

一种基于 P2P 支持检索条件重构的构件库互联技术

李 琰 李 田 谢 冰 张 路 孙 家

(北京大学信息科学技术学院软件研究所 北京 100871)

(高可信软件技术教育部重点实验室 北京 100871)

(liyan@sei.pku.edu.cn)

A P2P-Based Component Library Interconnection Technique Supporting Query Refactoring

Li Yan, Li Tian, Xie Bing, Zhang Lu, and Sun Jiasu

(Software Institute, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871)

(Ministry of Education Key Laboratory of High Confidence Software Technologies, Beijing 100871)

Abstract Software reuse is a feasible way to solve the software crisis. With the development of software reuse techniques and network techniques, more and more component libraries emerge on the Internet. However, the components that the reuser needs usually distribute in multiple libraries, and the ways of component description in those libraries are different. This makes the acquirement of components quite difficult. Thus, it is necessary to provide reusers with an effective mechanism to help them acquire components from multiple component libraries. Proposed in this paper is a component library interconnection technique called DCLITTA which supports resource sharing among distributed component libraries and supplies a 'transparent' retrieval mechanism to reusers. DCLITTA organizes independent component libraries in a flexible way by leveraging the peer to peer (P2P) network architecture. Meanwhile, to deal with the differences in component description models among component libraries, DCLITTA refactors reusers' queries automatically to improve the retrieval effect. Based on the technique introduced in this paper, the authors designed and implemented the supporting system which has already been put into practical use in the component libraries in Beijing and Shanghai software parks.

Key words component library; component library interconnection; peer to peer network; component retrieval; query refactoring

摘 要 软件复用是解决软件危机的有效途径。随着软件复用技术和网络技术的发展,在 Internet 上出现了越来越多的软件构件库。复用者需要的构件往往分散在多个构件库中,而各个库中构件的描述方式也各不相同,这给复用者获取构件带来了一定的困难。因此,需要有一种有效的机制来帮助他们在多个构件库中获取构件。提出一种构件库互联技术 DCLITTA 以支持在分布的构件库之间实现资源共享,并为复用者提供了“透明”的检索机制。DCLITTA 采用对等网络(P2P)体系结构将各个独立的构件库组织在一起。同时,针对构件库间构件描述模型的差异,DCLITTA 通过自动地将复用者的检索请求进行重构来改善检索的效果。基于该互联技术,已设计实现了相应的支持系统,并实际应用于北京、上海等软件园的构件库中。

关键词 软件构件库;构件库互联;对等网络;构件检索;检索条件重构

中图法分类号 TP311.5

软件复用是提高软件开发效率和质量的一条切实可行的途径^[1]。作为软件复用的一种形式,基于构件的软件开发(CBSD)是利用已有的软件构件来实现新的软件,避免了从零开始进行开发。CBS D 成功的关键因素是必须存在大量可复用的软件构件,并且复用者能够方便、准确地找到自己所需要的软件构件(以下简称构件)^[2]。

为了充分利用构件资源,必须对构件进行有效的组织和管理。软件构件库就是对构件资源进行管理,并对软件复用提供支持的基础设施^[3]。近些年来,随着软件复用技术和网络技术的发展,在 Internet 上出现了很多公用和商用的软件构件库,例如,Download.Com^①,Component Source^②等。这些软件构件库(以下简称构件库)为复用者获取所需的构件提供了便利。

然而在很多情况下,复用者所需的构件分散在多个构件库中,这些库具有各自独立的描述和组织机制,例如不同的分类法、术语集合等。这给复用者获取所需的构件带来了一定的困难。例如,当开发完成飞机票预定功能的应用程序时,开发人员希望获得能够实现机票查询和银行账户处理功能的构件以降低开发成本。那么,开发人员需要首先找出可能包含上述构件的构件库,再分别访问各个构件库进行检索。上述过程要求开发人员既能够发现可能拥有所需构件的构件库,又能够有效地使用各个构件库中的检索机制,这给开发人员带来了额外的负担。如果能在 Internet 上的各个构件库之间实现资源共享,形成一个整体的共享系统,并为复用者提供有效的检索机制,便能为软件复用提供有力的支持。

为了实现构件资源共享,需要将构件库按照一定的体系结构有效地组织起来。传统的服务器模式(如 C/S 结构),往往会受制于服务器端的带宽和处理能力而影响整体系统的响应速度和可扩展性。另外,在实现构件资源共享的过程中还需要考虑到构件库的独立性。构件库会自由地加入或退出资源共享的系统,构件库中构件的可用情况也会随时变化;而构件资源共享的系统应该能够适应此类动态的变化。由此可见,如何有效地将多个构件库组织在一起,并使得整体系统具有较好的可扩展性和健壮性,是实现构件资源共享需要解决的基础问题。

