

Contents

Part I Core of the Semantic Web

1	A Web of Data: Toward the Idea of the Semantic Web	3
1.1	A Motivating Example: Data Integration on the Web	4
1.1.1	A Smart Data Integration Agent	4
1.1.2	Is Smart Data Integration Agent Possible?	10
1.1.3	The Idea of the Semantic Web	11
1.2	A More General Goal: A Web Understandable to Machines . . .	12
1.2.1	How Do We Use the Web?	12
1.2.2	What Stops Us From Doing More?	15
1.2.3	Again, the Idea of the Semantic Web	17
1.3	The Semantic Web: A First Look	17
1.3.1	The Concept of the Semantic Web	17
1.3.2	The Semantic Web, Linked Data and the Web of Data . . .	18
1.3.3	Some Basic Things About the Semantic Web	20
	Reference	21
2	The Building Block for the Semantic Web: RDF	23
2.1	RDF Overview	23
2.1.1	RDF In Official Language	23
2.1.2	RDF in Plain English	25
2.2	The Abstract Model of RDF	29
2.2.1	The Big Picture	29
2.2.2	Statement	30
2.2.3	Resource and Its URI Name	31
2.2.4	Predicate and Its URI Name	36
2.2.5	RDF Triples: Knowledge That Machines Can Use	38
2.2.6	RDF Literals and Blank Node	40
2.2.7	A Summary So Far	47

- 2.3 RDF Serialization: RDF/XML Syntax 48
 - 2.3.1 The Big Picture: RDF Vocabulary 48
 - 2.3.2 Basic Syntax and Examples 49
 - 2.3.3 Other RDF Capabilities and Examples 66
- 2.4 Other RDF Sterilization Formats 73
 - 2.4.1 Notation-3, Turtle and N-Triples 73
 - 2.4.2 Turtle Language 73
- 2.5 Fundamental Rules of RDF 81
 - 2.5.1 Information that Is Understandable by Machines 81
 - 2.5.2 Distributed Information Aggregation 84
 - 2.5.3 A Hypothetical Real World Example 85
- 2.6 More About RDF 88
 - 2.6.1 Dublin Core: Example of Predefined RDF Vocabulary 88
 - 2.6.2 XML vs. RDF? 90
 - 2.6.3 Use a RDF Validator 93
- 2.7 Summary 94
- 3 Other RDF-Related Technologies: Microformats, RDFa and GRDDL 97**
 - 3.1 Introduction: Why Do We Need These? 97
 - 3.2 Microformats 98
 - 3.2.1 Microformats: The Big Picture 98
 - 3.2.2 Microformats: Syntax and Examples 99
 - 3.2.3 Microformats and RDF 105
 - 3.3 RDFa 106
 - 3.3.1 RDFa: The Big Picture 106
 - 3.3.2 RDFa Attributes and RDFa Elements 107
 - 3.3.3 RDFa: Rules and Examples 108
 - 3.3.4 RDFa and RDF 115
 - 3.4 GRDDL 116
 - 3.4.1 GRDDL: The Big Picture 116
 - 3.4.2 Using GRDDL with Microformats 117
 - 3.4.3 Using GRDDL with RDFa 118
 - 3.5 Summary 119
- 4 RDFS and Ontology 121**
 - 4.1 RDFS Overview 121
 - 4.1.1 RDFS in Plain English 121
 - 4.1.2 RDFS in Official Language 123
 - 4.2 RDFS + RDF: One More Step Toward Machine-Readable 123
 - 4.2.1 A Common Language to Share 123
 - 4.2.2 Machine Inferencing Based on RDFS 125
 - 4.3 RDFS Core Elements 126
 - 4.3.1 The Big Picture: RDFS Vocabulary 126
 - 4.3.2 Basic Syntax and Examples 127
 - 4.3.3 Summary So Far 146

