维普资讯 http://www.cqvip.com

基于 OIL 和 RDFS 的语义化 Web 逻辑扩展

An Extension on Logic of Semantic Web Based on OIL and RDF

姚绍文[1,3] 宗 勇[2] 刘爱莲[1] 周明天[3]

(昆明理工大学通信与电子工程系 昆明650051)[1](云南大学电教中心 昆明650091)[2]

(电子科技大学计算机科学与工程学院 成都610054)[3]

Abstract RDF(S) is the standard for metadata of Web that aims to turn the current Web into the foundation of a machine-understandable knowledge system. By employing the flow-blown technique of KE, the Web-NG is targeted to provide the semantic inter-operability for data and knowledge exchange. As a hot spot in KE, ontology can be well integrated with the Web and such integration will support knowledge modeling and description on the Web-based applications. This paper introduces RDF(S), Web-NG and an Web-based ontology modeling language OIL. Based on the introduction, the paper defines an extension of OIL/RDFS on proposition logic, and presents an approach for modeling proposition formula and inferring rule.

Keywords Semantic Web, Ontology, OIL, RDF(S)

1 引言

Web 的发展使得 Internet 能够迅速地普及,通过 Web 人们可以得到各种各样的服务。随着新的应用不断加入到 Web 应用,Web 的发展面临着应用标准不兼容、语义定义不严格、有效性和结构化缺乏等大量问题。这些问题使得针对 Web 资源难以进行机器自动查询、检索。Web 上现有的信息完全是非结构化或半结构化的信息,即使通过 CGI 从数据库中得到了结构化的数据,插入到 Web 页面中也失去了其原有的语义属性。要使得 Web 能够象数据库管理系统甚至知识库系统一样支持诸如描述、搜索、管理、交换和更新等准确的处理,就需要处理 Web 的元信息,把元信息以某种方式结构化[9,15]。

元信息就是关于信息的信息,包括标记、分类和描述信息,元信息从数据的角度出发又称为元数据。在数据库中元数据就是数据库的模式,传统的网络、计算机技术和知识工程定义了多种描述元数据的标准。为了从现有的 Web 体系结构进行平滑的过渡,W3C 和其他的研究机构开发了多种 Web 元数据应用,这些应用包括资源分类、软件代理、资源发现、数据知识挖掘、电子商务、内容分等级、智能属性版权、数字签名以及个人隐私等[1,2,11]。

利用 XML[4]提供的表示能力,W3C 开发了在 Web 上进行元数据处理和交换的基础,定义了 PICS(Internet 内容选择平台)[1]和 RDF(S)[1,7,8],其中 RDF 比 PICS 提供了更通用的元数据处理。RDF(S)是在 PICS 基础上,组合了 Netscape 公司的 MCF(元内容框架)、微软公司的 Web Collections 以及传统的知识表达理论形成的,同时 RDF 的设计也受到了面向对象编程和建模语言、数据库的影响。XML 促进了基于 Web 的各种新型的软件体系结构、数据交换、资源管理等应用开发,推动了 Web 技术向下一代发展过渡[5,10]。下一代 Web (Web-NG)被称为"可知识化的 Web",Tim Berners Lee 又称之为"语义化的 Web"[11]。RDF(S)的出现使 Web 转变为机器可理解的知识库成为可能,因为 RDF(S)的出现使得知识工程研究中成熟的技术成为 Web-NG 发展中技术和方法上的动力。

作为人工智能(AI)的一个新学科,知识工程(KE)旨在以一种系统的和可控制的方式构建基于知识库的系统。KE 的研究形成了一些关于知识模型构造、知识模型体系构建和重用的主要研究成果,但成功的应用实例还相当有限[12]。KE 开发的建模理论包括了传统的谓词逻辑、语义网络、框架逻辑、描述逻辑和领域模型(Ontology)建模,其中领域模型成为近年研究的热点[13]。领域模型是指对应用领域中一系列的概念或术语进行抽象,开发一些公共的概念以利于知识的共享和重用[14]。Web-NG 需要知识工程更先进的方法和技术,而 Web 的飞速发展为知识工程的应用建立了一种知识共享的基础结构平台。

