

 WILEY

Semantic Web Programming

# Web 3.0与 Semantic Web 编程

(美) John Hebler  
Matthew Fisher  
Ryan Blace  
Andrew Perez-Lopez  
唐富年 唐荣年

著  
译



清华大学出版社

# Web 3.0 与 Semantic Web 编程

John Hebler  
(美) Matthew Fisher 著  
Ryan Blace  
Andrew Perez-Lopez  
唐富年 唐荣年 译

清华大学出版社

北 京

John Hebler, Matthew Fisher, Ryan Blace, Andrew Perez-Lopez

Semantic Web Programming

EISBN: 978-0-470-41801-7

Copyright © 2009 by Wiley Publishing, Inc. Indianapolis, Indiana

All Rights Reserved. This translation published under license.

本书中文简体字版由 Wiley Publishing, Inc. 授权清华大学出版社出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

北京市版权局著作权合同登记号 图字: 01-2009-4076

本书封面贴有 Wiley 公司防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

Web 3.0 与 Semantic Web 编程/(美)赫布勒(Hebler, J.)等著; 唐富年, 唐荣年 译.

—北京: 清华大学出版社, 2010.6

书名原文: Semantic Web Programming

ISBN 978-7-302-22636-9

I. W… II. ①赫…②唐…③唐… III. 主页制作—程序设计 IV. TP393.092

中国版本图书馆 CIP 数据核字(2010)第 082560 号

责任编辑: 王 军 韩宏志

装帧设计: 孔祥丰

责任校对: 成凤进

责任印制: 何 芊

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者: 北京鑫丰华彩印有限公司

装 订 者: 三河市新茂装订有限公司

经 销: 全国新华书店

开 本: 185×260 印 张: 31 字 数: 794 千字

版 次: 2010 年 6 月第 1 版 印 次: 2010 年 6 月第 1 次印刷

印 数: 1~4000

定 价: 60.00 元

---

产品编号: 033396-01

# 译者序

自从 Web 之父 Tim Berners-Lee 在 *Scientific American* 上发表那篇著名的文章以来，有关语义 Web 的研究逐年升温，目前已经开始从理论研究走向实际应用的层面，语义 Web 的蓝图已经初步显现。我们已经看到，基于 RDF 和 OWL 的语义 Web 应用已经开始渗透到商业产品中，数据库开始支持 RDF 和 OWL 文档的存储，浏览器开始支持 RDF 和其他本体文档的解析……也许要不了多久，语义 Web 这个概念就会真正走入我们的日常生活中。

目前国内对于语义 Web 的研究大多仍然处于理论研究阶段，尽管也涌现出来了研究成果，但是很可惜，目前尚未出现面向用户的语义 Web 产品。究其原因，是因为在语义 Web 技术的理论和具体的软件开发实践中缺少一个中间环节，即对语义 Web 程序设计的系统阐述。尽管目前关于语义 Web 技术的开源项目很多，但是由于缺乏文档支持和相关的教程，只有少数科研人员能够对这些资源加以利用，而大多数语义 Web 的初学者在起步阶段则需要耗费较多的精力。这本书的亮点就在于引导您从实践出发来学习和理解语义 Web，使您从现实角度感受语义 Web 技术的强大魅力。

本书内容丰富、覆盖面广且工程背景浓厚，对于近年来语义 Web 领域的绝大多数主流技术都有所涉及，通过实例讲解语义 Web 程序设计的具体过程，突出了“以数据为中心”的程序设计思想。语义 Web 程序设计最大的优势是将语义从信息中剥离出来，对语义的重用必然会大大减轻程序开发的负担。通过本书的学习，您将能够从众多的代码示例中获得第一手的编程经验，深刻理解“语义”对于程序设计的深远影响。总之，我们认为这本书会为您打开通向语义 Web 应用程序开发的大门，成为您学习本体和语义 Web 技术的得力帮手。

尽管这本书对内容的讲述深入浅出，但是在开始本书的学习之前，我们建议您至少对 Java 语言有一定了解。鉴于本体和语义 Web 技术相对比较抽象，如果读者具有一定的人工智能、数据库和 Web 信息管理方面的知识基础，那么学习起来会相对轻松一些。此外，语义 Web 技术是一门发展极为迅速的技术，新的标准和思想也不断涌现出来，建议您能够在学习过程中参考 W3C 的相关文档，并且充分利用本书所列出的资源，追踪语义 Web 技术发展的最新动态。

