

1 绪 论

1.1 我国煤矿安全现状

我国化石能源的基本特点是富煤、贫油、少气,据《中国统计年鉴 2011》基础储量数据,至 2010 年,我国煤炭基础储量为 2 793.9 亿 t,石油为 317 435.3 万 t,天然气为 37 793.2 亿 m³。煤炭占一次性化石能源基础储量的 96.6%(见图 1-1)。这就决定了我国在相当长一段时间内以煤炭为主的能源生产及消费结构不会发生根本性改变。

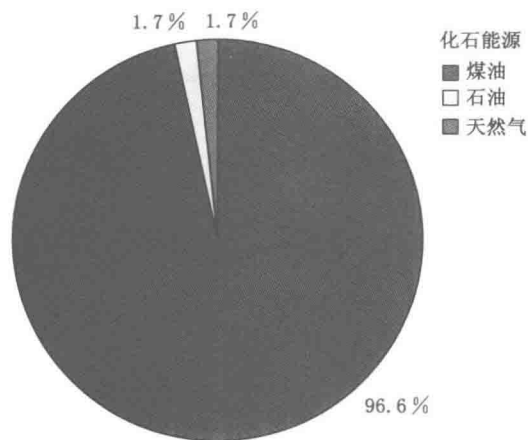


图 1-1 我国化石能源储量结构

但长期以来我国煤矿安全事故频发,给国家和人民群众造成了巨大的生命与财产损失,这与以人为本的科学发展观和构建社会主义和谐社会的重大战略指导思想是相违背的。图 1-2 显示了我国 2002~2011 年的历年煤矿死亡人数。

尽管近几年由于增加安全投入、加大关闭重组小煤窑力度、提高煤矿企业安全管理水平等,使得煤矿死亡人数呈下降趋势,但和国外相比仍然存在较大差距。

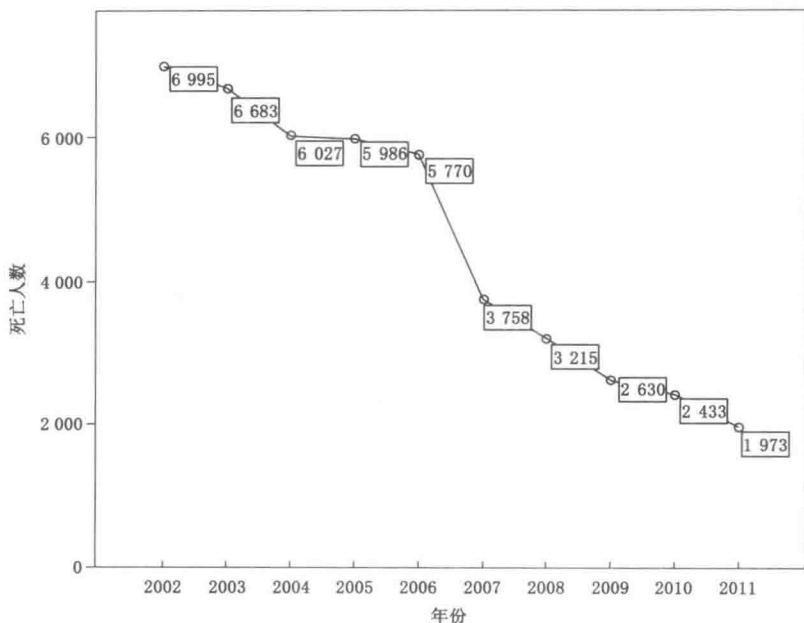


图 1-2 我国煤矿事故死亡人数

1.2 物联网技术的发展

1.2.1 起源与发展

物联网的实践最早可以追溯到 1990 年施乐公司的网络可乐贩售机——Networked Coke Machine。

1991 年,美国麻省理工学院(MIT)的 Kevin Ashton 教授首次提出物联网的概念。

1995 年,比尔盖茨在《未来之路》一书中也曾提及物联网,但未引起广泛重视。

1999 年,美国麻省理工学院建立了“自动识别中心(Auto-ID)”,提出“万物皆可通过网络互联”,阐明了物联网的基本含义。早期的物联网是依托射频识别(RFID)技术的物流网络,随着技术和应用的发展,物联网的内涵已经发生了较大变化。

2003 年,美国《技术评论》提出传感网络技术将是未来改变人们生活的十大技术之首。

2004年,日本总务省(MIC)提出 u-Japan 计划,该战略力求实现人与人、物与物、人与物之间的连接,希望将日本建设成一个随时、随地、任何物体、任何人均可连接的泛在网络社会。

2005年11月17日,在突尼斯举行的信息社会世界峰会(WSIS)上,国际电信联盟(ITU)发布《ITU 互联网报告 2005:物联网》,引用了“物联网”的概念。物联网的定义和范围已经发生了变化,覆盖范围有了较大的拓展,不再只是指基于 RFID 技术的物联网。

