

# 基于团队知识共享模型的知识订阅和发布机制

陈明, 鲁强

(中国石油大学(北京)计算机科学与技术系, 北京 102249)

**摘要:** 团队成员进行项目开发时需要共享彼此之间的知识, 这样能够增加项目的开发效率。针对在 Internet 环境下进行团队项目开发的特点, 提出了一种团队知识共享模型, 通过使用本体来描述模型下的知识结构, 在此基础上实现知识订阅和发布的形式化表示方法和规则, 实验表明, 此模型下的知识订阅和发布机制能够加快团队成员之间的知识共享。

**关键词:** 本体; 订阅-发布知识; 知识共享

## Subscribing and Publishing Knowledge Based on Team Knowledge Sharing Model

CHEN Ming, LU Qiang

(Department of Computer Science and Technology, China University of Petroleum, Beijing 102249)

**【Abstract】**In order to advance development efficiency, team members need to share their knowledge about the project on which they work. Aiming at the characteristics of the development of team project under Internet, this paper puts forward a team knowledge-sharing model and organizes the knowledge by using ontology. Based on the model, it defines the formalized method and rule as to subscribing and publishing knowledge. Through the experimental result, the method in the model enables team member to conveniently share knowledge with each other.

**【Key words】** ontology; subscribe-publish knowledge; knowledge sharing

团队中成员为了协作完成任务, 需要彼此之间共享信息和知识。传统的Web上知识共享模型是集中式的, 例如BBS、新闻组等。集中式知识共享模型具有知识分类明确、易于组织和方便检索等特点。但是, 由于组织知识的结构和形式比较单一, 不能够充分描述具有特性的知识, 在知识结构发生变化后已有的知识不容易被重新组织使用。为了克服Web上传统的知识共享模型缺点, 诸葛海将知识流模型用于分布式团队知识共享<sup>[1]</sup>, 并给出知识的三维空间描述模型<sup>[2]</sup>。文献<sup>[3]</sup>使用语义网络的方法来表示知识结构, 通过在HTML中加入隐藏域值和对链接进行标注的方法来实现知识获取、表示、推理和传递。

上述的知识共享模型虽然在知识存储结构、知识检索方法上对传统的知识共享模型进行了改进, 但是为了实现知识共享, 它们都需要知识获取者主动地判断并查询自己所需要的知识, 而这种查询操作需要具有专业知识背景, 对于普通使用者并不适用。本文根据团队知识共享模型<sup>[4]</sup>, 对知识结构进行本体化表示, 并在此基础上研究了知识共享的订阅-发布机制, 使得用户只需按规则订阅项目中所需要的知识, 系统就能够向其返回相关内容。

### 1 团队知识共享模型

#### 1.1 团队知识共享模型结构

文献<sup>[4]</sup>定义了团队知识共享模型, 如图1所示。它由两类知识(个人知识-IKM和项目知识-PKM)和知识共享规则(发布规则、订阅规则、安全规则和其它规则)组成。个人知识是组成项目知识的基础; 项目知识则是按一定的知识分类标准来聚合项目成员有关此项目的知识集合。项目知识保存自身描述信息和到各个项目成员知识内容的链接, 以及到其它项目中相似知识的链接(项目之间相似链接-PSL 和内容之间相

似链接-CSL)。团队知识共享模型的定义见文献<sup>[4]</sup>。

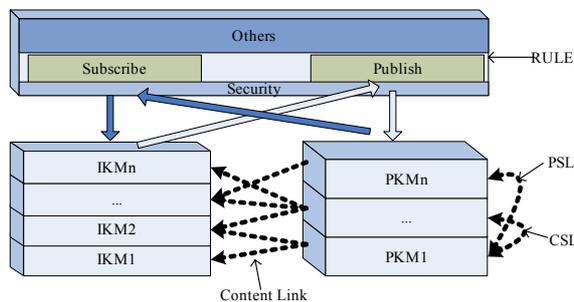


图1 团队知识共享模型

#### 1.2 团队知识共享模型中知识的本体表示

在团队知识共享模型中, 只有经过发布的个人知识才能成为共享知识, 才能够被其它项目成员或其它项目来订阅或检索, 将这部分共享知识称为领域知识。个人知识是按个人的偏好来进行分类的, 项目知识分类是按照项目的具体标准来进行分类。为了标志不同知识分类下的相同领域知识, 引入本体<sup>[5]</sup>来对此模型中的领域知识进行表示, 将此本体称为领域知识本体。

