

OMNISPECTIVE ANALYSIS AND REASONING

AN EPISTEMIC APPROACH TO SCIENTIFIC WORKFLOWS

SRINIVAS CHEMBOLI



Australian  
National  
University

A thesis submitted for the degree of  
Doctor of Philosophy  
of the  
Australian National University

September 2012



This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/3.0/>.

Copyright © 2012 Srinivas Chemboli  
version 12.09.04

## DECLARATION

I declare that the work in this thesis is entirely my own and that to the best of my knowledge it does not contain any materials previously published or written by another person except where otherwise indicated.

Srinivas Chemboli

*04 September 2012*



[S]cience includes any approach that is open to reason, to rational discussion, investigation, skepticism, to critical thinking, to questioning. . . I wouldn't say you have to put on a white coat and go into a laboratory in order to [pursue science].

Dawkins [2012]



## ACKNOWLEDGMENTS

The research presented in this dissertation could not have been done without the help of many colleagues and friends who supported me in many ways during all the years. Only the names of some of them can be listed here.

First and foremost, I would like to thank my supervisor, Clive Boughton, for his unstinting support and encouragement. Thank you, Clive, for skillfully ensuring the right mix of ‘supervision’ and ‘independent latitude’ in research. I have learnt much these past years, and that could not have happened without your guidance.

I have benefited greatly from the valuable feedback and suggestions of my advisors, Shayne Flint and Ramesh Sankaranarayana. Thank you Shayne, for the informed critique and many hours of brainstorming and discussion. I am grateful to Ramesh for steering me towards ‘scientific workflows.’

I am also thankful to Elisa Baniassad, chair of my supervisory panel — although Elisa came on board at a later stage of my study, her advice and insights have been very helpful in shaping the research.

Over the past several years, I have had the great pleasure of interacting with an incredible group of people in the School of Computer Science.

In particular, I would like to thank Henry Gardner for taking an active interest in my research and offering valuable advice.

I am also thankful to Lynette Johns–Boast, Chris Johnson, Clem Baker–Finch and Alexei Khorev for giving me the opportunity to learn to teach. The skills acquired from this experience have been very useful in my research. I would like to thank Tom Worthington, Malcolm Newey and Jay Larson for numerous helpful and enlightening discussions about conducting and presenting research.

To Julie Arnold, Chelsea Holton, Fiona Quinlan, Bethany Flanders, Suzy Andrew, Debbie Pioch, Jill Mayo, Marie Katselas, Sue van Haeften, Jonathan Peters, Paul Melloy and Jadon Radcliffe: A Big Thank You for cheerfully handling all administrative niggles. Special thanks to Bob Edwards, Hugh Fisher, Steven Hanley and in particular to James Fellows, for technical support and wizardry.

During my PhD, I also had the opportunity to work as an educational technologist in the Division of Information at the ANU. I would like to thank Karen Visser, Jenny Edwards, Marina Lobastov, Peter Yates, Jaymie Parker, Sam Primrose, Grazia Scotellaro, Candida Spence, Hans Joerg–Kraus, Kim Blackmore and Lauren Thompson — it was a privilege to work with you.

In addition, I also participated in the Pinnacle Teaching Program, the Graduate Teaching Program and the Foundations of University Teaching and Learning course. I am thankful to Trevor Vickers and Beth Beckmann for their support in these programs.

It would be remiss of me not to express my thanks to Mark Drechsler for insights of Moodle and learning management systems.

I would also like to thank Deborah Veness for her comments and suggestions on the application of OAR to course design and for encouraging me to present the work at ascilite.

I am thankful to Tony Karrys, Gina Denman, Kaori Oikawa and Walter Sauer for their assistance and kindness.

This research has been supported by the Australian National University, and the Commonwealth of Australia, through the Cooperative Research Centre for Advanced Automotive Technology. I would like to thank Kate Neely for her support and assistance in coordinating with the AutoCRC.

I would like to thank the many people with whom I have shared my time at CECS: Agung Fatwanto, Normi Abu Bakar, Zoe Brain, Luke Nguyen–Hoan, Alvin Teh, Amir Hadad, Sukanya Manna, Josh Milthorpe, Jaison Mulerikkal, Derek Wang and Ian Wood. You have been excellent company. I must particularly thank my officemate and friend Ziyad Alshaikh. Thank you for the sparkling wit and interesting discussions — no research ideas were too outlandish for conversation!

Special thanks to:

Aditi Barthwal and Kate Sullivan, for your friendship, support and encouragement.