本书的译者此前已经对语义 Web 进行了多年的研究，也做过若干和语义 Web 相关的工程。但是当面对这样一本系统讲述语义 Web 的著作时，仍然感到水平有限，唯恐造成读者对书中内容的误解。由于目前有关语义 Web 的中文书籍非常少，很多专用术语还没有出现统一的译法，本书的翻译难免出现不当之处，请广大读者批评指正。

本书主要由唐富年(funiantang@gmail.com)翻译。此外，唐荣年承担了第2至第6章、第12、13章的翻译。全书的统稿工作由唐富年负责。**Be Flying**工作室负责人肖国尊负责本书译员的选定、翻译质量和进度的控制与管理。敬请广大读者提供反馈意见，读者可以将意见发到 [be-flying@sohu.com](mailto:be-flying@sohu.com)，我们会仔细阅读读者发来的每一封邮件，以求进一步提高今后译著的质量。同时欢迎各位进入 **Be Flying** 工作室博客 [http://blog.csdn.net/be\\_flying/](http://blog.csdn.net/be_flying/)，或者 **China-Pub** 上的宣传链接 <http://www.china-pub.com/main/sale/renwu/GetInfo.asp?theID=64>，来了解 **Be Flying** 工作室的所有其他译著。



## 作者简介

John Hebeler 是一位求知若渴的资深研究人员，目前仍然热衷于研究大规模分布式系统开发方面的新技术。过去 5 年来，他主要研究语义 Web 和新近出现的分布式系统。他撰写了多篇相关的研究论文，与其他学者合著了一本 P2P 网络方面的书，并且出席过很多重要的国际技术会议。目前，他正在马里兰大学攻读信息系统博士学位。此外，他还是 BBN 科技公司某分部的科学家。

Matthew Fisher 在软件和系统开发这一领域拥有 15 年以上的工作经验。他在很多工程性很强的岗位上工作过，他所供职的单位中既有刚处于创业阶段的小公司，有研究与开发公司，也有被《财富》杂志列入全球企业 50 强的公司。他经常为 Semantic Report(关于语义 Web 技术和应用的一份月度通讯)撰稿，也经常参加一些和语义 Web 技术相关的国际会议(如 OWLED、ISWC 和语义技术会议等)。Matthew 是 Progeny Systems 公司的首席系统工程师，拥有 Penn State 大学计算机科学专业的学士学位，以及乔治梅森大学计算机科学专业的硕士学位。

Ryan Blace 已经在 BBN 技术公司工作 5 年之久，一直是一位语义 Web 开发人员。他致力于为政府和商业部门开发基于语义 Web 的大型知识管理系统。Ryan 拥有马里兰大学计算机工程专业的学士学位，而且目前正在弗吉尼亚理工大学攻读计算机科学专业的硕士学位。如果 Ryan 没有坐在计算机前熬到深夜，那么他十有八九会骑车去兜风，去蹬山地车，也有可能会在汽车俱乐部的跑道上从事培训工作。

Andrew Perez-Lopez 是一位软件开发人员，从 2005 年起在 BBN 科技公司供职，他的工作就是使用本书讲述的各种语义 Web 技术实现大规模信息集成系统。他拥有 Virginia Tech 计算机科学专业的硕士学位，以及 Virginia 大学的认知科学(Cognitive Science)专业的学士学位。



## 技术编辑简介

Mike Dean(复审者/编辑)是BBN技术公司的首席工程师,他从1984年起就在这里工作了。他从2000年开始从事语义 Web 方面的研究,担任 DARPA Agent(美国国防部高级研究计划局)标记语言(DAML)计划的首席研究员。他是 W3C OWL Reference 的编辑之一,也是 SWRL 的作者之一。同时他还开发了多种语义 Web 工具、数据集和应用程序。目前,他为 BBN 科技公司的多个语义 Web 项目做技术指导。他拥有斯坦福大学计算机工程专业的学士学位。

