeProperty ;
67
67         rdfs:subPropertyOf :influenceFactor ;
69
69         rdfs:domain <http://www.w3.org/ns/prov#Influence> .
73
73
73
73 ###   http://purl.org/net/evident#influenceFactor
75
75 :influenceFactor rdf:type owl:DatatypeProperty ;
77
77         rdfs:domain <http://www.w3.org/ns/prov#Influence> .
81
81
81
81 ###   http://purl.org/net/evident#influenceType
83
83 :influenceType rdf:type owl:DatatypeProperty ;
85
85         rdfs:domain :Metric ;
87
87         rdfs:range xsd:anyURI .
91
91
91

```

```

91 ### http://purl.org/net/evident#influencee
93
93 :influencee rdf:type owl:DatatypeProperty ;
95
95         rdfs:domain :Metric ;
97
97         rdfs:range xsd:anyURI .
101
101
101
101 ### http://purl.org/net/evident#influencer
103
103 :influencer rdf:type owl:DatatypeProperty ;
105
105         rdfs:domain :Metric ;
107
107         rdfs:range xsd:anyURI .
111
111
111
111 ### http://purl.org/net/evident#normalInfluenceFactor
113
113 :normalInfluenceFactor rdf:type owl:DatatypeProperty ;
115
115         rdfs:subPropertyOf :continuousInfluenceFactor ;
117
117         rdfs:range xsd:double ;
119
119         rdfs:domain <http://www.w3.org/ns/prov#Influence> .
122
122
122 #####
123 #
124 #   Classes
125 #
126 #####
129
129
129 ### http://purl.org/net/evident#Metric
131
131 :Metric rdf:type owl:Class .
135
135
135
135 ### http://purl.org/net/evident#QualityMFrag
137
137 :QualityMFrag rdf:type owl:Class ;
139
139         rdfs:subClassOf <http://www.pr-owl.org/pr-owl2.owl#MFrag> .
144
144
144
144 ### Generated by the OWL API (version 3.4.2) http://owlapi.sourceforge.net

```

Appendix F

GAQ Multi-Entity Bayesian Network

```
1 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2 @prefix : http://www.w3.org/2002/07/owl#> .
3 @prefix xml: http://www.w3.org/XML/1998/namespace> .
4 @prefix xsd: http://www.w3.org/2001/XMLSchema#> .
5 @prefix rdfs: http://www.w3.org/2000/01/rdf-schema#> .
6 @prefix gaqmetric: <file:/Users/matthewgamble/Dropbox/PhD/Code/bayes/UnbBayes/
   gaqmetric.owl#>.
7 @prefix pr-owl2: http://www.pr-owl.org/pr-owl2.owl# .
8 @prefix evident: <http://purl.org/net/evident#> .
10
10 <file:/Users/matthewgamble/Dropbox/PhD/Code/bayes/UnbBayes/gaqmetric.owl>
11   a :Ontology ;
12   :imports <http://www.pr-owl.org/pr-owl2.owl> .
14
14 gaqmetric:CX1
15   pr-owl2:hasMExpression gaqmetric:MEXPRESSION_CX1 ;
16   pr-owl2:isContextNodeIn gaqmetric:Domain_MFrag.productGAQ ;
17   a pr-owl2:ContextNode, :NamedIndividual ;
18   rdfs:comment "CX1"^^xsd:string .
20
20 gaqmetric:CX1_1
21   pr-owl2:hasArgumentNumber 1 ;
22   pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_CX1 ;
23   pr-owl2:typeOfArgument gaqmetric:productGAQ.a ;
24   a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
26
26 gaqmetric:CX1_2
27   pr-owl2:hasArgumentNumber 2 ;
28   pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_CX1 ;
29   pr-owl2:typeOfArgument pr-owl2:CX1_2_inner ;
30   a pr-owl2:MExpressionArgument, :NamedIndividual .
32
32 gaqmetric:CX1_2_inner_1
```

```

33     pr-owl2:hasArgumentNumber 1 ;
34     pr-owl2:isArgumentOf pr-owl2:CX1_2_inner ;
35     pr-owl2:typeOfArgument gaqmetric:productGAQ.p ;
36     a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
37
38 gaqmetric:CX2
39     pr-owl2:hasMExpression gaqmetric:MEXPRESSION_CX2 ;
40     pr-owl2:isContextNodeIn gaqmetric:Domain_MFrag.meanGAQ ;
41     a pr-owl2:ContextNode, :NamedIndividual ;
42     rdfs:comment "CX2"^^xsd:string .
43
44 gaqmetric:CX2_1
45     pr-owl2:hasArgumentNumber 1 ;
46     pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_CX2 ;
47     pr-owl2:typeOfArgument gaqmetric:meanGAQ.p ;
48     a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
49
50 gaqmetric:CX2_2
51     pr-owl2:hasArgumentNumber 2 ;
52     pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_CX2 ;
53     pr-owl2:typeOfArgument pr-owl2:CX2_2_inner ;
54     a pr-owl2:MExpressionArgument, :NamedIndividual .
55
56 gaqmetric:CX2_2_inner_1
57     pr-owl2:hasArgumentNumber 1 ;
58     pr-owl2:isArgumentOf pr-owl2:CX2_2_inner ;
59     pr-owl2:typeOfArgument gaqmetric:meanGAQ.s ;
60     a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
61
62 gaqmetric:CX3
63     pr-owl2:hasMExpression gaqmetric:MEXPRESSION_CX3 ;
64     pr-owl2:isContextNodeIn gaqmetric:Domain_MFrag.groupGAQ ;
65     a pr-owl2:ContextNode, :NamedIndividual ;
66     rdfs:comment "CX3"^^xsd:string .
67
68 gaqmetric:CX3_1
69     pr-owl2:hasArgumentNumber 1 ;
70     pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_CX3 ;
71     pr-owl2:typeOfArgument gaqmetric:groupGAQ.p ;
72     a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
73
74 gaqmetric:CX3_2
75     pr-owl2:hasArgumentNumber 2 ;
76     pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_CX3 ;
77     pr-owl2:typeOfArgument pr-owl2:CX3_2_inner ;
78     a pr-owl2:MExpressionArgument, :NamedIndividual .
79
80 gaqmetric:CX3_2_inner_1
81     pr-owl2:hasArgumentNumber 1 ;
82     pr-owl2:isArgumentOf pr-owl2:CX3_2_inner ;
83     pr-owl2:typeOfArgument gaqmetric:groupGAQ.s ;
84     a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
85
86 gaqmetric:Defined

```