领域知识本体是一个知识分类目录, 在每个分类目录下保存个人发布知识的链接。领域知识本体包含两种关系: “part-of”和“is-a”。“part-of”关系表示类别之间是整体和部分的的关系; “is-a”关系表示类别之间有共同的属性, 用来

**基金项目:** 国家自然科学基金资助项目(60072006)

**作者简介:** 陈明(1949-), 男, 教授、博士生导师, 主研方向: 分布式并行计算; 鲁强, 博士

**收稿日期:** 2006-08-20 **E-mail:** coolmanlu@yahoo.com.cn

表示概念的逐步细化，类似于面向对象中的继承概念。如图2所示，实线表示“is-a”，虚线表示“part-of”。人工智能、数据库、程序设计语言都是计算机科学，即它们都具有计算机科学的共同特征，是计算机科学的子类，是“is-a”关系。而数据挖掘是交叉性学科，它包括了人工智能和数据库等方面的知识，它们之间是“part-of”关系。“part-of”和“is-a”关系都具有传递性质，例如 A part-of B, B part-of C, 那么可以得到 A part-of C。

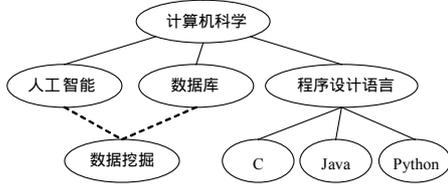


图2 领域知识本体片断

## 2 知识订阅和通知模型

### 2.1 形式化表示

根据团队共享模型中的定义和领域知识的表示，设  $PubSet = \{(pk, p, d, k) \mid pk \in PK, p \in PC, d \in DC, k \in CL\}$  为个人发布知识项的列表，其中，PK为项目标志；PC为项目的知识分类；DC为领域知识的分类；k为个人的知识链接；PK和PC可为空。PubSet中包含  $E = \{(pk, p, d_1, k_1), (pk, p, d_2, k_2), \dots, (pk, p, d_n, k_n)\}$  这样的知识发布项集合，用来表示项目中某一类知识由多个领域知识分类的知识组成。使用式(1)表示项目pk的p类型知识到领域知识本体Ω中的知识之间的映射。

$$\begin{aligned} A: p \rightarrow d \\ \Gamma_{pk}(p) = D \quad D \subseteq \Omega \end{aligned} \quad (1)$$

其中，D为项目中p类型知识到本体中映射的集合，将E看作矩阵，则  $D = d$ , d为E中的向量  $(d_1, d_2, \dots, d_n)$ 。

设  $SubSet = \{(pid, pk, p, subrule) \mid pid \in PID, pk \in PK, p \in PC, subrule \in SubRule\}$  为个人订阅知识列表，其中PID为个人标志，SubRule为订阅规则(见2.3节)。使用式(2)表示用户pid订阅项目pk下的p类型的知识。

$$\Psi_{pk}(pid) = p \quad (2)$$

设  $NotifySet = \{(pid, k) \mid pid \in PID, k \in CL\}$  为向用户返回其订阅知识的列表。使用式(3)表示向用户pid返回其订阅项目pk中的知识K。

$$N_{pk}(pid) = K \quad (3)$$

根据式(1)和式(2)可得

$$\Psi_{pk}(pid) = \Gamma_{pk}^{-1}(D) \quad (4)$$

其中， $\Gamma_{pk}^{-1}$ 为式(1)的逆映射，表示领域本体到项目知识之间的映射。由式(4)可知，用户订阅的项目知识内容d对应于领域知识本体D。D中的元素之间的基本关系，如式(5)所示。其中， $\phi$ 表示无关系； $\alpha$ 表示关系“is-a”； $\beta$ 表示关系“part-of”； $\alpha^+(d_i, d_j)$ 表示 $d_i$ 是 $d_j$ 的直接父类； $\alpha^-(d_i, d_j)$ 表示 $d_i$ 是 $d_j$ 的直接子类； $\beta^+(d_i, d_j)$ 表示 $d_i$ 是 $d_j$ 的最下整体； $\beta^-(d_i, d_j)$ 表示 $d_i$ 是 $d_j$ 的最上部分。所以， $\alpha^+(d_i, d_j) = \alpha^-(d_j, d_i)$ ,  $\beta^+(d_i, d_j) = \beta^-(d_i, d_j)$ 。