Mike Smith(技术编辑)是 Clark & Parsia 有限责任公司的一位高级工程师。这是一家软件开发和咨询公司,专门研究人工智能技术方面的开发和应用。Mike 是 W3C OWL 工作组的成员,在 OWL 领域表现活跃,并且经常在 <http://clarkparsia.com/weblog/>上发表自己的一些见解。他是 Pellet(开源的 OWL 推理机)的主要开发者之一,并且经常为 Protégé 和 OWL API 项目做一些工作。他拥有弗吉尼亚大学系统与信息工程专业的学士学位和硕士学位。

# 序 言

在 BBN 技术公司，我们这个团队自 2000 年以来就一直工作在语义 Web 技术的最前沿，刚开始是作为 DARPA Agent 标记语言(DAML)计划的一份子，随后就开始为政府和商业客户开发各种各样的工具、数据集和应用程序。本书的作者和技术编辑都是(或者曾经是)这个团队的成员，现在这个团队已经成长为一个拥有 30 位员工的团队。本书反映出了我们作为软件开发人员的知识背景，蕴含了我们在过去 8 年中所积累的经验来之不易的见解。

语义 Web 是一项国际性的工作，需要以便于自动处理、集成和推理的形式来表示数据(包括当前万维网上那些为人类用户所设计的数据)。数据极为重要，如果将分散的数据源连接起来形成一个互联的数据网，那么这些数据就会具有更高的价值。目前与语义 Web 相关的应用包括从 mash-up(混搭)到企业的数据集成、更高级的数据检索、服务组合、智能代理、桌面应用程序与移动应用程序以及协作等。

由于美国和欧盟一些相关研究计划的推动，研究语义 Web 的群体也在不断壮大，这包括 W3C 语义 Web 行动组、许多大大小小的经销商、一系列关于语义 Web 和语义技术的会议以及大量的开源项目和开发人员。

从许多方面来看，Web 3.0 是语义 Web 的另一个比较恰当的名称。Web 2.0 通过 FOAF、RSS 1.0、形形色色的语义 Wiki 项目以及与他人共享的收藏(如 MusicBrainz)等搭建社会网络 and 实现协作，而语义 Web 也同样重视这些方面。语义 Web 上的本体不但保留了 Web 2.0 标签(tag)、微格式(microformats)和自由分类法(folksonomy)的灵活性，而且比它们提供了更多的结构(数据的结构化有助于机器自动处理数据)。

语义 Web 标准，诸如 RDF、OWL 和 SPARQL 在使用过程中也在不断进行演化。与此同时大量面向不同编程环境的高质量工具也涌现了出来，其中有很多都是开源的。Link Open Data 项目满足了一项重要的需求，即能够为多个应用程序提供基础数据，而目前这种需求正日益增长。此外，当前很多与语义 Web 相关的工具和应用程序都具备了高度的可伸缩性。



开发人员通常都会通过阅读别人的代码而获得编程经验。贯穿本书始末，我们采用了一种实用主义的做法，通过大量代码示例和一个涉及多个章节的应用程序来讲述语义 Web 程序设计。

我们希望您能够认识到语义 Web 技术为计算机应用领域当前要面对或者即将要面对的一些挑战找到了有效的解决方法；也希望您能够像我们一样，通过学习和掌握语义 Web 技术而使自己的工作能力再跨上一个新台阶。

BBN Technologies 首席工程师 Mike Dean  
2008 年 11 月作于美国密歇根州安阿伯市



# 前 言

本书直接而果断地使用语义 Web 技术来解决现实世界中的实际问题，研究如何更为灵活地访问由便携式计算机、企业、甚至是万维网所提供的信息和服务。各个章节都按照统一的、基于代码的方式来讲述如何解决访问信息和服务所面临的挑战。通过列举代码示例，我们将从实用角度出发探索构成语义 Web 的多种技术，包括知识表示，如微格式、资源描述框架(RDF)、RDF Schema(RDFS)、Web 本体语言(OWL，包括其最新发布的版本 OWL 2 和语义 Web 规则语言 SWRL)等，以及语义 Web 程序设计框架(如 Jena)和其他一些有用的语义 Web 工具。我们在探索这些技术的时候，不是仅从这些技术本身出发来考虑问题，还要关注它们在解决实际问题的过程中所体现出来的作用和价值。因此，您的学习应当是基于效果的——即依据每种技术在您解决实际应用中的各种难题时所能发挥的效果。