2006年,韩国于确立了 u-Korea 计划,该计划旨在建立无所不在的社会(ubiquitous society),在民众的生活环境里建设智能型网络(如 IPv6、BcN、USN)和各种新型应用(如 DMB、Telematics、RFID),让民众可以随时随地享有科技智慧服务。2009年,韩国通信委员会出台了《物联网基础设施构建基本规划》,将物联网确定为新增长动力,提出到 2012 年实现“通过构建世界最先进的物联网基础设施,打造未来广播通信融合领域超一流信息通信技术强国”的目标。

2008年后,为了促进科技发展,寻找经济新的增长点,各国政府开始重视下一代的技术规划,将目光放在了物联网上。在中国,同年11月在北京大学举行的第二届中国移动政务研讨会“知识社会与创新 2.0”提出移动技术、物联网技术的发展代表着新一代信息技术的形成,并带动了经济社会形态、创新形态的变革,推动了面向知识社会的以用户体验为核心的下一代创新(创新 2.0)形态的形成,创新与发展更加关注用户、注重以人为本。而创新 2.0 形态的形成又进一步推动新一代信息技术的健康发展。

2009年,欧盟执委会发表了欧洲物联网行动计划,描绘了物联网技术的应用前景,提出欧盟政府要加强对物联网的管理,促进物联网的发展。行动方案的主要内容:

- (1) 加强物联网管理;
- (2) 完善隐私和个人数据保护;
- (3) 提高物联网的可信度、接受度和安全性;
- (4) 评估现有物联网的有关标准并推动新标准的制定;
- (5) 推进物联网方面的研发;
- (6) 通过欧盟竞争力和创新框架计划(CIP)推动物联网应用;
- (7) 加强对物联网发展的监测、统计和管理等。

2009年1月28日,奥巴马就任美国总统后,与美国工商业领袖举行了一次“圆桌会议”,作为仅有的两名代表之一,IBM 首席执行官彭明盛首次提出“智慧地球”这一概念,建议新政府投资新一代的智慧型基础设施。当年,美国将新能

源和物联网列为振兴经济的两大重点。

2009年2月24日,2009IBM论坛上,IBM大中华区首席执行官钱大群公布了名为“智慧的地球”的最新策略。此概念一经提出,即得到美国各界的高度关注,甚至有分析认为IBM公司的这一构想极有可能上升至美国的国家战略,并在世界范围内引起轰动。IBM认为,IT产业下一阶段的任务是把新一代IT技术充分运用在各行各业之中。在策略发布会上,IBM还提出,如果在基础建设的执行中植入“智慧”的理念,不仅能够短期内有力地刺激经济、促进就业,而且能够在短时间内为中国打造一个成熟的智慧基础设施平台。IBM希望“智慧的地球”策略能掀起互联网浪潮之后的又一次科技产业革命。IBM前首席执行官郭士纳曾提出一个重要的观点,认为计算模式每隔15年发生一次变革。这一判断像摩尔定律一样准确,人们把它称为“十五年周期定律”。

而今天,“智慧地球”战略被不少美国人认为与当年的“信息高速公路”有许多相似之处,同样被他们认为是振兴经济、确立竞争优势的关键战略。该战略能否掀起如当年互联网革命一样的科技和经济浪潮,不仅为美国所关注,更为世界所关注。

2009年8月,温家宝总理“感知中国”的讲话把我国物联网领域的研究和应用开发推向了高潮,无锡市率先建立了“感知中国”研究中心,中国科学院、运营商、多所大学在无锡建立了物联网研究院。自“感知中国”提出以来,物联网被正式列为国家五大新兴战略性产业之一,写入“政府工作报告”,物联网在中国受到了全社会极大的关注,其受关注程度是在美国、欧盟以及其他各国不可比拟的。

1.2.2 物联网的定义

物联网是新一代信息技术的重要组成部分。其英文名称是“The Internet of things”。顾名思义,“物联网就是物物相连的互联网”。其定义包含两层意思:第一,物联网的核心和基础仍然是互联网,是在互联网基础上延伸和扩展的网络;第二,其用户端延伸和扩展到了任何物品与物品之间,进行信息交换和通信。物联网是互联网的应用拓展。物联网就是“物物相连的互联网”。物联网是通过智能感知、识别技术与普适计算、泛在网络的融合应用。

中国物联网校企联盟将物联网定义为当下几乎所有技术与计算机、互联网技术的结合,实现物体与物体之间、环境以及状态信息的实时共享以及智能化的收集、传递、处理、执行。

物联网通过各种信息传感设备,实时采集任何需要监控、连接、互动的物体或过程等各种需要的信息,与互联网结合形成的一个巨大网络。其目的是实现物与物、物与人,所有的物品与网络的连接,方便识别、管理和控制。