$$Rel = \{\phi, \alpha, \beta\} \quad i \neq j \quad (5)$$

$$RelExd(d_i, d_j) = P(d_i, d_j) \quad i \neq j \quad (6)$$

在基本关系的基础上定义D中元素的扩展关系，如式(6)所示。其中  $P(d_i, d_j) = p(d_i, d_m) \cdot p(d_m, d_n) \cdot \dots \cdot p(d_k, d_j)$ ,  $p(d_k, d_j) \in Rel$ ,  $P(d_i, d_j)$ 表示两个本体之间的关系序列。例如图2中，领域知识本体数据挖掘和程序设计语言之间的关系为 $\beta^+$ (数据挖掘, 数据库) ·  $\alpha^-$ (数据库, 计算机科学) ·  $\alpha^+$ (计算机科学,

程序设计语言) ·  $\alpha^+$ (程序设计语言, C)。

$$K = \{d_i, d_j \mid W(P(d_i, d_j)) > C, d_i \in D, i \neq j\} \quad (7)$$

式(7)表示向用户返回的订阅知识K是与订阅领域知识本体关系数值大于C的领域本体知识，其中W(RelExd)表示领域本体之间的关系值，C为常量。

### 2.2 相似关系和相关关系

根据领域本体中的“is-a”关系，如果两个本体共同拥有一个祖先本体，则这两个本体间具有相似关系；根据领域本体中的“part-of”关系，如果两个本体分别是一个本体的不同部分则这两个本体间具有直接相关关系，或者如果两个本体之间有共同的部分本体，则这两个本体间以及这两个本体的部分本体间具有间接相关关系。两个本体之间可以同时具有相似和相关关系。根据文献[6]定义本体之间的相似度，如式(8)所示。

$$sim(s_1, s_2) = \sum_{i=1}^n \delta_i(s_1, s_2) \theta_i \quad (8)$$

$$\delta(c_1, c_2) = \begin{cases} 1 & \text{当 } c_1, c_2 \text{ 前 } i \text{ 个父类相同} \\ 0 & \text{当 } c_1, c_2 \text{ 前 } i \text{ 个父类不同} \end{cases}$$

其中， $s_1, s_2$ 为领域知识；n为 $s_1, s_2$ 在领域知识分类树中最大的深度； $\theta_i$ 为权重。

根据本体之间的相似度可以得到项目之间的相似度，如式(9)所示。

$$PSim(P_1, P_2) = \frac{|P_1 \cap P_2|}{|P_1| + |P_2|} + \alpha \frac{\sum_{i=1}^u \sum_{j=1}^v sim(s_i, s_j)}{uv} \quad (9)$$

其中 $P_1, P_2$ 表示两个项目知识对应的领域知识本体集合； $u = |P_1 - P_2|$ ； $v = |P_2 - P_1|$ ； $s_i \in |P_1 - P_2|$ ； $s_j \in |P_2 - P_1|$ ； $\alpha$ 为权重( $0 \leq \alpha < 1$ )。定义本体之间的直接相关度为

$$Corr(s_1, t_1) = E \prod_{i=1}^{d_1-1} \omega(s_i) \prod_{i=1}^{d_2-1} \omega(t_i) \quad (10)$$

其中，E为常数； $\omega(x) = \frac{1}{k-1}$ ；k为本体x的最小整体本体所具有部分本体的数量； $s_{i+1}, t_{i+1}$ 分别为 $s_i, t_i$ 的最小整体本体； $d_1, d_2$ 分别为 $s_1, t_1$ 到m的“part-of”关系数量。

定义本体之间的间接相关度为

$$InterCorr(s_1, t_1) = \begin{cases} \frac{|Z|}{|S|+|T|} & \text{if } z \text{ is part of } s_1 \\ & \text{and } t_1, z \in Z \\ \frac{|D|}{|S|} \times \sum_{z \in Z} Corr(z, t_1) & \text{if } d \text{ is part of } s_1 \\ & \text{and } t_1 \text{ is part of } t \\ \sum_{z \in Z} Corr(s_1, z) \times Corr(z, t_1) & \text{if } z \text{ is part of } s \text{ and } t, \\ & s_1 \text{ part of } s, t_1 \text{ part of } t \end{cases} \quad (11)$$