本书的编写得益于我们多年来设计大规模语义 Web 解决方案、建立语义 Web 工具和参与修订语义 Web 相关标准所积累的经验。我们对这一领域十分熟悉！正是因为具有这样的技术背景，我们不仅能够帮助您理解这门强大的技术，而且还能够帮助您直接将这门技术应用到实际应用中，解决信息领域中的多种挑战。

## 内容概述

语义 Web 技术为驾驭多种信息和信息服务提供了一种强大的、实用的方法。语义可以保证“使更多信息变得更有用”，避免了信息泛滥带来的负面影响。这就需要采用新的数据表示方法来改进人们捕捉和共享知识的能力，也需要新的程序设计构造(construct)和工具来确保这些信息能够为您的应用程序所使用。

本书通过实际的数据格式、工作代码和工具来探索语义 Web 技术。我们从开发人员的视角出发考查语义 Web 技术，目的就是为在实际应用中得到满意的效果。我们针对您在创建和管理语义 Web 应用时的需求给出了解释和说明。全书大量的工作代码示例既增加了技术背景可信度，也有助于读者对说明性文字的理解。一般来说，代码一目了然，更能说明问题。因此，我们强烈推荐您从示例代码入手，并且根据自己的需求进行调整。这样做有助于您尽快理解语义 Web，适应从语义 Web 的角度出发考虑问题，进而使您能够迅速在工作中应用语义 Web 技术。

## 编排方式

本书总共有 15 章，分为 4 个部分。在附录中，本书还给出了一系列与主要技术相关的参考资料。

第 I 部分：语义 Web 程序设计简介，由第 1 章和第 2 章组成。这一部分的内容将使您尽快认识语义 Web 程序设计。第 1 章“为以数据为中心的语义 Web 程序设计做好准备”介绍了语义 Web 的主要概念以及这些概念之间的相互关系。这一章能够使您了解作为一个语义 Web 开发人员所应知道的各种词汇。同时，第 1 章还指出了语义 Web 技术的优点和对程序设计的影响，并且在这一章结尾之处给出了当前使用语义 Web 时无法避开的问题实例。第 2 章“Hello Semantic Web World”围绕“Hello Semantic Web World”这一程序详细地对工作代码进行了分析。这个例子可以使您了解从如何建立开发环境到如何使用推理机的一系列操作。这一章的说明性文字比较简单，因为这一章仅仅是对本书其余部分的简介。如果您还是一位语义 Web 领域的新手，那么这一章对您来说至关重要。而对语义 Web 比较熟悉的读者可以选择跳过这两章的学习。

第 II 部分：语义 Web 程序设计基础，包括从第 3 章到第 7 章共 5 章的内容。语义 Web 应用的驱动力来自两个领域：知识表示和应用集成。这一部分主要讲述前一个领域——知识的表示和操作。第 3 章“信息建模”讲述了如何通过 RDF 构建数据模型。第 4 章“融入语义”在使用 RDFS 和 OWL 2 创建知识模型的过程中引入了本体(ontology)。第 5 章“现实世界中的知识建模”通过应用程序框架和推理机来练习使用工作本体。第 6 章“发现信息”对知识模型进行了深入分析，通过搜索、导航和借助 SPARQL 进行形式化的查询等方式从中抽取有用的信息。第 7 章“添加规则”通过研究语义规则语言来充实知识表示，包括 W3C 标准 SWRL。