Josephine Wright, for your optimism and friendly advice.

Ganesh Venkateswara, for taking time out and providing thoughtful insights and feedback.

Finally, I would like to thank my family for their continuous support and encouragement. Without you this work would never have been possible.



## ACKNOWLEDGMENTS OF PERMISSIONS

I want to acknowledge the permissions of various copyright holders.

The Workflow Management Coalition kindly permitted me to use the following material:

From D. Hollingsworth, "The Workflow Reference Model", Document Number TC00-1003, The Workflow Management Coalition, Hampshire, UK : 1995 <http://www.wfmc.org/standards/docs/tc003v11.pdf>,

Figure 6 (Figure 2.1 in the thesis, redrawn for style and appearance)

Dr Shayne Flint of the Australian National University kindly permitted me to use the following material:

From S. Flint, "Rethinking Systems Thinking", Proceedings of the 14th ANZSYS Australia New Zealand Systems Society Conference, Perth, Australia : 2008 <http://www.anzsys.org/anzsys08/papers/Shayne%20Flint%20Rethinking%20Systems%20Thinking.pdf>

Figure 2 (Figure 3.1 in the thesis, redrawn for style and appearance)

The International Society for the Systems Sciences kindly permitted me to use the following material:

From J.N. Warfield, "The Domain of Science Model: Evolution and Design", Proceedings of the Society for General Systems Research, Salinas, CA: Intersystems, 1986, H46-H59,

Figure 1 (Figure 3.2 in the thesis, redrawn for style and appearance)

Lynette Johns-Boast of the Australian National University kindly permitted me to use the following material:

"A Model of Curriculum Design", Personal communication : 2010,

(Figure 5.1 in the thesis, redrawn for style and appearance)



## ABSTRACT

This thesis presents the conceptualization, formulation, development and demonstration of Omnispersive Analysis and Reasoning (OAR), an epistemic framework for managing intellectual concerns in scientific workflows.

Although scientific workflows are extensively used to support the management of experimental and computational research, intellectual concerns are not adequately handled in current practice owing to the focus on low-level implementation details, limited context support, issues in developing shared semantics across disciplines and lack of support for verification and validation of the underlying science of the workflow. The management of intellectual concerns in scientific workflows can be improved by developing a framework for providing a layer of abstraction to lift focus from low-level implementation details, adding context as a workflow parameter, introducing localized ontologies and abstracting and mapping intellectual concerns in the research-domain to workflow specification and execution semantics.

Following an examination of typical definitions of scientific workflow offered in literature, the Scientific Method is applied to develop an enhanced definition of a scientific workflow. This definition, which extends the scope of ordered analysis and investigation to a generic problem scenario, is utilized in the OAR framework. The design of OAR is modular like the Domain of Science Model (DoSM). The structure and working of OAR incorporate the evolving nature of science, hierarchy of conceptualization, omnisppection, and the logical processes of analysis, reasoning and abstraction. These form the Foundation and Theory of OAR. Abstracting concerns in terms of unit knowledge entities (ukes) and groups of ukes (recipes), use of context to identify relation between recipes, the management of recipes in shelves, and the processes of concern refinement and context refinement constitute the Methodology.

A comprehensive and simple example of the application of OAR to the abstraction, analysis, formulation and orchestration of a scientific workflow at different levels of granularity is provided by applying it to the problem of origami paper folding. The use of OAR in capturing the rationale of design decisions and mapping them to desired outcomes is demonstrated by applying OAR for contextualizing course design. Another example illustrates the use of OAR in the analysis, understanding and management of complex systems. Localized ontologies enable the exposure of side-effects and emergent behavior in

large-scale systems due to the choice of any particular solution specification. These examples constitute a first step in building the Applications block of OAR. While OAR may be manually applied even to large-scale problems, it is expedient to avail of tool support. Soma — a simple and illustrative tool prototype is developed to indicate directions for a reference tool implementation.

The thesis concludes with a consideration of ideas for future work. The contribution in this thesis corresponds to an instance of the DoSM for scientific workflow management. The OAR framework has great potential for further development as a well-formed Science of Workflows.