其中， $S = \{s \mid \beta^+(s_1, s)\}$ ； $T = \{t \mid \beta^+(t_1, t)\}$ ；Z为两个本体的共同部分本体的集合，如果Z中本体之间具有“part-of”关系，则保留表示整体的本体，将表示部分的本体删除。因此，Z中的本体不具有“part-of”关系。当两个本体(s,t)之间具有共同部分本体时，式(11)表示了本体之间3种间接相关关系：s、t之间相关关系；s和t的部分本体之间相关关系；s的部分本体和t的部分本体之间相关关系。

### 2.3 订阅和通知规则

考虑到时间效率，为了得到合理的返回知识K，使用如下规则对需要计算的领域本体之间的关系做出限制。

设d为订阅的领域知识本体， $K_\alpha(d) = d \cup K_\alpha(u)$ ，其中u为d的子类； $K_\beta(d) = d \cup K_\beta(e) \cup K_\alpha(e)$ ，其中，e为d的部分； $B(d) = \{u_i \mid \alpha^+(k, u_i), \alpha^+(k, d), u_i \neq d\}$ ； $E(d) = \{e_i \mid \beta^+(m, e_i), \beta^+(m,$

$d), e_i \neq d\} ; R(d)=\{r_i|\beta^+(r_i, k), \beta^+(d, k)\} \cup \{r_i|\beta^+(k, r_i), \beta^+(k, z), \beta^+(d, z), r_i \neq z, d \neq k\} \cup \{r_i|\beta^+(k, r_i), \beta^+(z, d), \beta^+(k, n), \beta^+(z, n), k \neq z\}$ 。

如果用户订阅的知识为项目下的具体类型知识,则可以向用户返回以下几种类型的知识: $K_\beta(d)$  分类下的知识; $K_\alpha(d)$  分类下的知识; $U$ 分类下的知识,其中, $U=\{u_i | sim(d, u_i) > C_1, u_i \in B(d)\}, 0 < C_1 < 1$ ;  $M$ 分类下的知识,其中, $M=\{m_i | Corr(d, m_i) > C_2, m_i \in E(d)\}, 0 < C_2 < 1$ ;  $Z$ 分类下的知识,其中, $Z=\{z_i | InterCorr(d, z_i) > C_3, z_i \in R(d)\}, 0 < C_3 < 1$ 。如果用户用户订阅的知识为项目类型知识,则可以向用户返回以下两种类型的知识:根据式(1)向用户返回 $D$ 中的分类知识 $P_1$ ;根据式(9)向用户返回同项目 $p$ 相似度大于阈值 $C_p$ 的项目内容知识 $P_2$ 。

以上这些种类的知识,是根据提供给用户的订阅规则来向用户返回的知识。定义提供给用户订阅知识的规则  $SubRule = \{\chi, \varepsilon, \delta, \eta\}$ , 其中:

$\chi$  规则 根据订阅的项目 $p$ , 向用户返回知识 $P_1 \cup P_2$ 。

$\varepsilon$  规则 根据订阅的知识 $d$ 类型, 向用户返回知识 $K_\beta(d)$  或 $K_\alpha(d)$ 。

$\delta$  规则 根据订阅的知识 $d$ 类型, 向用户返回知识 $K_\beta(d) \cup U$ 或 $K_\alpha(d) \cup U$ 。

$\eta$  规则 根据订阅的知识 $d$ 类型, 向用户返回知识 $K_\beta(d) \cup M \cup Z$ 或 $K_\alpha(d) \cup M \cup Z$ 。

### 3 试验分析

在系统原型的试验测试过程中, 使用ODP(<http://rdf.dmoz.org>)作为试验测试数据。将structure.rdf.u8 文件中的内容作为领域知识本体, 其中, Topic标记中的内容表示一个领域知识本体; narrow标记表示领域知识本体之间的“is-a”关系; related标记表示领域知识本体之间的“part-of”; 将content.rdf.u8 文件中的内容作为发布到领域知识本体下的个人知识链接, 其中, ExternalPage标记描述了个人知识链接的内容。从ODP的文件中, 导入了 672 693 个领域知识本体, 以及发布在领域本体下的 4 529 429 条个人知识链接。由于在ODP中related标记表示“part-of”关系的语义较弱, 为了保证返回的订阅知识之间的相关度和缩小领域知识的检索范围, 设置在ODP中的“part-of”关系不具有传递性, 即 $K_\beta(d) = d \cup e \cup K_\alpha(e)$ , 其中 $e$ 为 $d$ 的部分,  $C_1=0.75, C_2=0.10$ 。根据各种规则所得的分类知识如表 1 所示。从表中的数据可以看出, 每一个规则分类下的知识都包括了基本分类下(表中URL数量列)的知识, 并且随着订阅规则的不同而返回与订阅知识相