第 III 部分：建立语义 Web 应用程序，涵盖从第 8 章到第 11 章的内容。这一部分针对的是上面提到的驱动语义 Web 技术的第二个领域——将知识和作用于知识之上的应用程序整合到一起。这一部分的内容奠定了可靠的语义 Web 程序设计基础。第 8 章“应用程序设计框架”借助 Jena 语义 Web 框架的大量实例全面探讨了语义 Web 框架。章末简要讲述了一个语义 Web 应用程序 FriendTracker，这个例子是贯穿后面三章的一条主线，围绕这个例子我们探讨了如何对格式和位置均有所不同的数据和信息进行集成、对准和输出。第 9 章“整合信息”重点研究了如何将来自多个数据源(如关系数据库、Web 服务和其他形式的数据源)的数据整合到一个知识模型中。第 10 章“对准信息”重点探讨了如何将数据和本体概念进行对准，进而实现异构信息的集成。第 11 章“信息共享”介绍了如何以多种格式输出信息，如 RDFa、微格式和 SPARQL 端点等。我们在上述的各章中都采用 FriendTracker 应用程序实例进行说明，希望能够更为直接地向您阐述语义 Web 程序设计的理念。

第 IV 部分：扩展语义 Web 程序设计，这一部分涵盖了从第 12 章到第 15 章的几章内容。现在您已经具有了可靠的知识表示和语义 Web 应用程序开发基础，我们可以在此基础之上扩展到功能更为强大、更具有实用价值的领域，包括语义服务、时空问题、语义 Web 架构和最优方法，并且几乎对本书所提到的各种工具都进行了扩展应用。第 12 章“开发和使用语义服务”研究了如何在服务中加入语义，使得这些服务能够融入到语义 Web 当中。第 13 章“管理空间和时间”从时空角度对您的知识表示进行了考查。第 14 章“语义 Web 模式和最佳实践”依次对各章内容进行了回顾，以本书前面讲述的所有内容为基础，介绍了一系列构建各

种语义 Web 应用程序的架构模式。第 15 章“进阶”通过洞悉语义 Web 技术的未来总结了全书。这一章重点对语义 Web 技术最为关键而且仍在不断演进的四个领域进行了研讨，包括本体管理、高级集成和分发、高级推理以及可视化。它们对目前语义 Web 领域中仍然不断演进的各种技术进行了较为严谨的阐述。

## 读者对象

本书从综合、实用的角度为使用语义 Web 技术开发应用程序提供了支持。语义 Web 可以对万维网上已有的多种分布式信息和服务、商业企业和您的个人资源都加以利用。因此，无论您仅仅关注信息还是关注整个应用程序，大多数技术性读者都会从本书中受益。

通过本书的学习，开发人员能够从众多的代码示例中获得第一手的程序设计经验，无论是应用程序开发人员还是仅仅关心多种形式的信息(从数据库模式到 XML 格式)的信息开发人员，学习本书都会有所收获。这本书为您开始设计自己的语义 Web 应用程序提供了必需的工具、背景知识和丰富的示例。

大型应用程序的架构师通过本书的学习能够对语义 Web 的作用有更为深入的理解。语义 Web 对所有使用信息的系统(几乎是所有的系统)都有所裨益，它能够迅速扩展系统的功能，进而可以更好地利用信息和服务。全书的应用程序对系统架构师来说也十分有用，而在信息和数据管理方面的详细阐述也对负责数据格式和数据处理的信息架构师有所裨益。

技术管理人员可以通过本书对语义 Web 的强大功能、风险和优势有更为深入的理解。语义 Web 是一门战略型的技术，是一种真正能够提供巨大优势的解决方案。目前信息领域面临很多很棘手但是却很有价值的挑战，这些难题通常涉及海量信息和服务，而语义 Web 技术为解决这些难题提供了新的思路。在任何解决方案中，是否需要采用语义 Web 技术都取决于动态信息和服务资源。本书所给出的代码示例揭示了这门技术的可信度，同时也洞悉了语义 Web 程序设计自身的挑战，即如何更好地进行规划。

## 所需工具

我们强烈建议您下载和定制本书提供的所有代码示例，以便巩固所学的知识。书中涉及到的所有软件工具都是开源的，可以直接从万维网上下载，我们在书中也提供了必要的说明和相关链接。只要您的操作系统支持 Java 1.5 虚拟机，那么您的计算机就能和书中提到的所有工具兼容。而且当我们在书中介绍每一种工具的时候，同时也会给出下载、安装和配置该工具的说明。此外，我们在附录 F 中还对所有工具进行了总结说明。