另一方面,由于各个构件库是按照自身用户群

体的特点以及领域特征来进行组织的,所以在构件描述术语的选择上必然存在着差异^[4]。因此,当把多个构件库中的资源组织在一起时,复用者的检索请求往往会因为不同构件库中术语使用上的差异而难以得到令人满意的效果。而如果缺乏有效的机制来帮助检索,那么复用者在获取构件时仍会面临困难^[5]。

本文提出一种能够有效支持构件资源共享的构件库互联技术 DCLITTA(dynamic component library interconnection technique with transparent access)。该技术采用混合对等网络体系结构将 Internet 上独立的构件库动态地组织在一起。同时,针对构件库之间在描述模型上的差异,DCLITTA 利用同义词典自动地将复用者的检索请求进行重构,使之适应于另外的构件库。这样复用者便可以通过单个库中的描述、检索模型来检索多个库中的构件。基于本文中的互联技术,我们设计实现了相应的支持系统,该系统已经实际应用于北京、上海等软件园的构件库系统中。

1 构件库互联技术 DCLITTA

为了帮助复用者在多个构件库中有效地获取所需的构件,需要解决下面两个方面的关键问题:

1) 构件库的发现与通信。如何将多个构件库灵活、健壮地组织成一个整体。

2) 构件检索。由于各个构件库的描述模型存在差异,如何为复用者提供一种“透明”的检索机制来帮助他们进行检索,即复用者可以利用单个库中的描述、检索机制便能有效地检索多个库中的构件。

针对上述两个问题,本文提出了一种基于对等网络体系结构并支持检索条件重构的构件库互联技术 DCLITTA。如图 1 所示,DCLITTA 采用混合对等网络体系结构将分散在 Internet 上的构件库组织成一个构件库互联体系,并使用 LDAP 目录服务器管理全局拓扑结构,该结构限定为一个树状层次图。针对构件检索中的问题,本文提出并研究了检索条件重构的方法来实现透明的检索。DCLITTA 中每个构件库结点拥有自己的描述模型,父结点计算并保存子结点间以及本结点与子结点间的术语之间的相关度。当进行检索时,复用者在某结点上按其描

① <http://www.download.com>

② <http://www.componentsource.com>

述模型构造检索请求,该结点与其邻近的上层和下层结点通信并对检索条件进行重构,收到重构后检

索条件的各结点执行本地检索,并将结果直接返回给检索的发起结点.

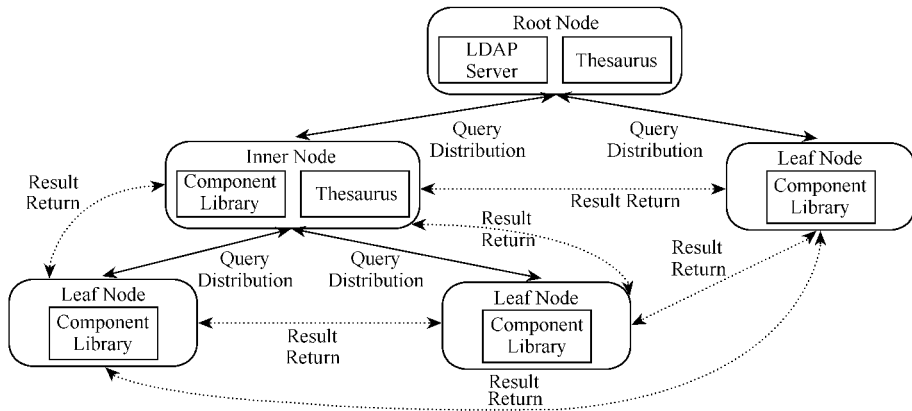


Fig. 1 The architecture of DCLITTA.

图1 DCLITTA 框架

1.1 DCLITTA 的构件库发现与通信机制

由于 Internet 环境下存在的构件库较多,所以发现与通信机制采用的体系结构必须具有较高的可扩展性.同时,考虑到构件的所有权分属于各个构件库,所以在进行构件资源共享的过程中还应该考虑到构件库的独立性.发现与通信机制中另一个重要的设计环节是互联体系的拓扑结构.

传统的构件库互联技术(如 NHSE^[6]等),往往采用服务器模式(如客户端/服务器、浏览器/服务器等)来进行结点的组织.这种模式中,中心服务器由于受到局部带宽和处理能力等方面的制约,比较容易成为系统的瓶颈,从而影响到整个体系的响应速度和可扩展性.