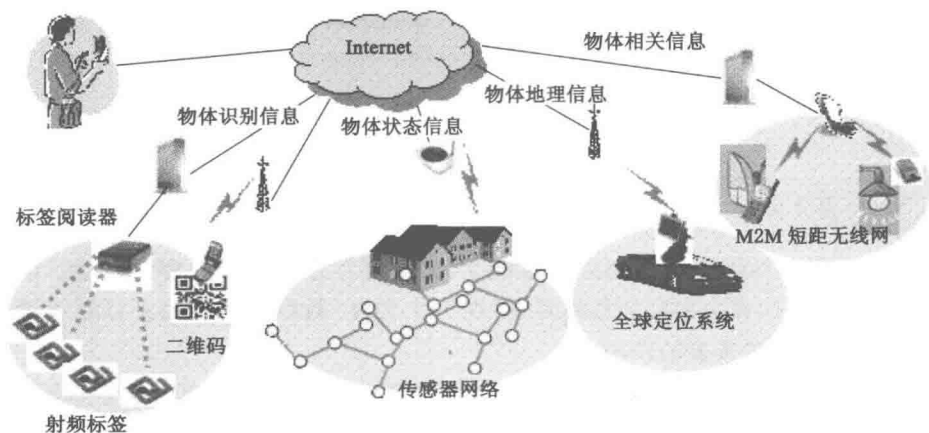


图 1-3 物联网示意图

物联网示意图如图 1-3 所示。

1.2.3 物联网技术架构

物联网技术架构(见图 1-4)分为三层:感知层、网络层和应用层。



图 1-4 物联网技术架构

物联网感知层由各种传感器构成,包括温湿度传感器、二维码标签、RFID 标签和读写器、摄像头、GPS 等感知终端。感知层是物联网识别物体、采集信息的来源。

网络层由各种网络,包括互联网、广电网、网络管理系统和云计算平台等组成,是整个物联网的中枢,负责传递和处理感知层获取的信息。

应用层是物联网和用户的接口,它与行业需求结合,实现物联网的智能应用。

1.3 本体论

1.3.1 基本概念

本体论(Ontology)最初是哲学概念,是研究存在的本质的哲学问题。但在最近几十年里,被应用到计算机界,并在人工智能、数据库理论以及计算机语言中扮演着越来越重要的作用。

然而,计算机界对 Ontology 的理解也是逐步发展才走向成熟的。1991年, Neches 等人最早给出 Ontology 在信息科学中的定义:“给出构成相关领域词汇的基本术语和关系,以及利用这些术语和关系构成的规定这些词汇外延规则的定义”。后来在信息系统、知识系统等领域,随着越来越多的人研究 Ontology,产生了不同的定义。1993年,Gruber 定义 Ontology 为“概念模型的明确的规范说明”。1997年,Borst 进一步完善为“共享概念模型的形式化规范说明”。Studer 等人对上述两个定义进行了深入研究,认为 Ontology 是共享概念模型的明确的形式化规范说明,这也是目前对 Ontology 概念的统一看法。

Studer 等人的 Ontology 定义包含四层含义:概念模型(conceptualization)、明确(explicit)、形式化(formal)和共享(share)。“概念模型”是指通过抽象出客观世界中一些现象(phenomenon)的相关概念而得到的模型,其表示的含义独立于具体的环境状态;“明确”是指所使用的概念及使用这些概念的约束都有明确的定义;“形式化”是指 Ontology 是计算机可读的,也就是计算机可处理的;“共享”是指 Ontology 中体现的是共同认可的知识,反映的是相关领域中公认的概念集,它所针对的是团体而非个体。Ontology 的目标是捕获相关领域的知识,提供对该领域知识的共同理解,确定该领域内共同认可的词汇,并从不同层次的形式化模式上给出这些词汇(术语)和词汇之间相互关系的明确定义。

尽管定义有很多不同的方式,但是从内涵上来看,不同研究者对于 Ontology 的认识是统一的,都把它当作是领域(领域的范围可以是特定应用中,也可以是更广的范围)内部不同主体(人、机器、软件系统等)之间进行交流(对话、互操作、共享等)的一种语义基础,即由 Ontology 提供一种共识。而且 Ontology 提供的这种共识更主要的是为机器服务,机器并不能像人类一样理解自然语言中表达的语义,目前的计算机也只能把文本看成字符串进行处理。因此,在计算机领域讨论 Ontology,就要讨论如何表达共识,也就是概念的形式化问题。Ontology 这个哲学范畴,被人工智能界赋予了新的定义,从而被引入信息科学中。

本体的构建需要完备的领域知识、明确统一的视点、良好的本体构建方法,以及合适的本体建模语言和工具。

1.3.2 本体建模准则

1993年,T. R. Gruber提出了5条被广泛接受的本体建模准则,其主要内容如下:

(1) 明确性和客观性:本体应该明确地传达所定义的术语内涵。术语定义应该具有客观性。

(2) 一致性:一个本体应该前后保持一致,尤其是推断出的隐性概念定义应与本体中显示的概念定义一致,至少所有的公理具有逻辑一致性。

(3) 可扩展性:一个本体提供的共享词汇应该在预期的范围内提供形成新概念的基础,使得能够在不改变原有定义的情况下,以存在的词汇为基础定义新的词汇。

(4) 最小编码偏差:本体应该处于知识的层次,而与特定的符号编码无关,它表现形式的选择不应该只考虑表示上或形式上的方便。本体的编码偏差应该控制在尽可能小的范围内,因为本体在哲学的含义上就是语言无关的,在人工智能领域本体提供了可共享的词汇,也应该独立于任何知识表示系统和风格。

(5) 最小本体承诺:一个本体应该在提供必需的共享知识的条件下,要求有最小本体承诺。即它应该对描述的事物产生尽可能少的推断,而让共享者自由地按照其需要去专门化和实例化该本体。

除了上述5条原则外,其他的一些研究人员根据自身实践情况,也提出了一些本体建模准则,如J. Arpirez等提出的3条规则:

(1) 概念名称命名标准化:尽可能使用标准规范术语。

(2) 语义距离最小化:同层次概念间保持最小的语义距离,尽可能把含义相似的概念抽象出来。

(3) 可以使用多种概念层次、采用多重继承机制增加表达能力。

1.3.3 本体工程

传统的知识工程着重于快速地开发包含知识获取、概念构建和表示模型的一组技术和工具。本体工程是知识工程的一个分支,它研究本体构建的方法学和方法论。本体工程涉及本体开发过程、本体生命周期、本体构建的方法学、适合本体开发的工具和支持本体构建的语言。

本体工程的目标是为了更加有效地开发、管理和使用本体提供指导方针。为加速本体开发和提供本体开发项目的管理,本体工程的研究着重于提供明确

的方法指导以帮助软件工程师和本体开发者构建本体,同时也提供必要的技术支持。

本体工程主要专注于下列过程和活动:本体需求说明;计划和规划;控制、重用和非本体资源(如词典、分类表)的业务流程再设计;本体和本体设计模式的重用;规模本体协作式开发;本体建模;本体进化;多种语言本体间的映射;本体集成;本体评价等。

本体的构建是为了共享和重用,而本体构建的主观性较强,因此与一般系统的构建相比,本体构建的工程性更为明显。本体工程强调构建本体时要遵循一定的规范和标准,但因为本体工程目前仍处于不成熟的阶段,还没有标准化的方法、管理和制约,所以本体工程领域通常使用软件生命周期法(IEEE Std 1074—2006)作为参照标准对本体构建方法论的成熟度进行评价。

1.3.4 本体建模方法

本体建模主要分为两种类型:①从知识工程的角度进行本体构建,称为本体工程;②利用直接将现有的词汇表转化为本体进行建模。

典型的本体构建方法有:KACTUS、SENSUS法,结构化建模法(METH-ONTOLOGY),骨架法,TOVE法,IDEF5法,七步法等。每种建模方法各有特点及优缺点。

IEEE Std 1074—2006是软件开发生命周期的标准,并不是本体构建方法的标准。但是从广义上讲,本体构建方法也属于知识工程领域中的一种软件开发方法。将上述7种方法体系与IEEE Std 1074—2006标准进行比较,可以发掘现有本体构建方案作为软件开发方法的不完善之处,以及它们不同于一般软件开发方法的特别之处。

表 1-1 七种方法体系与 IEEE 标准的一致性比较

项目 名称	工程管 理阶段	开发 前期	需求 分析	设计	执行	开发后期	统一阶段
TOVE 法	没有	没有	有	有	有	没有	不全,没有训练、环境学习和配置管理
METHONTOLOGY	不全,没有建立工程环境阶段	没有	有	有	有	不全,没有安装、操作、支持、维护等阶段	不全,没有训练、环境学习

续表 1-1

项目 名称	工程管 理阶段	开发 前期	需求 分析	设计	执行	开发后期	统一阶段
骨架法	没有	没有	有	—	有	没有	不全, 没有训 练、环境学习和 配置管理
KACTUS 法	没有	没有	有	有	有	没有	没有
SENSUS 法	没有	没有	有	—	有	没有	没有
IDEF5 法	没有	没有	有	有	有	没有	不全
七步法	不全, 没有 建立工程 环境阶段	没有	有	有	有	不全	不全

表 1-2 七种方法体系的相互比较

项目 名称	生命周期	与 IEEE 标 准的一致性	相关技术	本体的应用	方法的细节
TOVE 法	不是真正的 生命周期	不完全一致	不确定	一个域	少
METHONTOLOGY	有	不完全一致	有, 不全	多个域	详细
骨架法	没有	不完全一致	不确定	一个域	很少
KACTUS 法	没有	不完全一致	不确定	一个域	很少
SENSUS 法	没有	不完全一致	不确定	多个域	一般
IDEF5 法	没有	不完全一致	不确定	多个域	详细
七步法	不是真正的 生命周期	不完全一致	有	多个域	详细