```

87     a :NamedIndividual, :Thing .
89
89 gaqmetric:Domain_MFrag.GAQScore_MFrag
90     pr-owl2:hasOrdinaryVariable gaqmetric:GAQScore_MFrag.a ;
91     pr-owl2:hasResidentNode gaqmetric:Domain_Res.ECR,
92                             gaqmetric:Domain_Res.GAQScore,
93                             gaqmetric:Domain_Res.depth,
94                             gaqmetric:Domain_Res.process ;
95     pr-owl2:isMFragOf gaqmetric:MEBN ;
96     a pr-owl2:DomainMFrag, :NamedIndividual ;
97     rdfs:comment "The GAQ score MFrag"^^xsd:string .
99
99 gaqmetric:Domain_MFrag.groupGAQ
100    pr-owl2:hasContextNode gaqmetric:CX3 ;
101    pr-owl2:hasInputNode gaqmetric:IX4 ;
102    pr-owl2:hasOrdinaryVariable gaqmetric:groupGAQ.p, gaqmetric:groupGAQ.s ;
103    pr-owl2:hasResidentNode gaqmetric:Domain_Res.groupGAQ, gaqmetric:Domain_Res.
104                             group_contains ;
105    pr-owl2:isMFragOf gaqmetric:MEBN ;
106    a pr-owl2:DomainMFrag, :NamedIndividual .
107
107 gaqmetric:Domain_MFrag.meanGAQ
108    pr-owl2:hasContextNode gaqmetric:CX2 ;
109    pr-owl2:hasInputNode gaqmetric:IX2 ;
110    pr-owl2:hasOrdinaryVariable gaqmetric:meanGAQ.p, gaqmetric:meanGAQ.s ;
111    pr-owl2:hasResidentNode gaqmetric:Domain_Res.contains, gaqmetric:Domain_Res.
112                             meanGAQ ;
113    pr-owl2:isMFragOf gaqmetric:MEBN ;
114    a pr-owl2:DomainMFrag, :NamedIndividual ;
115    rdfs:comment ""^^xsd:string .
116
116 gaqmetric:Domain_MFrag.productGAQ
117    pr-owl2:hasContextNode gaqmetric:CX1 ;
118    pr-owl2:hasInputNode gaqmetric:IX1 ;
119    pr-owl2:hasOrdinaryVariable gaqmetric:productGAQ.a, gaqmetric:productGAQ.p ;
120    pr-owl2:hasResidentNode gaqmetric:Domain_Res.go_annotation, gaqmetric:
121                             Domain_Res.productGAQ ;
122    pr-owl2:isMFragOf gaqmetric:MEBN ;
123    a pr-owl2:DomainMFrag, :NamedIndividual ;
124    rdfs:comment ""^^xsd:string .
125
125 gaqmetric:Domain_Res.ECR
126    pr-owl2:hasMExpression gaqmetric:MEXPRESSION_ECR ;
127    pr-owl2:hasParent gaqmetric:Domain_Res.process ;
128    pr-owl2:hasProbabilityDistribution pr-owl2:ECR_Table ;
129    pr-owl2:isResidentNodeIn gaqmetric:Domain_MFrag.GAQScore_MFrag ;
130    a pr-owl2:ContinuousResidentNode, :NamedIndividual ;
131    rdfs:comment "Continuous resident for ECR"^^xsd:string .
133
133 gaqmetric:Domain_Res.GAQScore
134    pr-owl2:hasInputInstance gaqmetric:IX1 ;
135    pr-owl2:hasMExpression gaqmetric:MEXPRESSION_GAQScore ;
136    pr-owl2:hasParent gaqmetric:Domain_Res.ECR, gaqmetric:Domain_Res.depth ;
137    pr-owl2:hasProbabilityDistribution pr-owl2:GAQScore_Table ;

```

```

138     pr-owl2:isResidentNodeIn gaqmetric:Domain_MFrag.GAQScore_MFrag ;
139     a pr-owl2:ContinuousResidentNode, :NamedIndividual ;
140     rdfs:comment "Continuous resident node0"^^xsd:string .
141
142 gaqmetric:Domain_Res.contains
143     pr-owl2:hasMExpression gaqmetric:MEXPRESSION_contains ;
144     pr-owl2:isResidentNodeIn gaqmetric:Domain_MFrag.meanGAQ ;
145     a pr-owl2:DomainResidentNode, :NamedIndividual ;
146     rdfs:comment "RX10"^^xsd:string .
147
148 gaqmetric:Domain_Res.depth
149     pr-owl2:hasMExpression gaqmetric:MEXPRESSION_depth ;
150     pr-owl2:hasProbabilityDistribution pr-owl2:depth_Table ;
151     pr-owl2:isResidentNodeIn gaqmetric:Domain_MFrag.GAQScore_MFrag ;
152     a pr-owl2:ContinuousResidentNode, :NamedIndividual ;
153     rdfs:comment "Continuous resident for depth"^^xsd:string .
154
155 gaqmetric:Domain_Res.go_annotation
156     pr-owl2:hasMExpression gaqmetric:MEXPRESSION_go_annotation ;
157     pr-owl2:isResidentNodeIn gaqmetric:Domain_MFrag.productGAQ ;
158     a pr-owl2:DomainResidentNode, :NamedIndividual ;
159     rdfs:comment ""^^xsd:string .
160
161 gaqmetric:Domain_Res.groupGAQ
162     pr-owl2:hasMExpression gaqmetric:MEXPRESSION_groupGAQ ;
163     pr-owl2:hasParent gaqmetric:IX4 ;
164     pr-owl2:hasProbabilityDistribution pr-owl2:groupGAQ_Table ;
165     pr-owl2:isResidentNodeIn gaqmetric:Domain_MFrag.groupGAQ ;
166     a pr-owl2:ContinuousResidentNode, :NamedIndividual ;
167     rdfs:comment "Continuous resident node3"^^xsd:string .
168
169 gaqmetric:Domain_Res.group_contains
170     pr-owl2:hasMExpression gaqmetric:MEXPRESSION_group_contains ;
171     pr-owl2:isResidentNodeIn gaqmetric:Domain_MFrag.groupGAQ ;
172     a pr-owl2:DomainResidentNode, :NamedIndividual ;
173     rdfs:comment "RX12"^^xsd:string .
174
175 gaqmetric:Domain_Res.meanGAQ
176     pr-owl2:hasMExpression gaqmetric:MEXPRESSION_meanGAQ ;
177     pr-owl2:hasParent gaqmetric:IX2 ;
178     pr-owl2:hasProbabilityDistribution pr-owl2:meanGAQ_Table ;
179     pr-owl2:isResidentNodeIn gaqmetric:Domain_MFrag.meanGAQ ;
180     a pr-owl2:ContinuousResidentNode, :NamedIndividual ;
181     rdfs:comment "Continuous resident node2"^^xsd:string .
182
183 gaqmetric:Domain_Res.process
184     pr-owl2:hasMExpression gaqmetric:MEXPRESSION_process ;
185     pr-owl2:isResidentNodeIn gaqmetric:Domain_MFrag.GAQScore_MFrag ;
186     a pr-owl2:DomainResidentNode, :NamedIndividual ;
187     rdfs:comment "RX4"^^xsd:string .
188
189 gaqmetric:Domain_Res.productGAQ
190     pr-owl2:hasInputInstance gaqmetric:IX2, gaqmetric:IX4 ;
191     pr-owl2:hasMExpression gaqmetric:MEXPRESSION_productGAQ ;

```

