

# Contents

<b>1</b>	<b>Introduction .....</b>	<b>1</b>
	Uwe Aßmann, Jürgen Ebert, Jeff Z. Pan, Steffen Staab, and Yuting Zhao	
1.1	Vision: From Type-Safe to Consistency-Based and Guided Software Development .....	2
1.2	This Book: From Model-Driven to ODSD .....	4
1.3	Who This Book Is for .....	11
1.3.1	I Know About Software Development, What Can I Learn from This Book? .....	11
1.3.2	I Know About Ontology, What Can I Learn from This Book? .....	12
1.3.3	The Website of This Book .....	12
1.4	How to Read This Book .....	12
1.4.1	Part I: Ontology and Software Technology .....	13
1.4.2	Part II: Foundational Technologies for ODSD .....	14
1.4.3	Part III: Consistency Checking in ODSD .....	14
1.4.4	Part IV: ODSD with Process Guidance .....	14

**Part I   Ontology and Software Technologies**

<b>2</b>	<b>Model-Driven Software Development .....</b>	<b>21</b>
	Fernando Silva Parreiras, Gerd Gröner, Tobias Walter, Andreas Friesen, Tirdad Rahmani, Jens Lemcke, Hannes Schwarz, Krzysztof Miksa, Christian Wende, and Uwe Aßmann	
2.1	Introduction of MDE .....	21
2.1.1	Models .....	22
2.1.2	Metamodels .....	22
2.1.3	Modelling Languages .....	22

2.2	MDE Languages.....	23
2.2.1	UML Class Diagram .....	23
2.2.2	Metamodelling Language .....	24
2.2.3	Process Modelling Languages .....	26
2.2.4	Domain-Specific Language .....	28
2.2.5	Graph-Based Languages .....	29
2.2.6	Feature Modelling Languages .....	33
2.3	Two-Dimensional Modelling .....	36
2.4	Model Transformation Languages.....	37
2.4.1	Atlas Transformation Language .....	38
2.4.2	Graph Repository Transformation Language <sup>(*)</sup> .....	39
2.5	Constraint and Query Languages.....	43
2.5.1	The Object Constraint Language .....	43
2.5.2	Graph Repository Query Language <sup>(*)</sup> .....	44
2.5.3	Process Algebra.....	48
2.6	Conclusion .....	50
<b>3</b>	<b>Ontology Languages and Description Logics.....</b>	<b>51</b>
	Yuting Zhao, Jeff Z. Pan, Edward Thomas, Nophadol Jekjantuk, and Yuan Ren	
3.1	Description Logics .....	51
3.1.1	The Basic Description Language $\mathcal{AL}$ .....	52
3.1.2	The Family of $\mathcal{AL}$ Languages .....	53
3.1.3	Reasoning Tasks in Description Logics .....	54
3.1.4	Computational Properties for Description Logics.....	55
3.2	The OWL Family of Ontology Languages and Related Others .....	57
3.2.1	OWL 2 Web Ontology Language (OWL 2) .....	57
3.2.2	OWL 2 Profiles .....	59
3.2.3	$\mathcal{EL}^{++}$ .....	60
3.2.4	DL-Lite.....	61
3.2.5	OWL DL .....	64
3.3	Conclusion .....	67
<b>4</b>	<b>Case Studies for Marrying Ontology and Software Technologies.....</b>	<b>69</b>
	Krzysztof Miksa, Pawel Sabina, Andreas Friesen, Tirdad Rahmani, Jens Lemcke, Christian Wende, Srdjan Zivkovic, Uwe Aßmann, and Andreas Bartho	
4.1	Case Studies on Domain Engineering .....	69
4.1.1	Problem Description.....	70
4.1.2	Implementation .....	72
4.1.3	Results .....	74
4.2	Case Studies on Process Modelling and Refinement .....	74
4.2.1	Introduction .....	75
4.2.2	Case Study: Process Refinement.....	78