## PUBLICATIONS

Some ideas and figures have appeared previously in the following presentations and publications:

Chemboli, S. and Boughton, C. [2012a]. “Managing Large and Complex Systems with Omnispective Analysis and Reasoning.” In: *Proceedings of SETE APCOSE 2012*. Brisbane, Australia. URL: <http://hdl.handle.net/1885/9009>

Chemboli, S. and Boughton, C. [2012b]. “Omnispective Analysis and Reasoning: A Framework for Managing Intellectual Concerns in Scientific Workflows.” In: *Proceedings of the 5th India Software Engineering Conference*. Kanpur, India, pp. 143–146. DOI: [10.1145/2134254.2134279](https://doi.org/10.1145/2134254.2134279)

Chemboli, S. and Boughton, C. [2011]. “Contextual Course Design with Omnispective Analysis and Reasoning.” In: *Changing Demands, Changing Directions. Proceedings ascilite*. Ed. by Williams, G. et al. Hobart, pp. 210–219. URL: <http://www.leishman-associates.com.au/ascilite2011/downloads/papers/Chemboli-full.pdf>

Chemboli, S. [Oct. 2010a]. *Contextualizing learning outcomes and course design in Moodle*. Presented at Moodleposium AU 2010. Canberra, Australia. URL: <http://hdl.handle.net/1885/9279>

Chemboli, S., Kane, L., and Johns-Boast, L. [July 2010]. *Translating Learning Outcomes in Moodle*. Presented at Moodle moot AU 2010. Melbourne, Victoria, Australia. URL: <http://ubuntuone.com/4RBjlozHyEyITuy21aDCfe>

Chemboli, S. [Feb. 2010b]. *Omnispective Analysis and Reasoning: An epistemic approach to scientific workflows*. Presented at the CECS Seminar Series, Australian National University. Canberra, Australia. URL: <http://cecs.anu.edu.au/seminars/more/SID/2503>



# CONTENTS

ACKNOWLEDGMENTS    [vii](#)

ABSTRACT    [xi](#)

PUBLICATIONS    [xiii](#)

CONTENTS    [xv](#)

LIST OF FIGURES    [xix](#)

LIST OF TABLES    [xxi](#)

## I Introduction 1

### 1 OVERVIEW 3

- 1.1 Initial Motivation and Research Aim 4
- 1.2 Research Design 4
- 1.3 Thesis Scope 5
- 1.4 Thesis Structure 5
  - 1.4.1 [Part I: Introduction](#) 5
  - 1.4.2 [Part II: Omnispersive Analysis and Reasoning](#) 7
  - 1.4.3 [Part III: Proof of Concept](#) 7
  - 1.4.4 [Part IV: Conclusion](#) 7
- 1.5 Main Contributions 7

### 2 BACKGROUND 9

- 2.1 Workflows 10
  - 2.1.1 Scientific Database Systems 10
  - 2.1.2 Unstructured Activities 11
  - 2.1.3 Dynamic Process Composition 13
  - 2.1.4 Distributed and Decentralized Processes 13
  - 2.1.5 The Workflow Reference Model 15
  - 2.1.6 Participatory Analysis 17
  - 2.1.7 Emergence of Scientific Workflows 18
  - 2.1.8 Scientific Workflows 20
- 2.2 Issues in Scientific Workflow Management 23

2.2.1	Focus on Low-level Detail	23
2.2.2	Limited Context Support	26
2.2.3	Inadequate Management of Intellectual Concerns	28
2.3	Impact of the Above Issues	29
2.3.1	Observation 1	30
2.3.2	Observation 2	30
2.3.3	Observation 3	30
2.4	Conjecture	31
2.5	Summary and Conclusion	32

## II Omnispersive Analysis and Reasoning 33

3	OMNISPECTIVE ANALYSIS AND REASONING	35
3.1	The Nature of Science	36
3.2	Fixation of Belief	38
3.3	Universal Priors to Science	39
3.4	Law of Triadic Compatibility	40
3.5	Hierarchy of Conceptualization	41
3.6	Domain of Science Model	42
3.7	Expanding Scale and Complexity of Scientific Work	44
3.8	Intellectual Concerns in Scientific Work	44
3.9	Defining Scientific Workflows	45
3.10	'Reforming' Scientific Workflow Management	48
3.10.1	Managing Intellectual Concerns	49
3.10.2	Dealing with Inadequate Context Support	51
3.10.3	Inadequate Support for Verification and Validation	51
3.11	Theoretical Foundations of OAR	52
3.11.1	Omnispersive Analysis	53
3.11.2	Lifting Focus from Low-level Details	54
3.11.3	Defining Context and Adding Context Support	56
3.11.4	Localized Ontologies	58
3.11.5	Epistemological Basis	60
3.12	Overview of the OAR Framework	60
3.13	Prototypes, Archetypes and Constraints	62
3.14	Concern Refinement	63
3.15	'Bootstrapping' External Shelves	65
3.16	Context Refinement	66
3.17	Constructed and Organic Solution Specifications	70
3.18	Rationale for the Structure of OAR	71