- 4.4 The Concept of Ontology 150
 - 4.4.1 What Is Ontology 151
 - 4.4.2 The Benefits of Ontology 151
- 4.5 Building the Bridge to Ontology: SKOS 152
 - 4.5.1 Knowledge Organization Systems (KOS) 152
 - 4.5.2 Thesauri vs. Ontologies 154
 - 4.5.3 Filling the Gap: SKOS 156
- 4.6 Another Look at Inferencing Based on RDF Schema 163
 - 4.6.1 RDFS Ontology Based Reasoning: Simple, Yet Powerful 163
 - 4.6.2 Good, Better and Best: More Is Needed 166
- 4.7 Summary 166
- 5 OWL: Web Ontology Language 169**
 - 5.1 OWL Overview 169
 - 5.1.1 OWL in Plain English 169
 - 5.1.2 OWL in Official Language: OWL 1 and OWL 2 170
 - 5.1.3 From OWL 1 to OWL 2 172
 - 5.2 OWL 1 and OWL 2: The Big Picture 173
 - 5.2.1 Basic Notions: Axiom, Entity, Expression and IRI Names 173
 - 5.2.2 Basic Syntax Forms: Functional-Style, RDF/XML Syntax, Manchester Syntax and XML Syntax 174
 - 5.3 OWL 1 Web Ontology Language 175
 - 5.3.1 Defining Classes: The Basics 176
 - 5.3.2 Defining Classes: Localizing Global Properties 178
 - 5.3.3 Defining Classes: Using Set Operators 188
 - 5.3.4 Defining Classes: Using Enumeration, Equivalent and Disjoint 191
 - 5.3.5 Our Camera Ontology So Far 194
 - 5.3.6 Define Properties: The Basics 197
 - 5.3.7 Defining Properties: Property Characteristics 203
 - 5.3.8 Camera Ontology Written Using OWL 1 212
 - 5.4 OWL 2 Web Ontology Language 216
 - 5.4.1 What Is New in OWL 2 217
 - 5.4.2 New Constructs for Common Patterns 217
 - 5.4.3 Improved Expressiveness for Properties 221
 - 5.4.4 Extended Support for Datatypes 232
 - 5.4.5 Punning and Annotations 237
 - 5.4.6 Other OWL 2 Features 241
 - 5.4.7 OWL Constructs in Instance Documents 246
 - 5.4.8 OWL 2 Profiles 250
 - 5.4.9 Our Camera Ontology in OWL 2 256
 - 5.5 Summary 262

- 6 SPARQL: Querying the Semantic Web 265**
 - 6.1 SPARQL Overview 265
 - 6.1.1 SPARQL in Official Language 265
 - 6.1.2 SPARQL in Plain Language 266
 - 6.1.3 RDF Datasets and SPARQL Endpoints 267
 - 6.2 SPARQL 1.0 Query Language 269
 - 6.2.1 The Big Picture 271
 - 6.2.2 SELECT Query 274
 - 6.2.3 CONSTRUCT Query 302
 - 6.2.4 DESCRIBE Query 305
 - 6.2.5 ASK Query 306
 - 6.2.6 What Is Missing from SPARQL 1.0? 307
 - 6.3 SPARQL 1.1 Query Language 308
 - 6.3.1 Introduction: What Is New? 308
 - 6.3.2 SPARQL 1.1 Query 309
 - 6.3.3 SPARQL 1.1 Federated Query 327
 - 6.3.4 SPARQL 1.1 Update 330
 - 6.3.5 Other SPARQL 1.1 Features 342
 - 6.4 Summary 352

Part II Applied Semantic Web

- 7 FOAF: Friend of a Friend 357**
 - 7.1 What FOAF Is and What It Does 357
 - 7.1.1 FOAF in Plain English 357
 - 7.1.2 FOAF in Official Language 358
 - 7.2 Core FOAF Vocabulary and Examples 359
 - 7.2.1 The Big Picture: FOAF Vocabulary 360
 - 7.2.2 Core Terms and Examples 361
 - 7.3 Create Your FOAF Document and Get into the Friend Circle . . . 368
 - 7.3.1 How Does the Circle Work? 369
 - 7.3.2 Create Your FOAF Document 371
 - 7.3.3 Get into the Circle: Publish Your FOAF Document . . . 371
 - 7.3.4 From Web Pages for Human Eyes to Web Pages for
Machines 374
 - 7.4 Semantic Markup: A Connection Between the Two Worlds . . . 375
 - 7.4.1 What Is Semantic Markup? 376
 - 7.4.2 Semantic Markup: Procedure and Example 376
 - 7.4.3 Semantic Markup: Feasibility and Different
Approaches 380
 - 7.5 Summary 382