OIL(Ontology Inference Layer 或 Ontology Interchange Layer)是欧洲的 OntoKnowledge 联盟针对基于 Web 环境进行知识工程建模提出的一种领域模型语言,用分层的方法进行定义[3]。运用 OIL 可以定义面向应用的领域模型,同时定义的模型可以用 XML 的 DTD、XML 模式或 RDF 数据模型进行描述,这种模型的描述可以在 Web 环境中实现透明的交换与共享。由于 RDFS 提供了类似面向对象的特性,可以进行语义上的扩展,同时 RDFS 又是一种能定义基本词汇的简单领域模型语言,因此我们认为:比起 DTD 和 XMLS,RDF(S)将是语义化 Web 发展和基于 Web 的领域模型建模更为主要的动力。

本文以命题逻辑公式和推理规则的领域模型抽象作为研究对象。通过对 RDF(S)和 OIL 进行扩展,我们定义了命题逻辑的 OIL/RDFS 结构包容关系和相应的 RDF 模式定义,并以推理公式作为实例进行表示。本文接下来首先介绍了 RDF 数据模型、RDFS 的建模原语和语义化 Web 的结构,然后介绍了领域模型论和 OIL 对 RDFS 的扩展。在此基础上,我们定义了一种针对命题逻辑的扩展方法和对应的 RDF 模式定义,并用所定义的方法表示了命题逻辑典型的推理公式。

2 RDF(S)和语义化 Web 的结构

2.1 RDF 数据模型和 RDFS 建模原语

RDF 定义了资源、属性、事实三种基本实体,RDF 中任何

维普资讯 http://www.cqvip.com

可以通过 URI 表示的对象都是资源。资源之间可相互联系或通过属性-值对联系，这种联系表示了一个事实。此外事实本身可被看作一个资源，即可以建立具体化的 RDF 数据模型。文[7,9]深入地介绍了 RDF 模型和概念。

RDFS[8]定义了一种模式定义语言，提供了一个定义在 RDF 之上抽象的词汇集。该词汇集构成了一种基本类型系统及该系统的基础类，这些基础类包括用来定义新类的必要的元对象类型，因此 RDFS 允许显式地定义对象语义对对象模型建模，图1示出了 RDFS 中定义的类和资源。针对领域模型建模，RDFS 因其具有扩展特性显得更为重要。RDFS 定义的主要类如下：

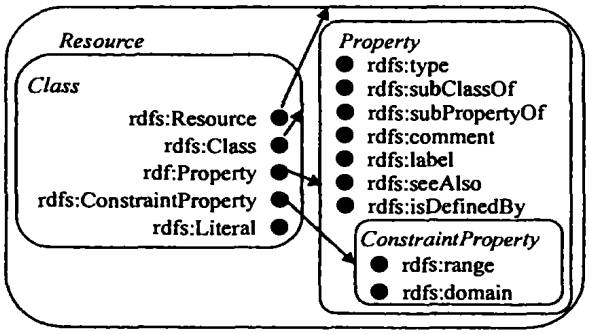


图1　RDFS 中定义的类和资源

·rdfs:Resource：RDF(S)中最通用的类。它有两个子类，即 rdfs:Class 和 rdfs:Property。当定义一个特定领域的 RDFS 模式，这个模式定义的类和属性将成为这两个资源的实例。

·rdfs:Class：表示了关于资源的所有类的集合。这就意味着诸如 appl:Person 和 appl:Organization 是元类 rdfs:Class 的实例。

·属性与类一样，即在一个特定应用的 RDF 模式定义中的每一个属性是一个 rdf:Property 的实例。

·RDFS 定义了特征属性 rdfs:subClassof，rdfs:subClassof 定义了类之间的子类继承关系。由于 rdfs:subClassof 是可传递的，rdfs:subClassof 定义了一个类的层次关系，特殊的类继承更通用类的特性。如果一个资源是一个类的实例，则它自然是该类的超类的实例。

·与 rdfs:subClassof 类似，另一种特殊类型的关系 rdfs:subPropertyof 定义了一种属性的层次关系，如 FATHEROF 与 PARENTOF 之间的关系可以视为一个 rdf:subPropertyof 关系。