近几年来,对等网络(P2P)^[7]技术日益成为分布式系统的常见模式.与传统的服务器模式相比,对等网络中结点之间相互提供服务,是平等的关系.这样就降低了对中心服务器的依赖,消除了单一失效结点,使得系统具有较好的可扩展性;易于处理网络的动态变化.而且,由于不存在集中的控制机制,所以对等网络体系结构不易受到如拒绝访问攻击等网络故障所带来的影响,使得系统具有一定的健壮性.鉴于对等网络体系结构比较适合构件资源共享的场景,因此 DCLITTA 中采用这种体系结构来组织结点.

根据对等网络中是否存在提供目录服务的结点,对等网络可以分为纯对等网络和混合对等网络.在纯对等网络中,结点采用广播的形式来查找服务者,这样网络负载较大,而构件库互连是一种重量级的服务器之间的互联,所以更适宜采用混合对等网

络的形式.

为了使结点能够发现、加入互联体系, DCLITTA 设置了中心结点对整个体系进行管理,并提供发现和查找服务.中心服务器将对申请加入的构件库进行验证,从保证构件共享可信性的角度出发来决定是否批准加入.为了避免中心结点成为系统的瓶颈,保证系统的可扩展性, DCLITTA 中的中心结点专一提供目录服务,不再运行构件库系统以减轻中心结点的负载.另外,由于互联体系中的拓扑结构和结点信息等目录数据相对稳定,修改频率较低,而在执行构件检索等操作时需要频繁地读取目录数据;所以,这类数据具有读频率远远高于写频率的特点.轻权路径访问协议 LDAP 通过对读取、查询操作进行优化处理,保证了其在响应速度方面的优势.因此, DCLITTA 中的中心结点采用 LDAP 来管理目录数据,从而提高中心结点的处理能力.

在拓扑结构的选择上, DCLITTA 选用的是树形结构.它具有两个优点:首先,树形结构是层次形的结构,易于管理,可以减少网络带宽的使用量,可扩展性较好.其次,当部分结点出现网络故障时,影响仅仅局限在子树,因此系统拥有较好的稳定性.

因此, DCLITTA 中采用树形的拓扑结构,并将提供目录服务的中心结点作为树的根结点.采用树形拓扑结构的另一个优势在于树形结构和 LDAP 目录服务器的信息组织形式相同,因此采用树形结构可以进一步发挥 LDAP 的优势.

1.2 DCLITTA 的检索机制

在构件库中,复用者的检索过程可以表示为图

图 2(a) 中所示的模型^[8]: 1) 复用者根据自身对问题的理解, 形成检索需求; 2) 复用者根据构件的描述模型

和相应的检索模型将检索需求构造成检索条件; 3) 复用者进行检索, 得到检索结果。

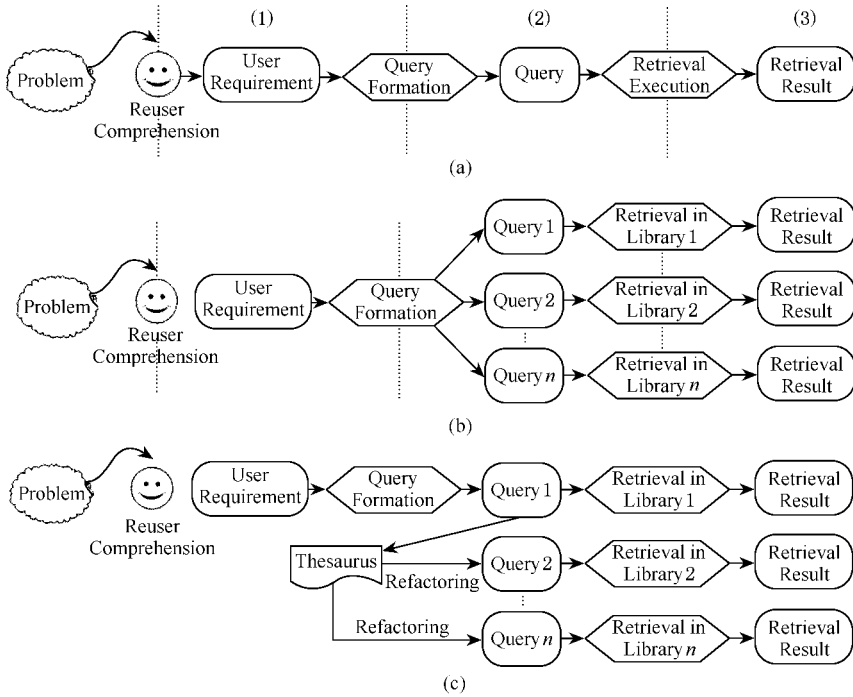


Fig. 2 Retrieval model in component library. (a) Component retrieval model; (b) Retrieval model in which queries are formed for each description model; and (c) Retrieval model in DCLITTA.