```

192     pr-owl2:hasParent gaqmetric:IX1 ;
193     pr-owl2:hasProbabilityDistribution pr-owl2:productGAQ_Table ;
194     pr-owl2:isResidentNodeIn gaqmetric:Domain_MFrag.productGAQ ;
195     a pr-owl2:ContinuousResidentNode, :NamedIndividual ;
196     rdfs:comment "Continuous resident node0"^^xsd:string .
197
198 gaqmetric:ECR_1
199     pr-owl2:hasArgumentNumber 1 ;
200     pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_ECR ;
201     pr-owl2:typeOfArgument gaqmetric:GAQScore_MFrag.a ;
202     a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
203
204 gaqmetric:EXPe
205     a :NamedIndividual, :Thing .
206
207 gaqmetric:GAQScore_1
208     pr-owl2:hasArgumentNumber 1 ;
209     pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_GAQScore ;
210     pr-owl2:typeOfArgument gaqmetric:GAQScore_MFrag.a ;
211     a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
212
213 gaqmetric:GAQScore_MFrag.a
214     pr-owl2:isSubstitutedBy "http://purl.uniprot.org/core#Protein"^^xsd:anyURI ;
215     pr-owl2:isTypeOfArgumentIn gaqmetric:ECR_1, gaqmetric:GAQScore_1, gaqmetric:
        depth_1, gaqmetric:process_1 ;
216     a pr-owl2:OrdinaryVariable, :NamedIndividual ;
217     rdfs:comment "OX1"^^xsd:string .
218
219 gaqmetric:IBA
220     a :NamedIndividual, :Thing .
221
222 gaqmetric:IBD
223     a :NamedIndividual, :Thing .
224
225 gaqmetric:IC
226     a :NamedIndividual, :Thing .
227
228 gaqmetric:IDA
229     a :NamedIndividual, :Thing .
230
231 gaqmetric:IEA
232     a :NamedIndividual, :Thing .
233
234 gaqmetric:IEP
235     a :NamedIndividual, :Thing .
236
237 gaqmetric:IGC
238     a :NamedIndividual, :Thing .
239
240 gaqmetric:IGI
241     a :NamedIndividual, :Thing .
242
243 gaqmetric:IKR
244     a :NamedIndividual, :Thing .

```



```

246
246 gaqmetric:IMP
247     a :NamedIndividual, :Thing .
249
249 gaqmetric:IMR
250     a :NamedIndividual, :Thing .
252
252 gaqmetric:IPI
253     a :NamedIndividual, :Thing .
255
255 gaqmetric:IRD
256     a :NamedIndividual, :Thing .
258
258 gaqmetric:ISA
259     a :NamedIndividual, :Thing .
261
261 gaqmetric:ISM
262     a :NamedIndividual, :Thing .
264
264 gaqmetric:ISO
265     a :NamedIndividual, :Thing .
267
267 gaqmetric:ISS
268     a :NamedIndividual, :Thing .
270
270 gaqmetric:IX1
271     pr-owl2:hasMExpression gaqmetric:MEXPRESSION_IX1 ;
272     pr-owl2:isInputNodeIn gaqmetric:Domain_MFrag.productGAQ ;
273     a pr-owl2:GenerativeInputNode, :NamedIndividual ;
274     rdfs:comment "IX1"^^xsd:string .
276
276 gaqmetric:IX1_1
277     pr-owl2:hasArgumentNumber 1 ;
278     pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_IX1 ;
279     pr-owl2:typeOfArgument gaqmetric:productGAQ.a ;
280     a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
282
282 gaqmetric:IX2
283     pr-owl2:hasMExpression gaqmetric:MEXPRESSION_IX2 ;
284     pr-owl2:isInputNodeIn gaqmetric:Domain_MFrag.meanGAQ ;
285     a pr-owl2:GenerativeInputNode, :NamedIndividual ;
286     rdfs:comment "IX2"^^xsd:string .
288
288 gaqmetric:IX2_1
289     pr-owl2:hasArgumentNumber 1 ;
290     pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_IX2 ;
291     pr-owl2:typeOfArgument gaqmetric:meanGAQ.p ;
292     a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
294
294 gaqmetric:IX4
295     pr-owl2:hasMExpression gaqmetric:MEXPRESSION_IX4 ;
296     pr-owl2:isInputNodeIn gaqmetric:Domain_MFrag.groupGAQ ;
297     a pr-owl2:GenerativeInputNode, :NamedIndividual ;
298     rdfs:comment "IX4"^^xsd:string .

```