## 网站资源

在本书配套网站 <http://semwebprogramming.org> 上还有大量可供参考的资源。在该网站上您可以访问本书参考的所有文献资料、完整的代码示例以及本体等，而且您还有机会参与当前正在进行的一些讨论和活动。该网站上也正在不断更新与语义 Web 相关的书籍和代码，反

映了语义 Web 的持续扩展和演进。我们欢迎广大读者针对本书和本书相关的示例留下宝贵意见。

我们的网站还提供了博客和 wiki 等功能，等待您在这里发表您的作品和见解。网站提供的 wiki 是一种语义 wiki，提供了 SPARQL 端点。您既可以自由地申请博客或注册 wiki，也可以自由提问或者参与到当前的讨论中来。我们发现最好的学习方式就是经常提问并多与他人交流——到处打听您想知道的答案。

## 总结

语义 Web 程序设计是一种令人振奋、功能强大的新方法，能够更好地利用已有的海量信息和服务。伴随这种强大而令人兴奋的功能同时出现的还有一系列新词汇、新工具和构建工作应用程序的新思路。前面提到的各章内容以一种实用的方式铺就了一条有效构建语义 Web 应用程序的平坦大道，而这些语义 Web 应用程序整合了丰富的、可访问的动态信息和服务。让我们开始学习吧！

# 目 录

## 第 I 部分 语义 Web 程序设计简介

第 1 章 为以数据为中心的语义 Web 程序设计做好准备	3
1.1 定义语义 Web	4
1.2 确定主要的程序设计组件	9
1.3 确定语义 Web 技术对程序设计的影响	11
1.3.1 确立以 Web 数据为中心的视角	11
1.3.2 表达语义数据	12
1.3.3 共享数据	13
1.3.4 使数据动态而灵活	14
1.4 避免阻碍、神化和夸张	16
1.4.1 语义 Web 遇到的阻碍	16
1.4.2 对语义 Web 的神化	17
1.4.3 对语义 Web 的夸大	18
1.5 了解语义 Web 的起源	19
1.6 探索语义 Web 示例	21
1.6.1 语义 Wiki(semantic-mediawiki.org)	21
1.6.2 Twine(www.twine.com)	22
1.6.3 FOAF 项目(www.foaf-project.org)	23
1.6.4 RDFa 和微格式	24
1.6.5 语义查询端点(dbpedia.org/sparql)	26
1.6.6 语义搜索(www.trueknowledge.com)	26

1.7 总结与展望	27
1.8 参考资源	27

第 2 章 Hello Semantic Web World	29
2.1 建立语义 Web 开发环境	29
2.2 编写“Hello Semantic Web World”应用程序	31
2.3 小结	48

## 第 II 部分 语义 Web 程序设计基础

第 3 章 信息建模	53
3.1 软件中的信息建模	53
3.1.1 共享信息：语法和语义	54
3.1.2 信息共享中的元数据和数据	56
3.2 语义 Web 信息模型：资源描述框架(RDF)	57
3.2.1 节点：资源和文字	57
3.2.2 边：谓词	59
3.3 使用 RDF 进行信息交换	59
3.3.1 可看作点的陈述	60
3.3.2 RDF 序列化	61
3.3.3 更多关于 RDF 的内容	69
3.4 小结	75
第 4 章 融入语义	77
4.1 Web 上的语义	78
4.1.1 动机因素	78
4.1.2 使用 RDF 表达语义	81
4.2 本体入门	84