·RDFS 允许定义与属性相关的领域和范围的约束(rdfs:domain 和 rdfs:range)。

用 RDFS 可以定义面向不同应用领域的 RDF 模式——词汇集。其中针对数字图书馆领域定义的通用模式是 Dublin Core(DC)元数据词汇集，DC 不仅是影响 RDF 开发的主要因素之一，而且其中的元素可以通过名字空间在 RDF 描述中运用。除了面向数字图书馆的资源描述定义通用的元数据外，DC 元数据定义的基本目标之一就是应用于 Web 资源的描述。DC 元数据词汇集中定义了15个元素(标题、创作者、主题、描述、出版商、贡献者、日期、类型、格式、标识、源、语言、关系、覆盖和版权)，这些元素包含了与资源描述相关的主要方面。DC 词汇集得到广泛的应用，备受博物馆、图书馆、政府和商业组织等资源描述机构的关注。

2.2　语义化 Web 的结构

Web 是一个交换数据和知识的通用媒介，语义化的 Web 需要语义级上的互操作性，而语义上的互操作性，不仅需要针对文档语法形式的标准，而且还需要针对语义内容的标准。近年来的 W3C 标准化过程中提出的建议主要是针对语义上的互操作性，特别是 XML/XMLS[6]和 RDF/RDFS，因此形成了建立语义化 Web 的不同看法和观点，一种看法是基于 XML/XMLS，另一种看法是基于 RDF/RDFS。我们认为基于 RDF/RDFS 实现语义上的互操作比基于 XML/XMLS 更具有优越性。

RDF 的特点在于其灵活性和易扩展性，因此可以根据 RDF(S)创建大量的模式，同时在 Web 上不断出现新的模式或改进已有的模式是很平常的事情。RDFS 的类型系统类似于面向对象编程语言的类型系统，如 JAVA，因而容易实现扩展。RDF 以属性为中心的方法的一个好处就是任何人关于现有的资源都可以很容易地说明他们想要说的任何东西，而这正是 Web 结构化原则的一个方面。对 RDF 的扩展包括了针对特定应用领域的 RDF 模式扩展定义和理论抽象上的深入扩展。构建语义化的 Web，需要使用分层的方法，因此所有对 RDF(S)的扩展在理论上都需要并且也可以进行一种逻辑的概念分层。

语义化的 Web 通过分层方式来实现已经在不同的研究者之间达成了共识，图2是语义化 Web 的三层结构。最低层需要一种表示机器可读的数据语义的通用机制，这种机制可以由 RDF 担当，RDF 提供了一种描述元数据的简单的数据模型和标准的语法。中间层是模式层，针对元数据需要引入定义词汇集、结构和约束的方法，RDFS 可以满足该层的要求。关于第三层，Tim Berners-Lee 描述为："下一层是逻辑层，我们需要把逻辑编写为文档的方式，形成诸如规则、检查和解析的东西，这些规则表示了从文档的一种类型到另一种类型的演绎，检查是分析文档是否违反了一致性规则集，解析是通过从未知的术语到已知的术语的转换解析一个查询请求"。

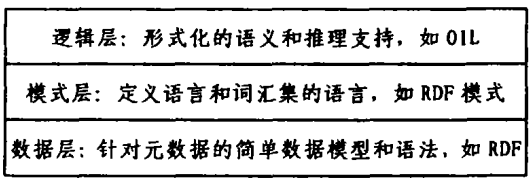


图2　语文化 Web 的三层结构

目前 W3C 还没有关于第三层的标准和草案，基于领域模型的 OIL 设计目的就是提供该层的功能服务。关于领域模型，RDFS 做出了两个重要的贡献：编写领域模型的标准化的语法、一组标准的类似于"instance-of"和"subclass-of"关系的建模原语。因此我们认为，语义化 Web 的第三层标准一定是在 RDFS 基础上进行扩展，提供形式化的语义和推理支持。

3　Ontology 与 OIL

3.1　领域模型及其应用

领域模型论是哲学中形而上学的一个分支，它研究事物存在的性质。领域模型近年来被 AI 采用，系指一系列的概念和术语抽象[13,14]。这种抽象可用来描述知识的某个方面或者建立知识的一种表达。一个领域模型可能是很高层次的抽象，这种抽象由一些概念组成，这些概念形成一个知识库的高层

维普资讯 http://www.cqvip.com

抽象部分；它也可能是特定领域的抽象，例如交通工具的领域模型。在特定知识领域内，领域模型论就是关于对象种类、对象特性以及对象之间可能存在的关系的内容理论，它提供描述该领域知识的抽象术语。