```

300
300 gaqmetric:IX4_1
301     pr-owl2:hasArgumentNumber 1 ;
302     pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_IX4 ;
303     pr-owl2:typeOfArgument gaqmetric:groupGAQ.p ;
304     a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
306
306 gaqmetric:MEBN
307     pr-owl2:hasMFragment gaqmetric:Domain_MFrag.GAQScore_MFrag, gaqmetric:
308         Domain_MFrag.groupGAQ, gaqmetric:Domain_MFrag.meanGAQ, gaqmetric:
309         Domain_MFrag.productGAQ ;
308     a pr-owl2:MTheory, :NamedIndividual ;
309     rdfs:comment "^^xsd:string .
311
311 gaqmetric:MEXPRESSION_CX1
312     pr-owl2:hasArgument gaqmetric:CX1_1, gaqmetric:CX1_2 ;
313     pr-owl2:isMExpressionOf gaqmetric:CX1 ;
314     pr-owl2:typeOfMExpression pr-owl2:equalTo ;
315     a pr-owl2:BooleanMExpression, pr-owl2:MExpression, :NamedIndividual .
317
317 gaqmetric:MEXPRESSION_CX2
318     pr-owl2:hasArgument gaqmetric:CX2_1, gaqmetric:CX2_2 ;
319     pr-owl2:isMExpressionOf gaqmetric:CX2 ;
320     pr-owl2:typeOfMExpression pr-owl2:equalTo ;
321     a pr-owl2:BooleanMExpression, pr-owl2:MExpression, :NamedIndividual .
323
323 gaqmetric:MEXPRESSION_CX3
324     pr-owl2:hasArgument gaqmetric:CX3_1, gaqmetric:CX3_2 ;
325     pr-owl2:isMExpressionOf gaqmetric:CX3 ;
326     pr-owl2:typeOfMExpression pr-owl2:equalTo ;
327     a pr-owl2:BooleanMExpression, pr-owl2:MExpression, :NamedIndividual .
329
329 gaqmetric:MEXPRESSION_ECR
330     pr-owl2:hasArgument gaqmetric:ECR_1 ;
331     pr-owl2:isMExpressionOf gaqmetric:Domain_Res.ECR ;
332     pr-owl2:typeOfMExpression gaqmetric:RV_ECR ;
333     a pr-owl2:SimpleMExpression, :NamedIndividual .
335
335 gaqmetric:MEXPRESSION_GAQScore
336     pr-owl2:hasArgument gaqmetric:GAQScore_1 ;
337     pr-owl2:isMExpressionOf gaqmetric:Domain_Res.GAQScore ;
338     pr-owl2:typeOfMExpression gaqmetric:RV_GAQScore ;
339     a pr-owl2:SimpleMExpression, :NamedIndividual .
341
341 gaqmetric:MEXPRESSION_IX1
342     pr-owl2:hasArgument gaqmetric:IX1_1 ;
343     pr-owl2:isMExpressionOf gaqmetric:IX1 ;
344     pr-owl2:typeOfMExpression gaqmetric:RV_GAQScore ;
345     a pr-owl2:MExpression, :NamedIndividual .
347
347 gaqmetric:MEXPRESSION_IX2
348     pr-owl2:hasArgument gaqmetric:IX2_1 ;
349     pr-owl2:isMExpressionOf gaqmetric:IX2 ;
350     pr-owl2:typeOfMExpression gaqmetric:RV_productGAQ ;

```

```

351     a pr-owl2:MExpression, :NamedIndividual .
353
353 gaqmetric:MEXPRESSION_IX4
354     pr-owl2:hasArgument gaqmetric:IX4_1 ;
355     pr-owl2:isMExpressionOf gaqmetric:IX4 ;
356     pr-owl2:typeOfMExpression gaqmetric:RV_productGAQ ;
357     a pr-owl2:MExpression, :NamedIndividual .
359
359 gaqmetric:MEXPRESSION_contains
360     pr-owl2:hasArgument gaqmetric:contains_1 ;
361     pr-owl2:isMExpressionOf gaqmetric:Domain_Res.contains ;
362     pr-owl2:typeOfMExpression gaqmetric:RV_contains ;
363     a pr-owl2:SimpleMExpression, :NamedIndividual .
365
365 gaqmetric:MEXPRESSION_depth
366     pr-owl2:hasArgument gaqmetric:depth_1 ;
367     pr-owl2:isMExpressionOf gaqmetric:Domain_Res.depth ;
368     pr-owl2:typeOfMExpression gaqmetric:RV_depth ;
369     a pr-owl2:SimpleMExpression, :NamedIndividual .
371
371 gaqmetric:MEXPRESSION_go_annotation
372     pr-owl2:hasArgument gaqmetric:go_annotation_1 ;
373     pr-owl2:isMExpressionOf gaqmetric:Domain_Res.go_annotation ;
374     pr-owl2:typeOfMExpression gaqmetric:RV_go_annotation ;
375     a pr-owl2:SimpleMExpression, :NamedIndividual .
377
377 gaqmetric:MEXPRESSION_groupGAQ
378     pr-owl2:hasArgument gaqmetric:groupGAQ_1 ;
379     pr-owl2:isMExpressionOf gaqmetric:Domain_Res.groupGAQ ;
380     pr-owl2:typeOfMExpression gaqmetric:RV_groupGAQ ;
381     a pr-owl2:SimpleMExpression, :NamedIndividual .
383
383 gaqmetric:MEXPRESSION_group_contains
384     pr-owl2:hasArgument gaqmetric:group_contains_1 ;
385     pr-owl2:isMExpressionOf gaqmetric:Domain_Res.group_contains ;
386     pr-owl2:typeOfMExpression gaqmetric:RV_group_contains ;
387     a pr-owl2:SimpleMExpression, :NamedIndividual .
389
389 gaqmetric:MEXPRESSION_meanGAQ
390     pr-owl2:hasArgument gaqmetric:meanGAQ_1 ;
391     pr-owl2:isMExpressionOf gaqmetric:Domain_Res.meanGAQ ;
392     pr-owl2:typeOfMExpression gaqmetric:RV_meanGAQ ;
393     a pr-owl2:SimpleMExpression, :NamedIndividual .
395
395 gaqmetric:MEXPRESSION_process
396     pr-owl2:hasArgument gaqmetric:process_1 ;
397     pr-owl2:isMExpressionOf gaqmetric:Domain_Res.process ;
398     pr-owl2:typeOfMExpression gaqmetric:RV_process ;
399     a pr-owl2:SimpleMExpression, :NamedIndividual .
401
401 gaqmetric:MEXPRESSION_productGAQ
402     pr-owl2:hasArgument gaqmetric:productGAQ_1 ;
403     pr-owl2:isMExpressionOf gaqmetric:Domain_Res.productGAQ ;
404     pr-owl2:typeOfMExpression gaqmetric:RV_productGAQ ;

```