通过表 1-1 和表 1-2 的比较分析可知:

- (1) 与 IEEE 标准相比较, 没有一种方法体系是完全成熟的;
- (2) 七种方法均允许在系统间进行互操作;
- (3) 任何方法都提供知识共享和复用的机制;
- (4) 七种方法体系的成熟度依次为: 七步法 > METHONTOLOGY 法 > IDEF5 法 > TOVE 法 > 骨架法 > SENSUS 法 > KACTUS 法。

因此, 本书决定采用七步法构建煤矿事故智能预警本体。

1.3.5 本体建模语言

Ontology 成为研究热点还与语义网(Semantic Web)的提出与发展有着直接的关系。语义网是 Tim Berners-Lee 在 1998 年提出的一个概念。Tim Berners-Lee 认为,当前的 Web 是供人阅读和理解的,但并不便于实现数据和信息的自动化处理。新一代的语义网将不仅仅为人类所理解而且能为计算机(智能体)带来语义内容,使计算机(智能体)能够“理解”Web 内容,进而实现信息处理的自动化。他认为语义网不是与当前 Web 隔离的另一个 Web,而是对当前 Web 的扩充,在语义网中,信息的语义经过完好的定义,能够更好地促进计算机和人之间的相互合作。

为了实现语义网的功能,需要提供一种计算机能够理解的、结构化的语义描述机制,以及一系列的推理规则以实现自动化推理。语义网的挑战在于提供一种语言。它能够表述数据和在数据中进行推理,而且需要这种语言能够将目前存在于知识表述系统之中的规则应用到 Web 上。

在 Tim Berners-Lee 的语义网框架中,几个关键的组成元素是 XML、RDF(S) 和 Ontology。RDF(Resource Description Frame,资源描述框架)和 OWL(Web Ontology Language,本体描述语言)是 W3C(World Wide Web Consortium,万维网联盟)推荐的语义网的标准,为资源管理、企业数据整合、Web 的数据共享和重用提供了框架,使不同领域、不同类型的用户可以共享同样的信息。

使用本体建模语言可以为领域模型编写清晰的、形式化的概念描述。2004 年 2 月 W3C 发布了支持语义网络的两个技术标准:资源描述框架 RDF 和 Web 本体语言 OWL。

(1) RDF 和 RDF Schema

资源描述框架 RDF 定义了一种描述资源及其相互关系的简单模型,通过属性(Property)和值(Value)来描述资源以及资源与资源之间的关系,资源描述框架模型可以看作是<属性,值>对模型。

RDF 采用基于 XML 的语法,其表达式的结构是三元组集合,每个三元组包括一个主词(subject)、一个谓词(predicate)和一个宾词(object)。RDF 三元组的断言声明了由谓词表示的、在主词和宾词所指称的事物之间的关系。RDF 具有形式语义并提供推理功能,它是 OWL 的前身。

RDF Schema 是在 RDF 三元组描述事物的基础上的进一步扩展,添加了诸如 rdfs:Class, rdfs:subClassOf, rdfs:subPropertyOf, rdfs:domain, rdfs:range 等原语对类、父子类、父子属性以及属性的定义域和值域进行定义和表达,使得 RDF(S)成为一个能够对本体进行初步描述的标准语言。

(2) DL 与 OWL

描述逻辑 DL (Description Logic) 是人工智能领域研究和开发的一个重要的知识表示语言。“描述”指利用概念和规则构造符将原子概念(一元谓词)和原子规则(二元谓词)构建出描述表达式;“逻辑”指 DL 采用了正规的基于逻辑的语义,这与语义网络及框架等知识表示机制是不同的。

描述逻辑是基于对象的知识表示的形式化工具,是一阶谓词逻辑的可判定子集,提供可判定的推理服务,并且具有语义特征。其特性主要有:① 具有良好的语义定义和表达能力;② 保证计算推理的可判定性;③ 基于逻辑的推理能力;④ 有明确的推理算法,常用的有 Tableaux 算法;⑤ 支持诸如 FACT、RACER、Pellet 等的推理工具。

OWL 是一种基于描述逻辑 DL 的形式化本体描述语言,它使用一组限定的词汇来表达逻辑定义和关系,具有丰富的语义表达能力,并可进行合理、一致的推理。OWL 使用类 (Classes)、类的相关属性 (Properties) 和类的实例 (Individuals) 来表示组织知识关系,并使用公理 (Atoms) 和函数 (Function) 实现对语义推理的支持。