领域模型论在 AI 中主要指表示词汇或描述某个领域的知识体。表示词汇的领域模型通常专用于某个领域或主题，它可能包括描述概念上的元素词汇和这些元素之间的关系，确定这种词汇及背后的概念化常常需要对领域内的对象和关系类型进行仔细的分析，而知识体的领域模型通常利用表示词汇描述具有共同知识的领域。

由于跨越不同的系统进行知识重用或共享越来越重要，领域模型论越来越受到关注。因为不同的系统使用不同的概念和术语来描述不同的领域，而这些不同使得从一个体系获得的知识很难应用到另一个体系，所以这种不同就成为知识共享的主要障碍。如果能够开发出领域模型，使它成为一种多种体系的基础，通过领域模型来共享共同的术语，便能方便地实现共享和重用。有了共享的领域模型，我们能够建立描述指定情况的指定知识库，如：不同的电子设备制造厂可以使用共同的词汇和语法来建立描述自己产品的目录，这样制造厂可以共享目录，并且用它们来自动设计系统，这样共享将大大提高知识重用的潜力。

开发一种可以重用的领域模型是领域模型论研究的一个重要的目标。类似地，如果我们可以发展形成一种工具，它可以支持合并领域模型，并且使它们之间可以相互转化，那么就可能在不同领域模型基础体系之上实现共享。

3.2 领域模型语言

领域模型论是关于某个领域描述的形式化理论，因此需要一种形式化的逻辑语言进行表达。随着 Web 技术和知识工程研究的发展，很多语言标准可以用作领域模型的描述语言。

传统的领域模型描述语言包括了 CycL、KIF、Ontolingua、框架逻辑、描述逻辑等。其中 CycL 和 KIF 是扩展了的一阶谓词逻辑语言；而 Ontolingua 和框架逻辑是基于框架的方法，两种方法都把基于框架的建模原语结合到一阶逻辑框架中；描述逻辑根据概念和角色约束描述知识，角色用来推理出分类结果。

在现有的基于 XML 的 Web 平台中，XMLS、RDFS 可以用作表示领域模型的语言，或者说至少可以表示领域模型的某些方面。OntoKnowledge 在现有的领域模型语言和 RDFS 基础上，提出了一种三层结构的领域模型语言 OIL。XML 可作为 OIL 的表示语法使用，这样针对 XML 开发的确认和释义技术可直接用于用 OIL 定义的领域模型。RDFS 本身就是一种能定义基本词汇的简单的领域模型语言，然而，由于 XML 的 DTD 和 XMLS 缺乏对 OIL 定义的领域模型的语义支持，所以 OIL 与 RDF/RDFS 之间的关系要比 OIL 与 XML/XMLS 之间的关系密切得多。

3.3 OIL 的基础及其分层结构

针对语义化 Web 对机器可理解的语义表达的要求，OIL 可以表示 Web 知识的领域模型并进行推理。作为一个建议，OIL 结合了广泛使用的基于框架语言的模型原语和描述逻辑提供的推理服务，同时 OIL 与 RDFS 兼容，并包含一个精确的语义来描述术语的含义。

OIL 综合了来自三个主要不同研究机构的工作，为语义化 Web 提供一种通用的标记语言。OIL 三个主要的来源是描述逻辑、基于框架的系统、基于 XML 与 RDF 语法的 Web 语言标准[6,7]，它们与 OIL 之间的关系如图3所示。

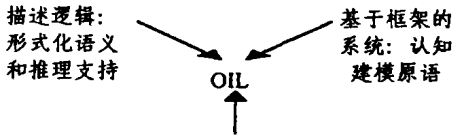


图3 OIL 的来源及关系

OIL 本身也采用了分层的方法定义标准的领域模型语言，如图4所示。其中上层在下一层基础上增加了功能和复杂性，这样当领域模型按任何一层的语义进行描述，即使只能处理和理解低层语义的主体（人或机器），也能够部分理解高层的领域模型描述。OIL 的各层功能简单介绍如下：