4.2.1	分布式知识	84	5.4.6	选择合适的基础本体	147
4.2.2	本体元素概述	85	5.5	小结	148
4.3	本体的元素	87	<b>第 6 章</b>	<b>发现信息</b>	<b>149</b>
4.3.1	OWL 2 类型	88	6.1	导航语义 Web	150
4.3.2	本体首部	88	6.2	搜索语义 Web	153
4.3.3	标注	89	6.3	查询语义 Web	154
4.3.4	基本分类	89	6.3.1	SPARQL 快速入门	155
4.3.5	定义和使用属性	93	6.3.2	四种基本的查询形式	158
4.3.6	属性的定义域和值域	93	6.3.3	SELECT 的精髓	159
4.3.7	描述属性	93	6.3.4	RDF 数据集、FROM 和 FROM NAMED	163
4.3.8	数据类型	99	6.3.5	查询修饰符	169
4.3.9	否定属性断言	102	6.3.6	CONSTRUCT 的精髓	180
4.3.10	属性约束	103	6.3.7	DESCRIBE 的精髓	182
4.3.11	高级类描述	108	6.3.8	ASK 的精髓	183
4.3.12	OWL 中的等价	111	6.3.9	SPARQL 蕴含	184
4.4	小结	113	6.3.10	不支持的功能	185
<b>第 5 章</b>	<b>现实世界中的知识建模</b>	<b>115</b>	6.4	小结	186
5.1	探究语义 Web 的组件	115	<b>第 7 章</b>	<b>添加规则</b>	<b>187</b>
5.1.1	语义 Web 框架	116	7.1	规则的含义	188
5.1.2	存储和检索 RDF	117	7.2	使用规则的原因	188
5.1.3	辨识 OWL 的语义	120	7.2.1	不支持对属性的组合	188
5.1.4	常见框架和组件	124	7.2.2	内置的使用	188
5.1.5	知识库性能	127	7.2.3	本体式中介	188
5.2	探索 OWL Profile	128	7.2.4	有限制的假设	189
5.2.1	OWL Full 和 OWL DL	128	7.3	规则语言	189
5.2.2	OWL Profile	129	7.4	SWRL 的精髓	189
5.3	OWL 推理演示	131	7.4.1	抽象语法	191
5.3.1	本体	131	7.4.2	XML 的具体语法	192
5.3.2	示例应用程序	133	7.4.3	RDF 的具体语法	198
5.3.3	结果	136	7.4.4	内置	198
5.4	使用本体	139	7.4.5	DL-Safe 规则	199
5.4.1	将知识模型从应用程序中 分离出来	140	7.4.6	本体式中介	205
5.4.2	跨领域和应用程序边界 实现共享	141	7.5	Jena 规则	209
5.4.3	基础本体的含义	142	7.6	规则交换格式	211
5.4.4	常见的基础本体	143	7.6.1	深入研究细节	211
5.4.5	查找本体进行重用或扩展	146	7.6.2	RIF 的未来	212
			7.7	小结	212

**第 III 部分 建立语义 Web 应用程序**

<b>第 8 章 应用程序设计框架</b> .....	215
8.1 构建语义 Web 框架 .....	215
8.2 Jena 语义 Web 框架 .....	218
8.3 使用 Jena 进行程序设计 .....	221
8.3.1 建立 Jena 开发环境 .....	223
8.3.2 建立知识库: 建立模型 .....	224
8.3.3 使用语义 Web 数据 填充模型 .....	226
8.3.4 合并语义 Web 数据 .....	229
8.3.5 询问语义 Web 数据 .....	230
8.3.6 语义 Web 数据推理 .....	232
8.3.7 导出语义 Web 数据 .....	235
8.3.8 释放语义 Web 数据操作 所占用的资源 .....	236
8.3.9 管理语义 Web 数据 .....	236
8.4 通用应用程序示例 FriendTracker 概述 .....	242
8.5 小结 .....	243
<b>第 9 章 整合信息</b> .....	245
9.1 整合信息 .....	246
9.1.1 表示信息 .....	246
9.1.2 各种表示之间的转换 .....	247
9.1.3 解决转换过程中的难题 .....	248
9.1.4 介绍 FriendTracker 的 数据源 .....	249
9.2 以 RDF 格式公开基于 XML 的 Web 服务 .....	251
9.2.1 Weather.gov XML Feed 介绍 .....	252
9.2.2 使用 XSL 转换来公开 XML .....	252
9.2.3 使用 XML 绑定和 Velocity 来公开 XML .....	266
9.3 将关系数据库公开为 RDF .....	274
9.4 公开其他类型的数据源 .....	281
9.4.1 使用一个定制的流式 RDF Writer 来公开 Jabber .....	281