- 3.19 Nature of the OAR Framework 73
- 3.20 Summary and Conclusion 74

### III Proof of Concept

77

- 4 ORIGAMI FOLDING WORKFLOW 81
  - 4.1 Paper Folding as a Scientific Workflow 82
  - 4.2 Folding the Iris Flower 82
  - 4.3 Applying OAR to the Iris Flower Workflow 85
    - 4.3.1 Identifying Relevant Archetypes and Constraints 86
    - 4.3.2 Formulating the Solution Specification 86
  - 4.4 Implementing the Solution Specification 87
  - 4.5 Summary and Conclusion 88
- 5 CONTEXTUALIZING COURSE DESIGN 91
  - 5.1 Learning, Teaching and Course Design 92
    - 5.1.1 Learning Outcomes 94
    - 5.1.2 Translating Learning Outcomes to Course Design 95
  - 5.2 Contextualizing Course Design 96
  - 5.3 Translating Learning Outcomes for COMP8120 97
    - 5.3.1 Learning Outcomes for COMP8120 97
    - 5.3.2 Analyzing Context for LO-1 and LO-2 98
    - 5.3.3 Analyzing Context for LO-3 98
    - 5.3.4 Analyzing Context for LO-4 99
    - 5.3.5 Analyzing Context for LO-5 99
  - 5.4 Solution Specification for LO-5 99
    - 5.4.1 Initializing External Shelves 99
    - 5.4.2 Identifying Relevant Archetypes and Constraints 100
    - 5.4.3 Solution Shelf for LO-5 101
    - 5.4.4 Implementing the Solution Specification 102
  - 5.5 Summary and Conclusion 102
- 6 MANAGING LARGE AND COMPLEX SYSTEMS 103
  - 6.1 Large and Complex Systems 104
  - 6.2 Some Characteristics of Large Systems 105
  - 6.3 How complexity builds and escalates in large systems 106
  - 6.4 Applying OAR to Complex Systems 107
  - 6.5 The Ubuntu Platform as a Complex System-of-Systems 109
  - 6.6 Capturing intellectual concerns for the Ubuntu ecosystem 109
    - 6.6.1 Initializing External Shelves 110

- 6.7 Identifying Relevant Archetypes and Constraints 111
- 6.8 Solution Specification for Selecting the Default Music App 111
- 6.9 Utilizing a Solution Specification 113
- 6.10 Summary and Conclusion 117

## IV Conclusion 119

- 7 SOMA: OAR TOOL PROTOTYPE 121
  - 7.1 Soma: A Tool for Simple Omnispersive Analysis and Reasoning 122
    - 7.1.1 Initialization 122
    - 7.1.2 Building the Problem–domain Shelf 123
    - 7.1.3 Contextualization 124
    - 7.1.4 Practical Considerations in Soma 124
  - 7.2 Product Vision and Goal 126
  - 7.3 Architecture Vision and Sprint Planning 126
    - 7.3.1 Architecture Vision 127
  - 7.4 Soma Development 128
    - 7.4.1 Soma Sprint 1 128
    - 7.4.2 Soma Sprint 2 135
  - 7.5 Summary and Conclusion 137
- 8 SUMMARY AND CONCLUSIONS 141
  - 8.1 Summary of Contribution 142
    - 8.1.1 An Enhanced Definition of Scientific Workflow 142
    - 8.1.2 Omnispersive Analysis and Reasoning 142
  - 8.2 Limitations of Contribution 145
  - 8.3 Related Work 147
  - 8.4 Viewing Enterprise Architecture through OAR 148
    - 8.4.1 Enterprise Architecture and Architecture Frameworks 149
    - 8.4.2 Applying OAR — Architecture Views as Workflows 152
  - 8.5 Directions for Future Work 155
    - 8.5.1 Managing Fractional Values for Firmness and Influence in Recipe Context 155
    - 8.5.2 Tool Support 155
    - 8.5.3 Moodle OAR Plugin 156
    - 8.5.4 Application to Complex Systems 156
    - 8.5.5 OAR as Science of Workflows 156