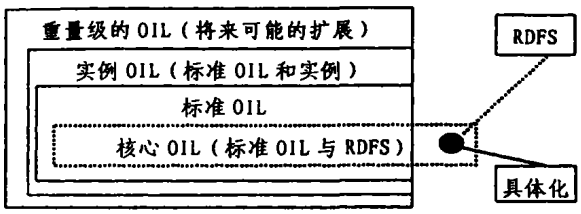


图4 OIL 的分层结构

核心 OIL 主要同 RDF 模式一致，除了 RDF 的具体化特性。这就意味着即使是简单的处理 RDF 模式的主体也能够处理 OIL 的领域模型描述，并尽可能多地理解 OIL 领域模型描述的语义；

标准的 OIL 是用来定义必要的主流建模原语，这些原语提供足够的表达能力，并且可以被很好地理解，因此使得语义可以被精确地定义，完整的推导可以实现；

实例 OIL 纯粹是具体的集成，因为前一层包括了建模构件，构件允许具体的填充器在术语定义中定义，所以实例 OIL 具有一种成熟的数据库能力；

重量级的 OIL 在将来进行扩展，可能具有额外的表示和推理能力。

用 OIL 定义一个应用的领域模型，通常包括了领域模型容器和领域模型定义，其中面向应用的抽象概念描述是领域模型定义，领域模型容器是领域模型定义的元数据描述，采用 DC 定义的元数据词汇集中的元素。机器可理解的领域模型表示对语义化 Web 的许多应用至关重要，例如 OIL 在搜索引擎、电子商务和知识管理等方面具有广阔的应用前景。

3.4 OIL 对 RDFS 的扩展分析

结合领域模型理论，OIL 对 RDFS 建模原语的扩展主要集中在类扩展定义、槽定义两个方面，同时类之间的约束关系通过槽约束表示，槽的定义通过 RDFS 的属性扩展定义实现。在 OIL 中，一个类的定义把一个类同一个名字、一个文档、一个类型、类的超类、为该类定义的属性联系在一起。在 RDFS 中，类只是通过给它们一个名字（具有 ID 属性）进行声明。OIL 类的定义通过 RDFS 构造机制和扩展 RDFS 按 RDF 数据模型进行。图5示出了 OIL 对 RDFS 扩展的包容结构关系。

OIL 引入了一个保留类 oil:ClassExpression。在 OIL 中，一个类可以说成是某个类表达式的子类，类表达式是类的一个布尔表达式。为了表示类表达式，OIL 引入了三种布尔运算符的类：与（AND）、或（OR）、非（NOT），它们均被定义为类 oil:ClassExpression 的子类。在类表达式中，由于布尔运算符应用的结果是一个类的定义，所以这样从建模角度来说也有

维普资讯 http://www.cqvip.com

意义。

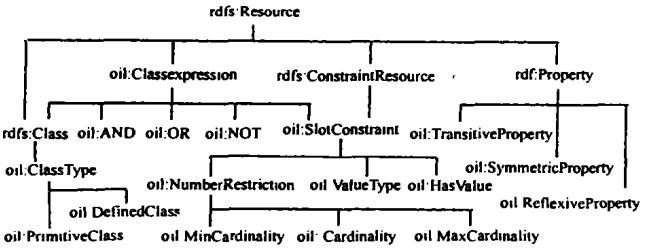


图5 包容结构关系中 OIL 对 RDFS 的扩展

运算符 AND、OR 和 NOT 通过利用 oil:hasOperand 属性同运算符的操作数连接起来。oil:hasOperand 属性只用于具体的类表达式(AND、OR 和 NOT 运算)的事实陈述,事实用 rdfs:domain 构造机制进行表达,这种类型的建模直接源自于 RDF 的以属性为中心的方法。

为了表示槽约束关系,OIL 引入了 oil:SlotConstraint 属性作为一个保留类,该属性是 rdfs:ConstraintProperty 的一种 rdf:type(类似于 rdfs:domain 和 rdfs:range)。而 RDFS 在本地的级别上没有约束类属性的机制,这主要是因为 RDF 数据模型以属性为中心的本性决定,而属性是全程定义的。引入了 oil:SlotConstraint 作为保留类,针对槽约束的某些特定的类,如 has-value,value-type,cardinality 等等,都是作为用 RDFS 建模的 oil:hasSlotConstraint 子类。