4.2.3	Case Study: Querying Tasks and Artefacts .....	87
4.2.4	Process Refinement Validation: Implementation and Results .....	90
4.2.5	Querying Tasks and Artefacts: Implementation and Results .....	90
4.3	Case Study on Software Product Line Engineering .....	91
4.3.1	Problem Description .....	91
4.3.2	Case Study Implementation .....	92
4.3.3	Results .....	93
4.4	Conclusion .....	94

## **Part II Foundational Technologies for Ontology-Driven Software Development**

<b>5</b>	<b>Scalable Ontology Reasoning Services .....</b>	<b>99</b>
	Edward Thomas, Yuan Ren, Jeff Z. Pan, Yuting Zhao, and Nophadol Jekjantuk	
5.1	TrOWL: Tractable Reasoning Infrastructure for OWL 2 .....	99
5.1.1	Applications .....	100
5.1.2	Technology .....	101
5.1.3	Query Answering Using Semantic Approximation <sup>(*)</sup> ....	103
5.1.4	Scalable Reasoning Using Syntactic Approximation <sup>(*)</sup> .....	107
5.1.5	Local Closed World Reasoning with NBox <sup>(*)</sup> .....	110
5.2	Justification of Reasoning Results .....	113
5.2.1	Justification in General .....	113
5.2.2	Justification on OWL 2 EL <sup>(*)</sup> .....	116
5.2.3	Justification on OWL 2 QL <sup>(*)</sup> .....	116
5.3	Conclusion .....	120
<b>6</b>	<b>Traceability .....</b>	<b>121</b>
	Hannes Schwarz and Jürgen Ebert	
6.1	Fields of Application .....	121
6.1.1	Change Management .....	122
6.1.2	Maintenance .....	122
6.1.3	Project Management .....	122
6.1.4	Quality Assurance .....	122
6.1.5	Requirements Management .....	123
6.1.6	Reuse .....	123
6.1.7	Reverse Engineering .....	123
6.2	Traceability-Related Activities .....	124
6.2.1	Definition .....	124
6.2.2	Identification .....	125
6.2.3	Recording .....	125
6.2.4	Retrieval .....	126

6.2.5	Utilisation .....	126
6.2.6	Maintenance .....	127
6.3	An Universal Traceability Concept.....	127
6.3.1	The Traceability Relationship Type Template.....	128
6.3.2	Traceability Retrieval Patterns .....	130
6.4	Implementation Based on Ecore <sup>(*)</sup> .....	132
6.4.1	Mapping of TRTT Properties .....	132
6.4.2	Transformation-Based Identification .....	135
6.4.3	Technologies for Recording .....	135
6.4.4	Querying Conforming to the Retrieval Patterns.....	136
6.5	Implementation Based on the TGraph Approach <sup>(*)</sup> .....	137
6.5.1	Mapping of TRTT Properties .....	138
6.5.2	Transformation-Based Identification .....	140
6.5.3	Technologies for Recording .....	141
6.5.4	Querying Conforming to the Retrieval Patterns.....	141
6.6	Implementation Based on OWL <sup>(*)</sup> .....	143
6.6.1	Mapping of TRTT Properties .....	143
6.6.2	Technologies for Recording .....	147
6.6.3	Querying Conforming to the Retrieval Patterns.....	147
6.7	Discussions .....	149
6.8	Conclusion.....	150
<b>7</b>	<b>Metamodelling and Ontologies<sup>(*)</sup> .....</b>	<b>151</b>
	Gerd Gröner, Nophadol Jekjantuk, Tobias Walter, Fernando Silva Parreiras, and Jeff Z. Pan	
7.1	Metamodelling for Ontologies.....	151
7.1.1	Ontology Definition Metamodel .....	152
7.1.2	NeOn OWL Metamodel .....	154
7.1.3	OWL 2 Metamodel .....	156
7.2	Ontologies for Metamodelling: OWL FA .....	157
7.2.1	Motivating Example .....	157
7.2.2	OWL FA Syntax and Semantics .....	159
7.2.3	Metamodelling with OWL FA .....	162
7.2.4	Reasoning in OWL FA .....	163
7.2.5	Preprocessing .....	163
7.2.6	Consistency Checking.....	165
7.2.7	Instance Retrieval .....	165
7.2.8	Justification on OWL FA .....	166
7.3	Metamodelling in Ontologies and Metamodelling in MOF .....	167
7.3.1	Open and Closed World Assumptions .....	168
7.3.2	Ensuring Integrity Constraints in a Closed Domain .....	170
7.4	Conclusion .....	174