OWL 是 W3C 推荐的本体描述语言标准,它是在描述语言 DAML+OIL 上发展而来的,处于 W3C 提出的本体语言栈的最上层,如图 1-5 所示。

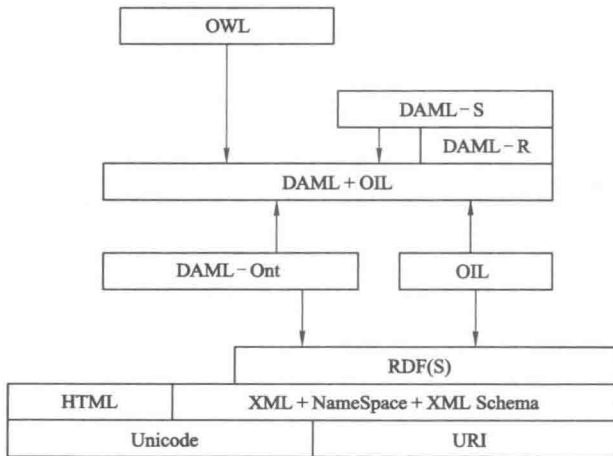


图 1-5 本体语言栈

根据表示和推理能力,W3C 制定了三种相应的 OWL 的子语言,即 OWL Lite、OWL DL 和 OWL Full,三种子语言的描述如表 1-3 所列。

表 1-3 OWL 的三种子语言

子语言	描述和推理能力
OWL Lite	支持概念(类)的分层体系和简单的约束,去除了枚举类、不交声明和任意基数约束,可以利用已经实现的逻辑推理系统进行推理
OWL DL	具有计算完备性(所有的结论都可计算)和可判定性(所有的计算都在有限的时间终止),支持有效的推理,但不具备和 RDF 的兼容
OWL Full	提供最强大的表达能力和使用 RDF 语法的最大自由,和 RDF 是完全兼容的,但不能保证计算性能,且由于表达能力太强,导致推理变得不可判定

从 OWL Lite、OWL DL 到 OWL Full 其表达描述能力逐步增强,推理复杂度也逐步提高。OWL Lite 只提供用户一个概念分类层次和简单约束;OWL DL 既保证推理的可判断性,又保证具有相当的表达描述能力;OWL Full 提供最强的表达能力和完全自由的 RDF 语法,但不保证计算性。

OWL DL 是一种基于描述逻辑的形式化本体描述语言。描述逻辑对于本体的推理操作主要有:① 实例自动归类(Automatic Classification/Subsumption);② 概念匹配/检索处理(Concept Matching/Query Processing);③ 概念一致性检验(Consistency Clustering),用于确定构建的本体在逻辑上是否是一致的。OWL DL 具备两种推理能力:一是可以推理某个个体是否是概念的实例,即进行 $A(a)$ 是否为真的推理;二是可以推理两个个体之间是否满足某关系,两个个体是否是属性 R 的定义域和值域概念的个体,即进行 $R(a,b) \rightarrow a \in A \& b \in B$ 是否为真的推理。

本书针对煤矿事故智能预警领域知识的表达和推理需求,采用 OWL DL 本体语言来形式化描述煤矿事故智能预警模型。

1.3.6 本体建模工具

本体建模工具,又称为本体论编辑器或者本体管理环境,是指设计用于辅助本体之创建、编辑、保存或处理等操作的应用程序。它们往往在众多的本体语言之中选用其一来表达本体。不过,有些本体编辑器还可以采用其他本体语言进行导出。尽管人们大都习惯于称此类工具为“本体编辑器”,而本体编辑器的功能实际上往往并不局限于本体的创建、编辑和保存。

目前存在大量的可以用来创建和操控本体的建模工具,考虑到研究内容的未来推广及现场应用,需要尽可能使用用户群广泛和功能可拓展的本体建模软件,拥有大量用户群及插件的主要有 Apollo、OntoStudio、Protégé、

Swoop and TopBraid Composer Free Edition。以下是这 5 个本体建模工具的比较(见表 1-4~表 1-9)。

表 1-4 工具的一般描述

特征	Apollo	OntoStudio	Protégé	Swoop	TopBraid Composer
开发者	KMI(Open University)	Ontoprise	SMI(Standord University)	MND(University of Maryland)	TopQuadrant
可用性	开源	软件授权	开源	开源	软件授权

表 1-5 软件架构和工具演化

特征	Apollo	OntoStudio	Protégé	Swoop	TopBraid Composer
语义网架构	单机	Eclipse 客户/服务器	单机和客户/服务器	基于 Web 的客户/服务器	单机 (Eclipse 插件)
可扩展性	支持插件	支持插件	支持插件	支持插件	支持插件
备份管理	无	无	无	无	有
本体存储	文件	关系数据库	文件和关系数据库	HTML 模型	关系数据库

表 1-6 互操作性

特征	Apollo	OntoStudio	Protégé	Swoop	TopBraid Composer
与其他本体工具	无	OntoAnnotate, OntoBroker, OntoMat, Semantic and Miner	PROMPT, OKBC, JESS, FaCT, Jena	无	Sesame, Jena, AllegroGraph
导入语言	Apollo Meta Language	L(s), OWL, Excel, RDF(S), UML2.0, databaseschemas (Oracle, MS-SQL, DB2, MySQL), Outlook Emails	XML(s), RDF(S), OWL, Html, (RDF, UML, XML) Backend, Text File, RDF File, Excel, BioPortal, DataMaster	OWL, XML, RDF, text famts	RDFa, WOL, XML(S), RDF(S), XHTML, UML, GRDDL, RDB, Oracle Database, Text File, RDF File, News Feed, Email, Excel