```

405     a pr-owl2:SimpleMExpression, :NamedIndividual .
407
407 gaqmetric:NAS
408     a :NamedIndividual, :Thing .
410
410 gaqmetric:ND
411     a :NamedIndividual, :Thing .
413
413 gaqmetric:NR
414     a :NamedIndividual, :Thing .
416
416 gaqmetric:Number
417     a :Class ;
418     rdfs:subClassOf :Thing .
420
420 gaqmetric:RCA
421     a :NamedIndividual, :Thing .
423
423 gaqmetric:RV_ECR
424     pr-owl2:hasArgument gaqmetric:RV_ECR_1 ;
425     pr-owl2:hasPossibleValues "gaqmetric:Number"^^xsd:anyURI ;
426     pr-owl2:isTypeOfMExpression gaqmetric:MEXPRESSION_ECR ;
427     a pr-owl2:RandomVariable, :NamedIndividual .
429
429 gaqmetric:RV_ECR_1
430     pr-owl2:hasArgumentNumber 1 ;
431     pr-owl2:isArgumentOf gaqmetric:RV_ECR ;
432     a pr-owl2:MappingArgument, :NamedIndividual .
434
434 gaqmetric:RV_GAQScore
435     pr-owl2:hasArgument gaqmetric:RV_GAQScore_1 ;
436     pr-owl2:hasPossibleValues "gaqmetric:Number"^^xsd:anyURI ;
437     pr-owl2:isTypeOfMExpression gaqmetric:MEXPRESSION_GAQScore, gaqmetric:
         MEXPRESSION_IX1 ;
438     a pr-owl2:RandomVariable, :NamedIndividual .
440
440 gaqmetric:RV_GAQScore_1
441     pr-owl2:hasArgumentNumber 1 ;
442     pr-owl2:isArgumentOf gaqmetric:RV_GAQScore ;
443     a pr-owl2:MappingArgument, :NamedIndividual .
445
445 gaqmetric:RV_contains
446     pr-owl2:definesUncertaintyOf "http://sierra-nevada.cs.manchester.ac.uk/goa#
         group_contains"^^xsd:anyURI ;
447     pr-owl2:hasArgument gaqmetric:RV_contains_1 ;
448     pr-owl2:isTypeOfMExpression gaqmetric:MEXPRESSION_contains, pr-owl2:
         CX2_2_inner ;
449     a pr-owl2:RandomVariable, :NamedIndividual .
451
451 gaqmetric:RV_contains_1
452     pr-owl2:hasArgumentNumber 1 ;
453     pr-owl2:isArgumentOf gaqmetric:RV_contains ;
454     a pr-owl2:MappingArgument, :NamedIndividual .
456

```

```

456 gaqmetric:RV_depth
457   pr-owl2:definesUncertaintyOf
458     "http://sierra-nevada.cs.manchester.ac.uk/goa#go-depth"^^xsd:anyURI
459   ;
459   pr-owl2:hasArgument gaqmetric:RV_depth_1 ;
460   pr-owl2:hasPossibleValues "gaqmetric:Number"^^xsd:anyURI ;
461   pr-owl2:isTypeOfMExpression gaqmetric:MEXPRESSION_depth ;
462   a pr-owl2:RandomVariable, :NamedIndividual .
464
464 gaqmetric:RV_depth_1
465   pr-owl2:hasArgumentNumber 1 ;
466   pr-owl2:isArgumentOf gaqmetric:RV_depth ;
467   a pr-owl2:MappingArgument, :NamedIndividual .
469
469 gaqmetric:RV_go_annotation
470   pr-owl2:definesUncertaintyOf "http://bio2rdf.org/goa_vocabulary#go-
471     annotation"^^xsd:anyURI ;
471   pr-owl2:hasArgument gaqmetric:RV_go_annotation_1 ;
472   pr-owl2:isTypeOfMExpression gaqmetric:MEXPRESSION_go_annotation, pr-owl2:
473     CX1_2_inner ;
473   a pr-owl2:RandomVariable, :NamedIndividual .
475
475 gaqmetric:RV_go_annotation_1
476   pr-owl2:hasArgumentNumber 1 ;
477   pr-owl2:isArgumentOf gaqmetric:RV_go_annotation ;
478   a pr-owl2:MappingArgument, :NamedIndividual .
480
480 gaqmetric:RV_groupGAQ
481   pr-owl2:hasArgument gaqmetric:RV_groupGAQ_1 ;
482   pr-owl2:hasPossibleValues "gaqmetric:Number"^^xsd:anyURI ;
483   pr-owl2:isTypeOfMExpression gaqmetric:MEXPRESSION_groupGAQ ;
484   a pr-owl2:RandomVariable, :NamedIndividual .
486
486 gaqmetric:RV_groupGAQ_1
487   pr-owl2:hasArgumentNumber 1 ;
488   pr-owl2:isArgumentOf gaqmetric:RV_groupGAQ ;
489   a pr-owl2:MappingArgument, :NamedIndividual .
491
491 gaqmetric:RV_group_contains
492   pr-owl2:definesUncertaintyOf "http://sierra-nevada.cs.manchester.ac.uk/goa#
493     group_contains"^^xsd:anyURI ;
493   pr-owl2:hasArgument gaqmetric:RV_group_contains_1 ;
494   pr-owl2:isTypeOfMExpression gaqmetric:MEXPRESSION_group_contains, pr-owl2:
495     CX3_2_inner ;
495   a pr-owl2:RandomVariable, :NamedIndividual .
497
497 gaqmetric:RV_group_contains_1
498   pr-owl2:hasArgumentNumber 1 ;
499   pr-owl2:isArgumentOf gaqmetric:RV_group_contains ;
500   a pr-owl2:MappingArgument, :NamedIndividual .
502
502 gaqmetric:RV_meanGAQ
503   pr-owl2:hasArgument gaqmetric:RV_meanGAQ_1 ;
504   pr-owl2:hasPossibleValues "gaqmetric:Number"^^xsd:anyURI ;

```