## BIBLIOGRAPHY 157

## LIST OF FIGURES

Figure 1.1	Activity diagram depicting the structure and flow of ideas and results throughout the thesis.	6
Figure 2.1	Components and interfaces in the workflow reference model (after Hollingsworth [1995]). Copyright ©1995 The Workflow Management Coalition. Adapted with permission.	16
Figure 2.2	Delineating high-level concerns and low-level details in an experiment.	25
Figure 2.3	Illustrating context.	28
Figure 3.1	An adaptation of Boyd's OODA loop for managing complex problem situations (after [Flint, 2008]). Copyright ©2008 Flint, S. Adapted with permission.	37
Figure 3.2	The Domain of Science Model (after [Warfield, 1994]). Copyright ©1986 International Society for the Systems Sciences. Adapted with permission.	43
Figure 3.3	A definition of the OAR framework.	52
Figure 3.4	Composite nature of Unit knowledge entity (uke).	55
Figure 3.5	Hierarchy of concerns	56
Figure 3.6	Uke context as a function of Firmness and Influence.	57
Figure 3.7	Relation between concern, recipe and uke.	62
Figure 3.8	Shelves and recipes.	64
Figure 3.9	Managing concerns in the OAR framework.	65
Figure 3.10	Visualization of shelf bootstrap.	67
Figure 3.11	Representing context by profiles of the attributes Influence and Firmness.	68
Figure 3.12	The process of context refinement.	69
Figure 4.1	Workflow for folding the Iris flower starting with a Frog Base.	83
Figure 4.2	Illustrating the implementation-level focus of current scientific workflow practice.	84
Figure 4.3	External shelves and prototypes.	85
Figure 4.4	Selected prototypes in the Problem-domain shelf for the Iris Flower specification.	86
Figure 4.5	Origami Iris Flower specification.	87

- Figure 4.6 Formulating a solution specification for implementing the Iris Flower solution shelf of Figure 4.5. 89
- Figure 5.1 A model of curriculum design. Copyright ©2010 Johns-Boast, L. Adapted with permission. 93
- Figure 5.2 External shelves and prototypes. 100
- Figure 5.3 Archetype and constraint identification for LO-5. 101
- Figure 5.4 Solution specification for LO-5. 101
- Figure 6.1 Escalation of problem complexity when additional inputs are provided. 106
- Figure 6.2 External shelves and recipes for deciding the default music app. 110
- Figure 6.3 Archetype and constraint identification in the problem domain shelf. 111
- Figure 6.4 Solution shelf for the default music app specification. 113
- Figure 6.5 Local ontology for default apps. 115
- Figure 6.6 The problem-domain shelf for selecting the default note app. 116
- Figure 6.7 In this recipe, the Ubuntu 12.04 LTS distribution does not ship with a default note app. 116
- Figure 6.8 In this recipe, the Ubuntu 12.04 LTS distribution can ship with GNote as the default note app even though it lacks critical synchronization features. 117
- Figure 7.1 The initial state of Soma with several external shelves and an empty problem-domain shelf canvas. 123
- Figure 7.2 The newly initialized PDS is populated 124
- Figure 7.3 Possible ambiguity due to visual simplification of constraint representation in a solution shelf. 125
- Figure 7.4 The requirements pyramid for the Soma OAR tool prototype. 127
- Figure 7.5 Sliced view of the Soma architecture cluster. 127
- Figure 7.6 Data architecture model of Soma. Storage of OAR data is independent of its representation in the Soma Logical Layer. The Soma Visual Layer can be customized to generate different outputs. 128
- Figure 7.7 Soma data ring during Sprint 1. 129
- Figure 7.8 Soma data model. 130
- Figure 7.9 Opening screen of Soma. 131
- Figure 7.10 Creating a new recipe. 132

Figure 7.11	Filling in the details for a new recipe. The recipe UUID is auto-generated.	132
Figure 7.12	A recipe for the Flat technique.	133
Figure 7.13	A recipe for Valley fold.	133
Figure 7.14	Entering the details of a new specification.	134
Figure 7.15	Recipes in the Problem-domain Shelf view.	134
Figure 7.16	Assigning recipe state.	135
Figure 7.17	Constraints for the origami iris flower specification. The <i>Frog Base</i> , <i>Mountain Fold</i> and <i>Flap</i> recipes are not shown in this figure.	136
Figure 7.18	Selecting recipes for context refinement.	136
Figure 7.19	Assigning recipe influence.	137
Figure 7.20	Saving the solution specification.	138
Figure 8.1	OAR process for generating architecture views.	154

## LIST OF TABLES

Table 3.1	Ideas in the OAR framework.	72
Table 3.2	Some OAR translation engines.	73
Table 7.1	Soma release and sprint planning.	129