## **Part III Consistency Checking in Ontology-Driven Software Development (ODSD)**

<b>8</b>	<b>Ontology and Bridging Technologies</b> .....	179
	Uwe Aßmann, Jürgen Ebert, Tobias Walter, and Christian Wende	
8.1	Ontology Services .....	179
8.1.1	Pattern for Specification of Ontology Services .....	180
8.1.2	Specification of Ontology Services .....	180
8.2	Transformation Bridges <sup>(*)</sup> .....	183
8.2.1	Transformation Services .....	183
8.2.2	Example: OWLizer .....	184
8.3	Integration Bridge <sup>(*)</sup> .....	185
8.3.1	Integration Bridge Establishment .....	186
8.3.2	Integration Bridge Use .....	188
8.3.3	Example: M3 Integration Bridge .....	190
8.4	Conclusion .....	192
<b>9</b>	<b>Ontology Reasoning for Consistency-Preserving Structural Modelling</b> .....	193
	Christian Wende, Katja Siegemund, Edward Thomas, Yuting Zhao, Jeff Z. Pan, Fernando Silva Parreiras, Tobias Walter, Krzysztof Miksa, Pawel Sabina, and Uwe Aßmann	
9.1	Reasoning for Requirement Engineering .....	194
9.1.1	The ODRE Approach .....	194
9.1.2	Rules for Consistency Checking .....	194
9.1.3	Rules for Completeness Checking .....	197
9.1.4	Verification of the Requirement Specification Base .....	198
9.1.5	Results .....	199
9.2	Reasoning for Feature Model Analysis .....	199
9.2.1	Ontology-Based Formalisation of Feature Models .....	199
9.2.2	Application of Ontology Services for Feature Analysis .....	201
9.2.3	Results .....	203
9.3	Reasoning for Domain Engineering .....	204
9.3.1	Physical Devices Ontology .....	204
9.3.2	Integrating PDDSL and OWL 2 .....	207
9.3.3	Metamodel Integration .....	207
9.3.4	Reasoning with Integrated PDDSL and OWL 2 Models .....	211
9.3.5	Results .....	214
9.4	Reasoning for UML Class Diagrams .....	214
9.4.1	TwoUse .....	214
9.4.2	Results .....	217
9.5	Conclusion .....	217

<b>10</b>	<b>Ontology Reasoning for Process Models</b> .....	219
	Yuan Ren, Gerd Gröner, Tirdad Rahmani, Jens Lemcke, Andreas Friesen, Srdjan Zivkovic, Yuting Zhao, and Jeff Z. Pan	
10.1	A Running Example .....	220
10.2	Reasoning for Process Modelling and Retrieval .....	221
10.2.1	Semantic Representation of Process Models .....	221
10.2.2	Reasoning for Process Retrieval .....	223
10.3	Reasoning for Process Refinement in BPMN <sup>(*)</sup> .....	224
10.3.1	Defining Valid Process Refinement .....	225
10.3.2	Constraints on Refinement Relations .....	229
10.3.3	Validating Parallel-Free Process Refinement with Ontology .....	232
10.3.4	Extending Specific Process with Parallel Gateways .....	234
10.3.5	Extending Abstract Process with Parallel Gateways .....	235
10.3.6	Validating Grounding with Ontology .....	237
10.3.7	Pinpointing and Explaining Sources of Invalidity with Ontology Reasoning .....	240
10.3.8	Efficient Refinement Validation with Approximate Ontology Reasoning .....	242
10.4	Reasoning for Process Guidance .....	244
10.4.1	Concepts and Knowledge Assets of a Modelling Scenario .....	244
10.4.2	Formalising Guidance Knowledge into Ontologies .....	246
10.4.3	Retrieving Tasks by Query Answering .....	248
10.4.4	Generalised Solution for Representation and Retrieval ...	250
10.4.5	Computational Properties .....	250
10.5	Conclusion .....	251