- 8 DBpedia** 383
 - 8.1 Introduction to DBpedia 383
 - 8.1.1 From Manual Markup to Automatic Generation of Annotation 383
 - 8.1.2 From Wikipedia to DBpedia 384
 - 8.1.3 The Look-and-Feel of DBpedia: Page Redirect 385
 - 8.2 Semantics in DBpedia 389
 - 8.2.1 Infobox Template 389
 - 8.2.2 Creating DBpedia Ontology 392
 - 8.2.3 Infobox Extraction Methods 398
 - 8.3 Accessing DBpedia Dataset 401
 - 8.3.1 Using SPARQL to Query DBpedia 401
 - 8.3.2 Direct Download of DBpedia Datasets 406
 - 8.3.3 Access DBpedia as Linked Data 412
 - 8.4 Summary 414
 - Reference 414
- 9 Linked Open Data** 415
 - 9.1 The Concept of Linked Data and Its Basic Rules 415
 - 9.1.1 The Concept of Linked Data 415
 - 9.1.2 How Big Are the Web of Linked Data and the LOD Project? 417
 - 9.1.3 The Basic Rules of Linked Data 418
 - 9.2 Publishing RDF Data on the Web 419
 - 9.2.1 Identifying Things with URIs 420
 - 9.2.2 Choosing Vocabularies for RDF Data 431
 - 9.2.3 Creating Links to Other RDF Data 433
 - 9.2.4 Serving Information as Linked Data 440
 - 9.3 The Consumption of Linked Data 446
 - 9.3.1 Discover Specific Targets on the Linked Data Web 448
 - 9.3.2 Accessing the Web of Linked Data 452
 - 9.4 Linked Data Application 462
 - 9.4.1 Linked Data Application Example: Revyu 462
 - 9.4.2 Web 2.0 Mashups vs. Linked Data Mashups 470
 - 9.5 Summary 472
- 10 schema.org and Semantic Markup** 475
 - 10.1 Introduction to schema.org 475
 - 10.1.1 What Is schema.org? 475
 - 10.1.2 Understanding the schema.org Vocabulary 477
 - 10.2 Content Markup Using schema.org 479
 - 10.2.1 RDFa 1.1 Lite: A Simple Subset of RDFa 479
 - 10.2.2 What Markup Format to Use? 485
 - 10.2.3 Type Checking and Other Issues 486
 - 10.2.4 Validating Your Markup 488

- 10.3 Content Markup Example 1: Google Rich Snippets 490
 - 10.3.1 What Is Rich Snippets: An Example 490
 - 10.3.2 Google Rich Snippets: Semantic Markup Using schema.org 492
 - 10.3.3 Using Google Rich Snippets Testing Tool 501
- 10.4 Content Markup Example 2: LRMI Project 505
 - 10.4.1 The Idea of LRMI 505
 - 10.4.2 LRMI Specification 507
 - 10.4.3 LRMI Implementation Examples 510
- 10.5 Summary 514
- References 515
- 11 Social Networks and the Semantic Web 517**
 - 11.1 Overview of Social Networking Websites 517
 - 11.2 Facebook’s Open Graph Protocol 519
 - 11.2.1 Open Graph Protocol 520
 - 11.2.2 How Does It Work: Creating Typed Links Using OGP . . . 524
 - 11.2.3 Implications for the Semantic Web 529
 - 11.3 Twitter Cards for Structured Information 530
 - 11.3.1 Twitter Cards Overview 530
 - 11.3.2 How Does It Work: Structured Information for Rich Tweets 533
 - 11.3.3 Structured Information, But Not Semantic Web Yet . . . 538
 - 11.4 Rich Pins for Structured Information 541
 - 11.4.1 Rich Pin Overview 541
 - 11.4.2 How Does It Work: Generating Rich Pins Using schema.org 543
 - 11.4.3 Semantic Markup at Work 547
 - 11.5 Summary 549
- 12 Other Recent Applications: data.gov and Wikidata 551**
 - 12.1 Data.gov and the Semantic Web 551
 - 12.1.1 Understanding Data.gov 551
 - 12.1.2 How Is Data.gov Related to the Semantic Web? 557
 - 12.1.3 Potential eGov Standards: Breaking the Boundaries of Datasets 561
 - 12.1.4 Example Data.gov Applications 564
 - 12.2 Wikidata and the Semantic Web 566
 - 12.2.1 From Wikipedia to Wikidata 566
 - 12.2.2 Three Phases of the Wikidata Project 571
 - 12.2.3 Wikidata as a Data Repository 574
 - 12.2.4 Wikidata and the Semantic Web 577
 - 12.3 Summary 585