9.4.2 使用 Reflection 来公开 Java 对象 .....	285
9.5 小结 .....	292
<b>第 10 章 对准信息</b> .....	293
10.1 数据源本体、领域本体和 应用程序本体 .....	293
10.2 本体的对准 .....	294
10.2.1 本体构造 .....	295
10.2.2 通过规则进行转换 .....	296
10.2.3 显式转换 .....	297
10.2.4 特定的转换方法 .....	297
10.3 FriendTracker .....	297
10.3.1 使用 OWL 和 SWRL 来 对准本体 .....	302
10.3.2 使用 XSLT 来对准本体 .....	307
10.3.3 使用代码来对准本体 .....	311
10.3.4 使用 RDFS 来对准 简单本体 .....	312
10.4 记录链接 .....	314
10.5 小结 .....	317
<b>第 11 章 信息共享</b> .....	319
11.1 微格式 .....	319
11.2 eRDF .....	321
11.3 RDFa .....	324
11.3.1 已支持的属性 .....	324
11.3.2 空白节点 .....	330
11.3.3 语言支持 .....	331
11.4 工具和框架 .....	331
11.4.1 RDF 转换工具 .....	332
11.4.2 SPARQL 端点 .....	332
11.4.3 xOperator .....	335
11.5 RDFa 版本的 FriendTracker .....	337
11.6 小结 .....	344
<b>第 IV 部分 扩展语义 Web 程序设计</b>	
<b>第 12 章 开发和使用语义服务</b> .....	347
12.1 背景 .....	348
12.1.1 发现 .....	350



12.1.2	调用	350	14.2.3	非结构化数据源和 客户端应用程序	392
12.1.3	协商	350	14.3	协调语义服务	393
12.1.4	错误处理	350	14.4	应用语义 Web 最佳实践	394
12.1.5	监视	350	14.4.1	创建 URI	394
12.1.6	组合	351	14.4.2	指定度量单位	397
12.2	实现语义服务	351	14.4.3	表示 N 元关系	399
12.3	Web 服务语义标记	352	14.4.4	管理不良数据	399
12.3.1	ServiceProfile	352	14.5	小结	400
12.3.2	ServiceModel	353	<b>第 15 章</b>	<b>进阶</b>	<b>401</b>
12.3.3	ServiceGrounding	353	15.1	改进本体	402
12.4	Web 服务建模本体	353	15.1.1	本体仓储和注册站	402
12.5	WSDL 语义标注	356	15.1.2	Linked Data	403
12.5.1	SAWSDL 举例	357	15.1.3	版本控制	405
12.5.2	SAWSDL 工具	358	15.1.4	本体指标	405
12.6	小结	359	15.2	改进集成	407
<b>第 13 章</b>	<b>管理空间和时间</b>	<b>361</b>	15.2.1	语义管道	407
13.1	软件中的空间和时间	361	15.2.2	分布式查询	408
13.1.1	空间信息	362	15.2.3	对准	409
13.1.2	时间信息	363	15.3	改进推理	414
13.2	表示语义 Web 上的 时空数据	364	15.3.1	规则交换格式(RIF)	414
13.3	使用 Jena 实现的时空软件	371	15.3.2	概率推理	415
13.3.1	使用空间数据	372	15.3.3	信任: 证明标记语言	418
13.3.2	示例: 空间查询	374	15.3.4	LarKC: 大型知识 对撞机	420
13.3.3	示例: 事务处理时间 界定的查询	381	15.4	改进可视化	421
13.4	小结	384	15.5	小结	424
13.5	参考资源	384	<b>附录 A</b>	<b>RDF</b>	<b>427</b>
<b>第 14 章</b>	<b>语义 Web 模式和最佳实践</b>	<b>385</b>	<b>附录 B</b>	<b>OWL Web 本体语言</b>	<b>433</b>
14.1	聚合不同的数据源	385	<b>附录 C</b>	<b>SWRL</b>	<b>439</b>
14.1.1	将数据源公开为 RDF	386	<b>附录 D</b>	<b>SPARQL</b>	<b>455</b>
14.1.2	将数据引入领域 知识模型	387	<b>附录 E</b>	<b>Jena 参考指南</b>	<b>465</b>
14.1.3	在知识库中存储信息	388	<b>附录 F</b>	<b>安装参考指南</b>	<b>473</b>
14.1.4	启动数据流程	388			
14.2	标注非结构化的数据	389			
14.2.1	标注管理	390			
14.2.2	本体管理	391			