OIL 和 RDFS 都允许槽作为一个 Ontology 的首类成员,因此,用 OIL 定义的槽可以很好地映射到 RDFS 中的属性定义。比起 RDFS 中属性的定义,在 OIL 中的全程槽定义允许规定一个槽的多个方面。除了 domain 和 range 约束外,OIL 的槽还可以具有"inverse(逆)"属性和类似于"transitive(传递)"和"symmetric(对称)"的特性。因此可以引入一个"inverseRelationOf(逆关系)"的属性具有"rdf:Property"属性的 domain 和 range 约束,并引入"TransitiveRelation(传递关系)"和"SymmetricRelation(对称关系)"类反映一个槽的不同的特性。

通过 OIL 对 RDFS 的扩展,可以获得足够的表达能力,将任何 OIL 类的定义翻译为 RDF 语法,这样就最大限度地利用了 RDF(S)的原语和构件而没有牺牲用 OIL 定义的概念抽象的清晰度,使得不理解 OIL 的 RDF 代理(agent)尽可能多地理解用 OIL 描述的概念定义,同时还能够将 RDF 代理解析后的内容无歧义地翻译到 OIL 描述。

4 命题逻辑的扩展与推理规则表示

4.1 命题逻辑的扩展定义

为了利用 OIL 表示命题逻辑公式和相应的推理规则,我们在 OIL 的基础上对 RDFS 进行扩展。首先引入保留的类 oile:RuleBase 和 oile:Formula,其中"oile"是对 OIL 扩展定义类模式的名字空间。在类 oile:Formula 基础上定义两种命题运算 oile:IMP(蕴含)和 oile:EQU(等价)。在类 oile:IMP 基础上,引入了类 oile:Premise 和 oile:Conclusion,同时在类 oile:RuleBase 基础上引入类 oile:INF 表示"推导出",这样我们可以得到对 OIL/RDFS 扩展后的包容结构关系如图6所示。这些引入的类进行 RDF 模式定义如图7所示。在命题逻辑中,"蕴含"和"等价"可以通过三个最基本的运算进行化归表示,但通常这种化归表示只适合人为地进行,所以我们就直接引入了对应于这两种运算的类。

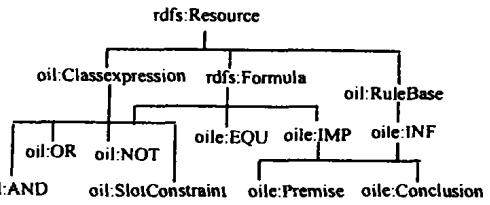


图6 对 OIL/RDF 扩展后的包容结构关系

```
<rdfs:Class rdf:ID="AND">
    <rdfs:comment>An expression corresponding to the conjunction of
        (two) class expressions</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#ClassExpression"/>
    <rdfs:subClassOf rdf:resource="# Formula"/>
</rdfs:Class>
<rdfs:Class rdf:ID="OR">
    <rdfs:comment>An expression corresponding to the disjunction of
        (two) class expressions</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#ClassExpression"/>
    <rdfs:subClassOf rdf:resource="# Formula"/>
</rdfs:Class>
<rdfs:Class rdf:ID="NOT">
    <rdfs:comment>An expression corresponding to the negation of
        a class expression</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#ClassExpression"/>
    <rdfs:subClassOf rdf:resource="# Formula"/>
</rdfs:Class>
<rdfs:Class rdf:ID="IMP">
    <rdfs:comment>An expression corresponding to the implication of
        a class expression</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#ClassExpression"/>
    <rdfs:subClassOf rdf:resource="# Formula"/>
</rdfs:Class>
<rdfs:Class rdf:ID="EQU">
    <rdfs:comment>An expression corresponding to the equivalence of
        a class expression</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#ClassExpression"/>
    <rdfs:subClassOf rdf:resource="# Formula"/>
</rdfs:Class>
<rdfs:Class rdf:ID="INF">
    <rdfs:comment>An inferening expression corresponding to premise
        and conclusion </rdfs:comment>
    <rdfs:subClassOf rdf:resource="#RuleBase"/>
</rdfs:Class>
<rdfs:Class rdf:ID="Premise">
    <rdfs:comment>An formula expression corresponding to premise
        </rdfs:comment>
    <rdfs:subClassOf rdf:resource="#IMP"/>
</rdfs:Class>
<rdfs:Class rdf:ID="Conlclusion">
    <rdfs:comment>An formula expression corresponding to conclusion
        </rdfs:comment>
    <rdfs:subClassOf rdf:resource="#IMP"/>
</rdfs:Class>
<rdf:Class rdf:ID="Formula">
    <rdfs:comment>A user-defined formula corresponding to a descri-
        ption of proposition by an external RDF-Schema</rdfs:comment>
    <rdfs:subClassOf rdf:resource="http://www.w3.org/TR/1999/PR-rdf-
        schema-19990303#Resource"/>
</rdfs:Class>
```