```

505     pr-owl2:isTypeOfMExpression gaqmetric:MEXPRESSION_meanGAQ ;
506     a pr-owl2:RandomVariable, :NamedIndividual .
508
508 gaqmetric:RV_meanGAQ_1
509     pr-owl2:hasArgumentNumber 1 ;
510     pr-owl2:isArgumentOf gaqmetric:RV_meanGAQ ;
511     a pr-owl2:MappingArgument, :NamedIndividual .
513
513 gaqmetric:RV_process
514     pr-owl2:definesUncertaintyOf "http://sierra-nevada.cs.manchester.ac.uk/goa#
515         ec_label"^^xsd:anyURI ;
516     pr-owl2:hasArgument gaqmetric:RV_process_1 ;
517     pr-owl2:hasPossibleValues "gaqmetric:Defined"^^xsd:anyURI, "gaqmetric:EXPe"
518         ^^xsd:anyURI, "gaqmetric:IBA"^^xsd:anyURI, "gaqmetric:IBD"^^xsd:anyURI,
519         "gaqmetric:IC"^^xsd:anyURI, "gaqmetric:IDA"^^xsd:anyURI, "gaqmetric:IEA"
520         ^^xsd:anyURI, "gaqmetric:IEP"^^xsd:anyURI, "gaqmetric:IGC"^^xsd:anyURI,
521         "gaqmetric:IGI"^^xsd:anyURI, "gaqmetric:IKR"^^xsd:anyURI, "gaqmetric:IMP"
522         ^^xsd:anyURI, "gaqmetric:IMR"^^xsd:anyURI, "gaqmetric:IPI"^^xsd:anyURI,
523         "gaqmetric:IRD"^^xsd:anyURI, "gaqmetric:ISA"^^xsd:anyURI, "gaqmetric:
524         ISM"^^xsd:anyURI, "gaqmetric:ISO"^^xsd:anyURI, "gaqmetric:ISS"^^xsd:
525         anyURI, "gaqmetric:NAS"^^xsd:anyURI, "gaqmetric:ND"^^xsd:anyURI, "
526         gaqmetric:NR"^^xsd:anyURI, "gaqmetric:RCA"^^xsd:anyURI, "gaqmetric:TAS"
527         ^^xsd:anyURI ;
528     pr-owl2:isTypeOfMExpression gaqmetric:MEXPRESSION_process ;
529     a pr-owl2:RandomVariable, :NamedIndividual .
531
531 gaqmetric:RV_process_1
532     pr-owl2:hasArgumentNumber 1 ;
533     pr-owl2:isArgumentOf gaqmetric:RV_process ;
534     a pr-owl2:MappingArgument, :NamedIndividual .
536
536 gaqmetric:RV_productGAQ
537     evident:definesMetricFor
538         "http://purl.uniprot.org/core/Protein"^^xsd:anyURI ;
539     pr-owl2:hasArgument gaqmetric:RV_productGAQ_1 ;
540     pr-owl2:hasPossibleValues "gaqmetric:Number"^^xsd:anyURI ;
541     pr-owl2:isTypeOfMExpression gaqmetric:MEXPRESSION_IX2, gaqmetric:
542         MEXPRESSION_IX4, gaqmetric:MEXPRESSION_productGAQ ;
543     a pr-owl2:RandomVariable, :NamedIndividual .
545
545 gaqmetric:RV_productGAQ_1
546     pr-owl2:hasArgumentNumber 1 ;
547     pr-owl2:isArgumentOf gaqmetric:RV_productGAQ ;
548     a pr-owl2:MappingArgument, :NamedIndividual .
550
550 gaqmetric:TAS
551     a :NamedIndividual, :Thing .
553
553 gaqmetric:contains_1
554     pr-owl2:hasArgumentNumber 1 ;
555     pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_contains ;
556     pr-owl2:typeOfArgument gaqmetric:meanGAQ.s ;
557     a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
559

```

```

547 gaqmetric:depth_1
548     pr-owl2:hasArgumentNumber 1 ;
549     pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_depth ;
550     pr-owl2:typeOfArgument gaqmetric:GAQScore_MFrag.a ;
551     a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
553
553 gaqmetric:go_annotation_1
554     pr-owl2:hasArgumentNumber 1 ;
555     pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_go_annotation ;
556     pr-owl2:typeOfArgument gaqmetric:productGAQ.p ;
557     a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
559
559 gaqmetric:groupGAQ.p
560     pr-owl2:isSubstitutedBy "http://purl.uniprot.org/core#Protein"^^xsd:anyURI ;
561     pr-owl2:isTypeOfArgumentIn gaqmetric:CX3_1, gaqmetric:IX4_1 ;
562     a pr-owl2:OrdinaryVariable, :NamedIndividual ;
563     rdfs:comment "OX1"^^xsd:string .
565
565 gaqmetric:groupGAQ.s
566     pr-owl2:isSubstitutedBy "http://sierra-nevada.cs.manchester.ac.uk/goa#Group"
567     ^^xsd:anyURI ;
568     pr-owl2:isTypeOfArgumentIn gaqmetric:CX3_2_inner_1, gaqmetric:groupGAQ_1,
569     gaqmetric:group_contains_1 ;
570     a pr-owl2:OrdinaryVariable, :NamedIndividual ;
571     rdfs:comment "OX2"^^xsd:string .
573
573 gaqmetric:groupGAQ_1
574     pr-owl2:hasArgumentNumber 1 ;
575     pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_groupGAQ ;
576     pr-owl2:typeOfArgument gaqmetric:groupGAQ.s ;
577     a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
579
579 gaqmetric:group_contains_1
580     pr-owl2:hasArgumentNumber 1 ;
581     pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_group_contains ;
582     pr-owl2:typeOfArgument gaqmetric:groupGAQ.s ;
583     a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
585
585 gaqmetric:meanGAQ.p
586     pr-owl2:isSubstitutedBy "http://purl.uniprot.org/core#Protein"^^xsd:anyURI ;
587     pr-owl2:isTypeOfArgumentIn gaqmetric:CX2_1, gaqmetric:IX2_1 ;
588     a pr-owl2:OrdinaryVariable, :NamedIndividual ;
589     rdfs:comment "OX1"^^xsd:string .
591
591 gaqmetric:meanGAQ.s
592     pr-owl2:isSubstitutedBy "http://sierra-nevada.cs.manchester.ac.uk/goa#Group"
593     ^^xsd:anyURI ;
594     pr-owl2:isTypeOfArgumentIn gaqmetric:CX2_2_inner_1, gaqmetric:contains_1,
595     gaqmetric:meanGAQ_1 ;
596     a pr-owl2:OrdinaryVariable, :NamedIndividual ;
597     rdfs:comment "OX2"^^xsd:string .
599
599 gaqmetric:meanGAQ_1
600     pr-owl2:hasArgumentNumber 1 ;

```