续表 1-6

特征	Apollo	OntoStudio	Protégé	Swoop	TopBraid Composer
输出语言	OCML, CLOS	OWL, RDF(S), RIF, SPARQL, F-Logic, Excel	XML(S), RDF(S), OWL, HTML, Java, Clips, SWRL-IQ, Instance Selection, MetaAnalysis, OWLDoc, Queries, (RDF, UML, XML) Backend	RDF(S), OIL, DAML	HTML, UML, XSD, Excel, RDB, Oracle Database, RDF File, XML File, Text File

表 1-7 知识表示和方法支持

特征	Apollo	OntoStudio	Protégé	Swoop	TopBraid Composer
知识模型的 KR 图表	框架(OKBC)	框架和一阶逻辑	框架、一阶逻辑、SWRL、元类	OWL	RDF, OWL, SWRL
公理语言	不严格	有(一阶逻辑)	有(PAL)	OWL DL	OWL DL
方法支持	无	支持(Onto-Knowledge)	无	无	无

表 1-8 相关推理服务

特征	Apollo	OntoStudio	Protégé	Swoop	TopBraid Composer
内嵌推理引擎	否	是(Ontobroker)	是(PAL)	否	WOL, SPARQL, Rule
其他可附加推理引擎	否	否	RACER, FACT, FACT++, F-logic, Pellet	Pellet, RDF-	OWLIM, Pellet, Jena Rules, Oracle Rules, SPARQL Rules
约束/一致性检测	是	是	是	仅由推理插件提供	是

表 1-9 工具可用性

特征	Apollo	OntoStudio	Protégé	Swoop	TopBraid Composer
工具分类	否	是	是	是	是
图形工具	否	是	是	否	是
协同工作	否	是	是	是	是
本体库	是	是	是	否	是

通过以上比较可以看出:Protégé 开源软件既支持单机也支持客户/服务器模式,既支持文件存储也支持数据库存储,支持插件且有大量的插件可用,有大量的其他本体工具可用,推理服务方面既有内嵌的推理引擎也有可附加的推理,同时支持约束/一致性检测,有大量的工具可用,因此,本书决定采用 Protégé 作为本体建模工具。

1.4 多智能体理论

多智能体系统(Multi-Agent Systems,简称 MAS)起源于分布式人工智能、基于个体的建模和软件主体等多方向的研究和探索。20 世纪 80 年代后,多智能体系统的一般性理论初步形成。多智能体系统首先是一个概念模型,它定义了主体、主体行为、主体与环境的交互和主体间的交互;其次,多智能体系统是一个以计算机仿真为工具的实验系统,它通过仿真达到分析、理解、调节和设计源系统的目标;近年来,Agent 和多 Agent 系统理论和技术频繁出现在大量应用系统的设计中,对 Agent 的研究已成为人工智能学科的一个热点,已成为人工智能学科的核心内容。目前,Agent 方法和多 Agent 系统等相关技术已日益应用于交通控制、智能机器人、分布式控制、计算机网络、电子商务、软件系统等领域,成为一种对复杂系统进行分析、设计的有力的思想方法和工具。

1.4.1 智能体及行为

多 Agent 系统是由多个自主协作的 Agent 及其所处的环境构成的。但目前不同的学术流派并未对 Agent 形成统一的定义。按照伍德里奇(Wooldridge)等关于 Agent 的一个广泛引用的经典定义,Agent 一般具有以下 4 个方面的特征,但不局限于以下 4 个方面:

(1) 自治性:拥有对自己动作的控制。它从自身状态中生成这些动作,在没有人的干预下(至少在大多数情况下)行动。

(2) 反应性:环境变化被主体感知,主体将在适当的时间对这个新信息作出反应。

(3) 主动性:除了对环境变化的反应外,软件主体追逐个体目标,并主动发起动作,表现出一些不是直接由环境触发的活动。

(4) 社会性:与其他主体(或人类)通信的能力是软件主体定义的基础能力。

伍德里奇的定义实际上揭示了主体与智能活动相关的一些特征,因此,主体一般描述的是本身就具有智能的“活”的东西,包括人、动物和具有智能的植物,也可以包括由这些活的东西所构成的系统,如银行、企业和学校等;主体还可以涉及一些赋予了智能的人工物和虚拟物,如机器人、网络爬虫和智能算法等。