Part III Building Your Own Applications on the Semantic Web

- 13 Getting Started: Change Your Data into Structured Data 589**
 - 13.1 RDF Data in General 589
 - 13.1.1 What Does RDF Data Refer to? 590
 - 13.1.2 Decide in Which Format to Publish Your RDF Data 591
 - 13.1.3 Decide Which Ontology to Use to Publish Your Data 596
 - 13.2 Creating RDF Data Manually 602
 - 13.2.1 Popular Editors and Validators 602
 - 13.2.2 Examples: Using TopBraid to Create RDF Data 603
 - 13.3 RDB2RDF: W3C’s Standard for Converting DB Content to RDF Triples 608
 - 13.3.1 RDB2RDF: General Background 608
 - 13.3.2 Direct Mapping from RDB to RDF 609
 - 13.3.3 R2RML: RDB to RDF Mapping You Can Control 613
 - 13.4 RDB2RDF Example Implementation 623
 - 13.4.1 RDB2RDF Direct Mapping 623
 - 13.4.2 Step-by-Step R2RML Example: Virtuoso 624
 - 13.5 Summary 642
- 14 Building the Foundation for Development on the Semantic Web 643**
 - 14.1 Development Tools for the Semantic Web 643
 - 14.1.1 Frameworks for the Semantic Web Applications 643
 - 14.1.2 Reasoners for the Semantic Web Applications 647
 - 14.1.3 Ontology Engineering Environments 650
 - 14.1.4 Other Tools: Search Engines for the Semantic Web 654
 - 14.1.5 Where to Find More? 654
 - 14.2 Semantic Web Application Development Methodology 655
 - 14.2.1 From Domain Models to Ontology-Driven Architecture 655
 - 14.2.2 An Ontology Development Methodology Proposed by Noy and McGuinness 661
 - 14.3 Summary 666
- Reference 667
- 15 Example: Using Jena for Development on the Semantic Web 669**
 - 15.1 Jena: A Semantic Web Framework for Java 669
 - 15.1.1 What Is Jena and What Can It Do for Us? 669
 - 15.1.2 Getting the Jena Package 670
 - 15.1.3 Using Jena in Your Projects 671
 - 15.2 Basic RDF Model Operations 676
 - 15.2.1 Creating an RDF Model 677
 - 15.2.2 Reading an RDF Model 683
 - 15.2.3 Understanding an RDF Model 685

- 15.3 Handling Persistent RDF Models 692
 - 15.3.1 From In-Memory Model to Persistent Model 692
 - 15.3.2 Setting up MySQL 693
 - 15.3.3 Database-Backed RDF Models 694
- 15.4 Inferencing Using Jena 702
 - 15.4.1 Jena Inferencing Model 702
 - 15.4.2 Jena Inferencing Examples 703
- 15.5 Summary 710
- 16 Follow Your Nose: A Basic Semantic Web Agent 711**
 - 16.1 The Principle of Follow-Your-Nose Method 711
 - 16.1.1 What Is the Follow-Your-Nose Method? 711
 - 16.1.2 URI Declarations, Open Linked Data and Follow-Your-Nose Method 713
 - 16.2 A Follow-Your-Nose Agent in Java 714
 - 16.2.1 Building the Agent 714
 - 16.2.2 Running the Agent 721
 - 16.2.3 More Clues for Follow-Your-Nose 724
 - 16.2.4 Can You Follow Your Nose on Traditional Web? 725
 - 16.3 A Better Implementation of Follow-Your-Nose Agent: Using SPARQL Queries 727
 - 16.3.1 In-Memory SPARQL Operation 727
 - 16.3.2 Using SPARQL Endpoints Remotely 732
 - 16.4 Summary 735
- 17 A Search Engine That Supports Rich Snippets 737**
 - 17.1 Why This Is an Interesting Project 737
 - 17.2 Introduction to Lucene 738
 - 17.2.1 Lucene and Our Own Customized Search Engine 738
 - 17.2.2 Core Components of Lucene 739
 - 17.2.3 Use Lucene in Your Development Environment 745
 - 17.3 Preparing the Semantic Markups 746
 - 17.3.1 From Semantic Markup to Rich Snippets 746
 - 17.3.2 Different Deployment Models of the Markup 747
 - 17.3.3 Examples of Markup 749
 - 17.4 Building the Search Engine 752
 - 17.4.1 Creating the Indexer 752
 - 17.4.2 Creating the Searcher 759
 - 17.4.3 Using Web Container to Start the Search 763
 - 17.5 Test It Out and Possible Expansions 768
 - 17.5.1 Test Runs of the Search Engine 768
 - 17.5.2 Possible Expansions 770
 - 17.6 Summary 772

- 18 More Application Examples on the Semantic Web 773**
- 18.1 Building Your Circle of Trust: A FOAF Agent You Can Use . . . 773
 - 18.1.1 Who Is on Your E-Mail List? 773
 - 18.1.2 The Basic Idea 774
 - 18.1.3 Building the EmailAddressCollector Agent . . . 777
 - 18.1.4 Can You Do the Same for the Traditional Web? 788
- 18.2 A ShopBot on the Semantic Web 788
 - 18.2.1 A ShopBot We Can Have 788
 - 18.2.2 A ShopBot We Really Want 790
 - 18.2.3 Building Our ShopBot 799
 - 18.2.4 Discussion: From Prototype to Reality 817
- 18.3 Summary 818
- Index 819**



<http://www.springer.com/978-3-662-43795-7>

A Developer's Guide to the Semantic Web

Yu, L.

2014, XXV, 829 p. 624 illus. in color., Hardcover

ISBN: 978-3-662-43795-7