```

597 pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_meanGAQ ;
598 pr-owl2:typeOfArgument gaqmetric:meanGAQ.s ;
599 a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
601
601 gaqmetric:process_1
602 pr-owl2:hasArgumentNumber 1 ;
603 pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_process ;
604 pr-owl2:typeOfArgument gaqmetric:GAQScore_MFrag.a ;
605 a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
607
607 gaqmetric:productGAQ.a
608 pr-owl2:isSubstitutedBy "http://bio2rdf.org/goa_vocabulary#GO-Annotation"^^
xsd:anyURI ;
609 pr-owl2:isTypeOfArgumentIn gaqmetric:CX1_1, gaqmetric:IX1_1 ;
610 a pr-owl2:OrdinaryVariable, :NamedIndividual ;
611 rdfs:comment "OX2"^^xsd:string .
613
613 gaqmetric:productGAQ.p
614 pr-owl2:isSubstitutedBy "http://purl.uniprot.org/core#Protein"^^xsd:anyURI ;
615 pr-owl2:isTypeOfArgumentIn gaqmetric:CX1_2_inner_1, gaqmetric:
go_annotation_1, gaqmetric:productGAQ_1 ;
616 a pr-owl2:OrdinaryVariable, :NamedIndividual ;
617 rdfs:comment "OX1"^^xsd:string .
619
619 gaqmetric:productGAQ_1
620 pr-owl2:hasArgumentNumber 1 ;
621 pr-owl2:isArgumentOf gaqmetric:MEXPRESSION_productGAQ ;
622 pr-owl2:typeOfArgument gaqmetric:productGAQ.p ;
623 a pr-owl2:OrdinaryVariableArgument, :NamedIndividual .
625
625 pr-owl2:CX1_2_inner
626 pr-owl2:hasArgument gaqmetric:CX1_2_inner_1 ;
627 pr-owl2:isTypeOfArgumentIn gaqmetric:CX1_2 ;
628 pr-owl2:typeOfMExpression gaqmetric:RV_go_annotation ;
629 a pr-owl2:MExpression, :NamedIndividual .
631
631 pr-owl2:CX2_2_inner
632 pr-owl2:hasArgument gaqmetric:CX2_2_inner_1 ;
633 pr-owl2:isTypeOfArgumentIn gaqmetric:CX2_2 ;
634 pr-owl2:typeOfMExpression gaqmetric:RV_contains ;
635 a pr-owl2:MExpression, :NamedIndividual .
637
637 pr-owl2:CX3_2_inner
638 pr-owl2:hasArgument gaqmetric:CX3_2_inner_1 ;
639 pr-owl2:isTypeOfArgumentIn gaqmetric:CX3_2 ;
640 pr-owl2:typeOfMExpression gaqmetric:RV_group_contains ;
641 a pr-owl2:MExpression, :NamedIndividual .
643
643 pr-owl2:ContinuousResidentNode
644 a :Class .
646
646 pr-owl2:ECR_Table
647 pr-owl2:hasDeclaration ""if any a have ( process = EXPe ) [
648 NormalDist(5,0)

```



```

649 ] else
650 if any a have ( process = IBA ) [
651 NormalDist(2,0)
652 ] else
653 if any a have ( process = IBD ) [
654 NormalDist(2,0)
655 ] else
656 if any a have ( process = IC ) [
657 NormalDist(4,0)
658 ] else
659 if any a have ( process = IDA ) [
660 NormalDist(5,0)
661 ] else
662 if any a have ( process = IEA ) [
663 NormalDist(2,0)
664 ] else
665 if any a have ( process = IEP ) [
666 NormalDist(3,0)
667 ] else
668 if any a have ( process = IGC ) [
669 NormalDist(3,0)
670 ] else
671 if any a have ( process = IGI ) [
672 NormalDist(5,0)
673 ] else
674 if any a have ( process = IKR ) [
675 NormalDist(2,0)
676 ] else
677 if any a have ( process = IMP ) [
678 NormalDist(5,0)
679 ] else
680 if any a have ( process = IMR ) [
681 NormalDist(2,0)
682 ] else
683 if any a have ( process = IPI ) [
684 NormalDist(5,0)
685 ] else
686 if any a have ( process = IRD ) [
687 NormalDist(2,0)
688 ] else
689 if any a have ( process = ISA ) [
690 NormalDist(2,0)
691 ] else
692 if any a have ( process = ISM ) [
693 NormalDist(2,0)
694 ] else
695 if any a have ( process = ISO ) [
696 NormalDist(3,0)
697 ] else
698 if any a have ( process = ISS ) [
699 NormalDist(2,0)
700 ] else
701 if any a have ( process = NAS ) [
702 NormalDist(2,0)

```

```

703 ] else
704 if any a have ( process = ND ) [
705 NormalDist(0,0)
706 ] else
707 if any a have ( process = NR ) [
708 NormalDist(1,0)
709 ] else
710 if any a have ( process = RCA ) [
711 NormalDist(3,0)
712 ] else
713 if any a have ( process = TAS ) [
714 NormalDist(4,0)
715 ]
716 else[
717 NormalDist(3,0)
718 ]
719 """^^xsd:string ;
720     a pr-owl2:DeclarativeDistribution, :NamedIndividual .
722
722 pr-owl2:GAQScore_Table
723     pr-owl2:hasDeclaration """[
724 (ECR * depth) * NormalDist(1,0.1)
725 ]
726 """^^xsd:string ;
727     a pr-owl2:DeclarativeDistribution, :NamedIndividual .
729
729 pr-owl2:contains_Table
730     a pr-owl2:DeclarativeDistribution, :NamedIndividual .
732
732 pr-owl2:depth_Table
733     pr-owl2:hasDeclaration """[
734 NormalDist(6.7,2.1)
735 ]"""^^xsd:string ;
736     a pr-owl2:DeclarativeDistribution, :NamedIndividual .
738
738 pr-owl2:equalTo
739     pr-owl2:isTypeOfMExpression gaqmetric:MEXPRESSION_CX1, gaqmetric:
740         MEXPRESSION_CX2, gaqmetric:MEXPRESSION_CX3 .
741
741 pr-owl2:go_annotation_Table
742     a pr-owl2:DeclarativeDistribution, :NamedIndividual .
744
744 pr-owl2:groupGAQ_Table
745     pr-owl2:hasDeclaration """[
746 Sum( productGAQ ) * NormalDist(1,0)
747 ]"""^^xsd:string ;
748     a pr-owl2:DeclarativeDistribution, :NamedIndividual .
750
750 pr-owl2:group_contains_Table
751     a pr-owl2:DeclarativeDistribution, :NamedIndividual .
753
753 pr-owl2:hasInputInstance
754     a :ObjectProperty .
756

```