## **Part IV    Ontology-Driven Software Development (ODSD) with Process Guidance**

<b>11</b>	<b>Ontology-Driven Metamodelling for Ontology-Integrated Modelling<sup>(*)</sup></b> .....	257
	Uwe Aßmann, Jürgen Ebert, Tobias Walter, and Christian Wende	
11.1	Reasoning for Language Engineering and Use .....	257
11.1.1	Defining an M3 Integration Bridge .....	258
11.1.2	Using an M3 Integration Bridge .....	261
11.2	OWLText .....	265
11.2.1	ODMM with OWLText .....	265
11.2.2	Realisation and Application of the M3 Bridges and M2 Bridges in OWLText .....	267
11.2.3	An Exemplary Application of OWLText .....	270
11.3	Conclusion .....	273

<b>12 A Platform for ODSD: The MOST Workbench</b>	275
Srdjan Zivkovic, Christian Wende, Edward Thomas, Fernando Silva Parreiras, Tobias Walter, Krzysztof Miksa, Harald Kühn, Hannes Schwarz, and Jeff Z. Pan	
12.1 Towards ODSD Tool Environments	275
12.1.1 Current Tool Environments	276
12.1.2 ODSD Tool Environments	278
12.1.3 Developing a Family of ODSD Tool Environments	280
12.2 Configurable and Tractable Ontology Reasoning	
Infrastructure for ODSD	283
12.2.1 Closed-World Reasoning in NBox	283
12.2.2 Justification and Explanation	284
12.2.3 Tractability	284
12.2.4 Configurability	284
12.3 Integrated Queries for ODSD	285
12.4 Validation and Explanation in ODSD	287
12.4.1 Validation and Explanations for Process Refinements	287
12.4.2 Validation and Explanations for Domain Engineering	288
12.4.3 Model Repair	289
12.5 Traceability in ODSD	290
12.5.1 Using the Integration Infrastructure for Traceability	290
12.5.2 Benefiting from Transformations Between Modelling and Ontology Languages	291
12.6 Conclusion	292
<b>13 Ontology-Guided Software Engineering in the MOST Workbench</b>	293
Uwe Aßmann, Srdjan Zivkovic, Krzysztof Miksa, Katja Siegemund, Andreas Bartho, Tirdad Rahmani, Edward Thomas, and Jeff Z. Pan	
13.1 Ontology-Based Guidance Engine	293
13.1.1 The Generic Guidance	293
13.1.2 The Generic Guidance Ontology	296
13.1.3 Implementation Technologies	298
13.2 Guidance for Process Refinement Engineering	300
13.2.1 Guidance as a Service for Refinement	301
13.2.2 Guidance as a Process for Refinement	302
13.3 Guidance for Domain Engineering	305
13.3.1 Consistency Guidance	305
13.3.2 Process Guidance	306
13.4 Guidance for Requirements Engineering	309
13.4.1 Requirements Guidance Ontology	310
13.4.2 Guidance for Requirement Engineering	312
13.5 Guidance for Documentation Engineering	314
13.6 Conclusion	318

<b>14 Conclusion and Outlooks .....</b>	<b>319</b>
Jeff Z. Pan, Steffen Staab, Uwe Aßmann, Jürgen Ebert, Yuting Zhao, and Daniel Oberle	
14.1 Key questions for Ontology-Driven Software Development (ODSD) .....	320
14.2 Where to Go from Here? .....	320
14.2.1 Things to Keep in Mind When Adopting ODSD .....	321
14.2.2 Contributing to the Linked Software Data .....	321
14.2.3 Further Visions in Software Engineering .....	322
<b>References .....</b>	<b>323</b>
<b>Index .....</b>	<b>335</b>



<http://www.springer.com/978-3-642-31225-0>

Ontology-Driven Software Development

Pan, J.Z.; Staab, S.; Aßmann, U.; Ebert, J.; Zhao, Y.  
(Eds.)

2013, XVIII, 338 p., Hardcover

ISBN: 978-3-642-31225-0