图7 命题逻辑的 RDF 模式扩展定义

4.2 命题逻辑的推理公式表示示例

基于 OIL 定义的 RDFS 扩展和我们在 OIL 基础上的扩展,我们不仅可以表示绝大多数的命题公式,而且还可以描述推理规则和规则库。我们选取如下通用的推理规则作为示例:

$$P_1, P_2, \cdots, P_m \vdash Q_1, Q_2, \cdots, Q_n$$

得到该推理公式的表示如图8所示。

```
〈oile:INF〉
    〈oile:Premise〉
        〈oil:AND〉
            〈oil:hasOperand rdf:resource="#P1"/〉
            〈oil:hasOperand rdf:resource="#P2"/〉
                ……
            〈oil:hasOperand rdf:resource="#Pm"/〉
        〈/oil:AND〉
    〈/oile:Premise〉
    〈oile:Conclusion〉
        〈oil:OR〉
            〈oil:hasOperand rdf:resource="#Q1"/〉
            〈oil:hasOperand rdf:resource="#Q2"/〉
```

维普资讯 http://www.cqvip.com

```
        ......
            〈oil:hasOperand rdf:resource="#Qn"/〉
        〈/oil:OR〉
    〈/oile:Conclusion〉
〈/oile:INF〉
```

图8 基于OIL及其扩展的推理公式表示示例

结论 XML的出现推动Web-NG的发展，语义化Web可以建立在XML/XMLS上，也可以建立在RDF/RDFS上，我们认为基于RDF/RDFS实现语义的互操作比基于XML/XMLS更具有优越性。关于Web资源的元数据处理除了需要元数据的语义表示模型和原语外，还需要先进的方法和技术，KE在领域模型及其它方面的研究成果可以为语义化Web带来新的动力。

本文介绍了语义化Web的关于语义建模的一种主要方法和建模原语规范——RDF(S)、语义化Web的分层结构，并且介绍了KE研究中的热点技术领域模型和相应的建模方法。在此基础上介绍了Web技术和领域模型建模方法结合的一种产物——OIL，分析了OIL对RDFS建模原语的扩展。基于OIL/RDFS定义，我们面向命题逻辑公式和推理规则的表示作了进一步的扩展，通过示例表明基于这种扩展可以表示命题公式和推理规则，进而面向应用可以构成规则库。

在本文的基础上，我们将进行以下两个方面的研究：在理论上，我们可以把公理视为对象，基于OIL/RDF(S)进行命题逻辑和谓词逻辑的公理化研究，同时表示Horn子句和Datalog等定义规则库模式；在具体应用方面，我们可以借用领域模型的方法建立基于Web-NG的网络管理、目录服务、智能信息集成、数据挖掘与知识发现等应用基础。

参考文献

1 W3C. PICS Label Distribution Label Syntax and Communication Protocols, Version 1.1. W3C Recommendation, Oct. 1996. http://www.w3.org/TR/REC-PICS-labels

2 Miller E. An Introduction to the Resource Description Framework. D-Lib Magazine, May. 1998

3 Horrocks I, et al. The Ontology Inference Layer OIL. http://www.ontoknowledge.org/oil.

4 Bray T, et al. Extensible Markup Language (XML) 1.0. W3C Recommendation, World Wide Web Consortium, 1998. http://www.w3c.org/TR/REC-xml

5 (美)Didier Martin 等. XML高级编程. 机械工业出版社，2001

6 Biron P V, et al. XML-Schema. W3C Recommendation, 2001. http://TR/REC-xml-schema

7 Lassila, Swick R R. Resource Description Framework (RDF) Model and Syntax. W3C Recommendation, World Wide Web Consortium, Feb. 1999. http://www.w3c.org/TR/REC-rdf-syntax