```

756 pr-owl2:meanGAQ_Table
757     pr-owl2:hasDeclaration """[
758 Mean( productGAQ ) * NormalDist(1,0)
759 ]"""^^xsd:string ;
760     a pr-owl2:DeclarativeDistribution, :NamedIndividual .
762
762 pr-owl2:process_Table
763     a pr-owl2:DeclarativeDistribution, :NamedIndividual .
765
765 pr-owl2:productGAQ_Table
766     pr-owl2:hasDeclaration """[
767 Sum(GAQScore) * NormalDist(1,0)
768 ]"""^^xsd:string ;
769     a pr-owl2:DeclarativeDistribution, :NamedIndividual .

```

Appendix G

Wikiprov RDF Serialization Example

```
1 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2 @prefix prov: <http://www.w3.org/ns/prov#> .
3 @prefix wikiprov: <http://purl.org/net/wikiprov#> .
4 @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
5 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
6 @prefix evident: <http://purl.org/net/evident#> .
7
8 wikiprov:United_States_National_Forest_223449
9   wikiprov:id "223449"^^xsd:string ;
10  wikiprov:pageid "42653"^^xsd:string ;
11  wikiprov:revid "223449"^^xsd:string ;
12  wikiprov:title "United_States_National_Forest"^^xsd:string ;
13  a prov:Entity, wikiprov:article .
14
15 wikiprov:559734generation
16   wikiprov:activity "comment559734"^^xsd:string ;
17   wikiprov:comment ""^^xsd:string ;
18   wikiprov:entity "559734"^^xsd:string ;
19   wikiprov:id "559734generation"^^xsd:string ;
20   wikiprov:relationshipName "559734generation"^^xsd:string ;
21   wikiprov:time "2002-09-17T03:11:31Z"^^xsd:string ;
22   a prov:Generation ;
23   prov:activity wikiprov:activity_559734 ;
24   prov:atTime "2002-09-17T03:11:31Z"^^xsd:dateTime .
25
26 wikiprov:Mav
27   wikiprov:id "Mav"^^xsd:string ;
28   wikiprov:user_name "Mav"^^xsd:string ;
29   a wikiprov:editor, prov:Agent .
30
31 wikiprov:Mav_559734
32   a prov:Attribution ;
33   prov:agent wikiprov:Mav ;
```

```

34     evident:normalInfluenceFactor "0.012578616352201255"^^xsd:double .
36
37   wiki prov:United_States_National_Forest_559734
38     wiki prov:comment ""^^xsd:string ;
39     wiki prov:id "559734"^^xsd:string ;
40     wiki prov:pageid "42653"^^xsd:string ;
41     wiki prov:parentid "223449"^^xsd:string ;
42     wiki prov:revid "559734"^^xsd:string ;
43     wiki prov:size "1116"^^xsd:string ;
44     wiki prov:time "2002-09-17T03:11:31Z"^^xsd:string ;
45     wiki prov:title "United_States_National_Forest"^^xsd:string ;
46     a wiki prov:article, prov:Entity ;
47     prov:qualifiedAttribution wiki prov:Mav_559734 ;
48     prov:qualifiedGeneration wiki prov:559734generation ;
49     prov:qualifiedRevision wiki prov:United_States_National_Forest_559734_223449
50     .
51   wiki prov:United_States_National_Forest_559734_223449
52     wiki prov:agent "Mav"^^xsd:string ;
53     wiki prov:changed 2 ;
54     wiki prov:common 157 ;
55     wiki prov:deleted 0 ;
56     wiki prov:entity1 "223449"^^xsd:string ;
57     wiki prov:entity2 "559734"^^xsd:string ;
58     wiki prov:id "United_States_National_Forest_559734_223449"^^xsd:string ;
59     wiki prov:parentid "223449"^^xsd:string ;
60     wiki prov:relationshipName "United_States_National_Forest_559734_223449"^^xsd:
61       :string ;
62     wiki prov:revid "559734"^^xsd:string ;
63     wiki prov:words 159 ;
64     a prov:Revision ;
65     prov:entity wiki prov:United_States_National_Forest_223449 ;
66     prov:hadActivity wiki prov:activity_559734 ;
67     prov:hadGeneration wiki prov:559734generation ;
68     evident:normalInfluenceFactor "0.9874213836477987"^^xsd:double .
69
70   wiki prov:activity_559734
71     wiki prov:comment ""^^xsd:string ;
72     wiki prov:endtime "2002-09-17T03:11:31Z"^^xsd:string ;
73     wiki prov:id "comment559734"^^xsd:string ;
74     wiki prov:revid "559734"^^xsd:string ;
75     wiki prov:starttime "null"^^xsd:string ;
76     a prov:Activity, "edit"^^xsd:string ;
77     prov:endedAtTime "2002-09-17T03:11:31Z"^^xsd:dateTime ;
78     prov:qualifiedAssociation wiki prov:comment559734Mav .
79
80   wiki prov:comment559734Mav
81     wiki prov:activity "comment559734"^^xsd:string ;
82     wiki prov:agent "Mav"^^xsd:string ;
83     wiki prov:id "comment559734Mav"^^xsd:string ;
84     wiki prov:publicationpolicy "null"^^xsd:string ;
85     wiki prov:relationshipName "comment559734Mav"^^xsd:string ;
86     wiki prov:user_name "Mav"^^xsd:string ;
87     a prov:Association ;

```

86 `prov:agent wikipro:Mav .`