伯利费米勒等将主体定义为一个具有自治性的软件组件,主体通过一个可以与任何系统进行互操作的接口,并/或像人一样,按照自己日程安排为某些客户工作。

Agent 的结构是构造和实现其智能的基础,不同的 Agent 结构决定了 Agent 具有不同的智能行为。Agent 研究的主要问题是把 Agent 特性用一组模块来表示,并使得这些模块关联成一个有机的整体,以便对感知器及其他方式获取的输入信息进行有效处理,使得 Agent 产生合理的动作来影响环境、其他 Agent 及自身的状态。Agent 的基本结构如图 1-6 所示。

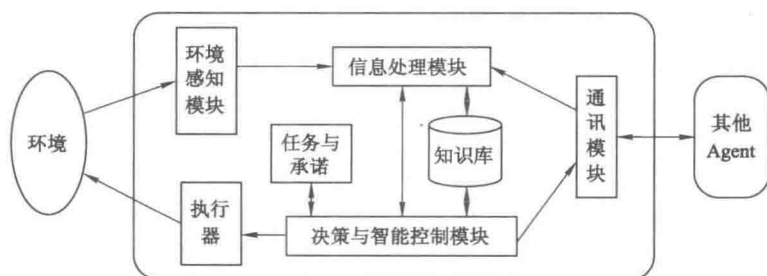


图 1-6 Agent 的基本结构

Agent 的基本结构由环境感知模块、信息处理模块、决策与智能控制模块、通讯模块、执行模块、知识库、任务与承诺等构成。其中 Agent 通过环境感知器或其他方式(如通过通讯模块获取其他 Agent 的相关信息)感知环境,然后通过信息处理模块对获取的信息进行处理,通过决策与智能控制模块进行决策控制,通过执行器作用于环境或通过通讯模块与其他 Agent 进行通讯,进而影响环境、其他 Agent 及自身状态。其间的信息处理与决策等均可能与知识库进行交互,并依据获取的知识信息进行推理,进而采取动作完成任务与承诺。

按照人类思维的层次模型,在 Agent 系统结构上,可以将 Agent 内部体系

进一步划分为三类:慎思型 Agent、反应型 Agent 和混合型 Agent。

(1) 慎思型 Agent。慎思型 Agent 又称为认知型 Agent 或思考型 Agent。其结构继承了传统的符号表示和符号推理的人工智能体系,将人工智能领域的感知、学习、方法、规则等进行封装,实现功能模块化。其结构如图 1-7 所示。

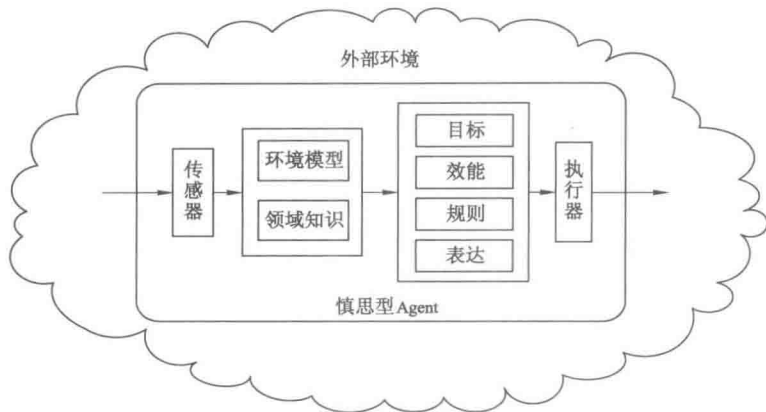


图 1-7 慎思型 Agent 的结构

慎思型 Agent 的核心部分是环境模型和规划器。慎思型 Agent 包含表示环境的符号模型,并且其决策是通过逻辑推理、基于模式匹配和符号操作进行的。这种方法中有一个基本假设:认知功能可以模块化,即将不同的认知功能如感知、学习、动作等分开来研究,再把它们组装在一起构成智能体。显然,功能模块化降低了系统的复杂性,使系统设计更易于实现。慎思型 Agent 体系结构的典型代表是 Bratman 设计的 IRMA, Burmeister 和 Sundermyerd 的 Cosy 和伦敦大学设计的 ARCHON。慎思型 Agent 的优点在于直观性、模块化和对复杂行为的抽象化描述的能力。

(2) 反应型 Agent。反应型 Agent 的研究依据是 R. A. Brooks 倡导的基于行为的人工智能,认为智能行为是 Agent 与它周围环境交互的结果,不包含用符号表示的世界模型,不使用复杂的符号推理,Agent 智能取决于感知和行动,并提出了 Agent 智能行为“感知-动作”模型,Agent 行为的复杂性可以是 Agent 运行环境复杂性的反映,而不是 Agent 内部复杂设计的反映。反应型 Agent 的设计思想是通过内部的简单设计实现复杂的系统行为。比较典型的结构如图 1-8 所示。

其典型代表是由 R. A. Brooks 研制的基于 Subsumptinn 的能对环境作出快速反应的机器人。反应型 Agent 虽然对外界环境的变化有很高的响应速度,但智能程度低,缺乏灵活性。