(上接第 31 页)

### 6 结论

本文描述了一款 64 位可重构加法器的设计, 详细介绍了加法器的各个组成模块。该加法器为超长指令字 DSP 处理器所设计, 以满足 DSP 运算的灵活性和高性能, 同时又具有较小的功耗和面积。该加法器适用于多媒体、图形处理等应用, 也可应用于通用处理器。

#### 参考文献

- 1 Bedrij O. Carry Select Adder[J]. IRE Trans. on Electronic Computers, 1962, 11(6): 340-346.
- 2 Hwang Kai. Computer Arithmetic—Principles, Architecture and

似或相关的知识。

表 1 ODP 库中订阅规则下知识统计

项目	项目	映射领域本体		$\chi$ 规则	$\varepsilon$ 规则	$\delta$ 规则	$\eta$ 规则
		本体名	URL 数量				
A	a1	Languages/C++	8	1 087	3 991	14 884	1 593
		Software/GUI	11	34	11	337	42
	a2	Wireless_Data/WAP	5	114	115	373	1 031
B	B1	Languages/Java	2	4 544	996	14 884	5 395
	B2	Languages/JavaScript	33	587	146	14 884	1 697
		Internet/HTML	60	681	744	797	810
B3	Databases/Oracle	13	822	809	2 548	620	
C	c1	Programming/Threads	23	92	66	92	103
	c2	Languages/Concurrent	9	119	185	14 884	314

### 4 结语

根据项目团队在进行项目开发时的知识共享特点, 建立了团队知识共享模型, 并在此基础上建立知识订阅和发布规则的形式化表示方法以及规则。通过以上的的工作, 使得项目团队成员可以最大限度地利用其他成员的基于项目的共享知识, 加强了团队成员之间的知识交流。在将来的工作中, 将建立基于此知识共享模型下的知识推荐模型, 通过分析用户的偏好, 主动向用户推荐其需要的知识。

#### 参考文献

- 1 Zhuge Hai. Knowledge Flow Management for Distributed Team Software Development[J]. Knowledge-Based Systems, 2002, 15(8): 465-471.
- 2 Zhuge Hai. A Knowledge Grid Model and Platform for Global Knowledge Sharing[J]. Expert Systems with Applications, 2002, 22(4): 313-320.
- 3 Brian R G, Mildred L G H. Knowledge Acquisition, Modeling and Inference Through the World Wide Web[J]. International Journal of Human-computer Studies, 1997, 46(6): 729-759.
- 4 鲁强, 陈明. 一种基于本体的团队开发知识模型[J]. 计算机工程, 2006, 32(3): 193-196.
- 5 邓志鸿, 唐世渭, 张铭, 等. Ontology 研究综述[J]. 北京大学学报(自然科学版), 2002, 38(5): 730-738.
- 6 朱礼军. 万维网环境下基于领域知识的信息资源管理模式研究[D]. 中国农业大学, 2004.

Design[M]. John Wiley and Sons Inc., 1979.

- 3 Kogge P M, Stone H S. A Parallel Algorithm for the Efficient Solution of a General Class of Recurrence Equations[J]. IEEE Trans. on Computers, 1973, 22(8): 786-793.
- 4 Brent R, Kung H T. A Regular Layout for Parallel Adders[J]. IEEE Trans. on Computers, 1982, 31(3): 260-264.
- 5 Schmoekler M S, Putrino M, Roth C, et al. A Low Power High Speed Implementation of a PowerPC Microprocessor Vector Extension[C] //Proc. of the 14th Arithmetic Conf., Adelaide, Australia. 1999.
- 6 Patterson D A, Hennessy J L. Computer Architecture: A Quantitative Approach[M]. 3rd ed. Beijing: China Machine Press, 2002.