8 Brickley D, Guha R V. Resource Description Framework (RDF) Schema Specification. W3C Recommendation, World Wide Web Consortium, Mar. 2000. http://www.w3c.org/TR/REC-rdf-schema

9 姚绍文，刘爱莲，等. 基于XML的Web自动化技术. 中国XML技术2000研讨会，北京，2000

10 姚绍文，周明天，等. 下一代的Web技术. 计算机科学，2001，28(1)：78～84

11 Berners-Lee T. Semantic web road map. World Wide Web Consortium, Sept. 1998. http://www.w3c.org/DesignIssues/Semantic.html

12 Studer R, et al. Situation and Perspective of Knowledge Engineering. In: Knowledge Engineering and Agent Technologies. IOS Press, 2000

13 Swartout W. Ontologies. IEEE Intelligent Systems, 1999(Jan.)：18～19

14 Chandrasekaran B, et al. What are ontologies, and why do we need them. IEEE Intelligent System, 1999(Jan.)：20～26

15 Lassila O. Web Metadata: A Matter of Semantics. IEEE Internet Computing, 1998(Jul.-Aug.)：30～37

(上接第54页)

4 Shi S, Waldvogel M. A Rate-Based End-to-End Multicast Congestion Control Protocol. 2000

5 Rubenstein D, Kurose J, Towsley D. The Impact of Multicast on Network Fairness. In: ACM SIGCOMM'99 8/99, MA USA

6 Delucia D, Obraczka K. A Multicast Congestion Control Mechanism for Reliable Multicast. In: Proc. IEEE INFOCOM, 1997

7 Mathis M, et al. SBMCC: A Generic Source-based Congestion control Algorithm for Reliable Multicast. Feb. 2000. Available at: http://www.ecse.rpi.edu/Homepages/shivkuma/research/papers/sbmcc2.ps

8 Golestani S J. Fundamental Observations on Multicast Congestion Control in the Internet. In: Proc. IEEE INFOCOM, 1999

9 Roca V. On the Use of On-Demand layer Addition (ODL) with Multi-Layer Multicast Transmission Techniques. WORK IN PROGRESS, June 2000

10 Vicisano L, Rizzo L, Crowcroft J. TCP-like Congestion Control for Layered Multicast Data Transfer. In: IEEE INFOCOM'98, Feb. 1998

11 Montgomery T A. Loss Tolerant Rate Controller for Reliable Multicast: [Tech Rep: NASA-IVV-97-011]. West Virginia Univ.

12 Sisalem D, Wolisz A. MLDA: A TCP-Friendly Congestion Control Framework for Heterogeneous Multicast Environments. In: Eighth Intl. Workshop on Quality of Service (IWQoS 2000), Pittsburgh, PA, June 2000

13 Lin J C. RMTP: A Reliable Multicast Transport Protocol. In: Proc. of IEEE Computer Society Press, 1996. 1414～1424

14 Whetten B, Montgomery T, Kaplan S. A High Performance Totally Ordered Multicast Protocol. In Theory and Pratice in Distributed Systems. LNCS 938, Spinger Verlag

15 Rhee I, Balaguru N, Rouskas G N. MTCP: Scalable TCP-like Congestion Control for Reliable Multicast. In: Proc. of the INFOCOM, March 1999

16 Donahoo M J, Ainapure S R. Scalable Multicast Representative Member Selection. In: IEEE INFOCOM 2001

17 Lacher M S, Nonnenmacher J, Biersack E W. Performance Comparison of Centralized Versus Distributed Error Recovery for Reliable Multicast. IEEE/ACM Transactions on Networking, 2000, 8(2)：224～238

18 Thapliyal P, et al. LE-SBCC: Loss-Event Oriented Source-Based Multicast Congestion Control. submitted to NOSSDAV 2001

19 Basu A, Golestani S J. Estimation of Receiver Round Trip Times in Multicast Communications. RMRG Meeting, Dec. 1998

20 Yano K, McCanne S. A Window-Based Congestion Control for Reliable Multicast Based on TCP Dynamics. In: ACM SIGMM. Electronic Proceedings (Available from: http://woodworm.cs.uml.edu/～rprice/ep/yano/)

21 Legout A, Biersack E W. Fast Convergence for Cumulative Layered Multicast Transmission Schemes. In: Proc. of SIGMETRICS'2000, Santa Clara, CA, July 2000