图 2 构件库检索模型。(a) 构件检索模型; (b) 针对不同的描述模型分别构造条件的检索模型; (c) DCLITTA 中的检索模型

因为构件库之间的术语使用方式存在差异, 所以当需要在多个构件库中检索构件时, 复用者要针对每个构件库分别构造检索条件, 如图 2(b) 所示。而这种方法虽然扩大了可选资源的集合, 但是却显著地增加了复用者的学习负担。并且, 不同的描述模型可能存在着语义差异, 复用者对一种描述模型中术语的理解可能会影响到对其他描述模型中术语的理解。

针对检索中的问题, DCLITTA 根据构件库间术语的关联程度来构造同义词词典, 并利用同义词词典对复用者的检索请求进行有效的重构。DCLITTA 中的检索流程如图 2(c) 所示, 当复用者给出了检索条件之后, DCLITTA 将借助同义词词典来重构该检索条件, 使它适用于另外的构件库。由此可以在图 1 的树形结构中不断进行分发、重构。收到重构后检索条件的结点执行本地检索, 将检索得到的结果直接返回给检索发起结点。这种透明的检索机制可以减轻复用者的学习负担, 又比简单的查询分发的方法改善了检索的效果。

1.2.1 检索机理

在 DCLITTA 的检索流程中, 同义词词典对术

语之间的关联程度进行量化。这种量化的术语间的关联程度被称为术语相关度。为了计算两个术语之间的相关度, 我们可以从语法的角度出发, 借助于自然语言处理领域中的技术来比较术语文本间的相似度。另外, 通常两个复用库中会存有一些相同构件, 我们可以利用这些构件在不同构件库中的描述信息, 从语义的角度来衡量两个构件库中术语的相似程度。在已有的工作^[9]中, 我们提出了利用两个构件库中的相同构件, 从术语所描述的语义信息的角度来计算术语相关度的方法, 取得了一定的效果。然而, 当构件库之间拥有的相同构件较少甚至不存在相同构件时, 上述方法的基础便不再成立。为此, 本文进一步将术语语法上的相似度加入到相关度的计算中, 从而提高了相关度计算的准确性, 扩大方法的适用范围。

1.2.2 术语语法方面相关度的计算

如果描述构件的术语字符串具有很高的相似度, 那么两个术语具有较高的关联程度。利用自然语言处理等相关领域中的技术, 我们可以按照如下的步骤来计算术语语法方面的相关度 R_1 。

1) 借助于词法分析工具对术语进行分析、切

词,得到一个单词的集合;

2) 在集合中去掉连词、助词等对语义影响不大的单词;

3) 求得两个术语对应单词集合的并集 A , 并由它得到对应的向量 T :

$$T = t_1, t_2, \dots, t_n,$$

其中 $n = |A|$; 并且 $\forall i, j, 1 \leq i \leq n, 1 \leq j \leq n$, 有 $t_i \in A$, 而且当 $i \neq j$ 时, $t_i \neq t_j$;

4) 相应地,两个术语也被表示为下面的向量形式:

$$V = b_1, b_2, \dots, b_n;$$

$$b_i = \begin{cases} 1, & \text{如果 } t_i \text{ 在术语 } V \text{ 中出现,} \\ 0, & \text{其他情况.} \end{cases}$$

5) 通过计算两个术语向量的余弦值,就可以得到术语语法方面的相关度 $R_1 = \cos(V_i, V_j)$.

1.2.3 术语语义方面相关度的计算

DCLITTA 仍使用文献[9]中的方法来完成语义方面相关度的计算,即根据相同构件在两个构件库中的描述来构造贝叶斯网络模型(如图3所示),从而以条件概率 $P(T_k/S_j)$ 来得到术语在语义方面的相关度 R_2 . 根据文献[9]中的定义,我们可以利用条件概率公式得到:

$$R_2 = P(T_k/S_j) = P(T_k S_j) / P(S_j). \quad (1)$$

在构件库中各个术语相互独立的情况下,可以推导出:

$$P(T_k S_j) = \sum_{c \in C_0} P(T_k S_j / c) \times P(c) = \sum_{c \in C_0} P(T_k / c) \times P(S_j / c) \times P(c), \quad (2)$$

其中 C_0 是两个构件库共有的构件集合. 由此,我们可以分别计算公式中的各个部分,并得到最终结果.

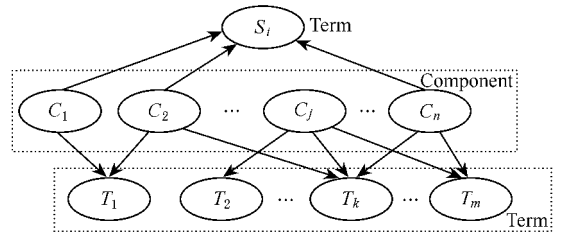


Fig. 3 Bayesian network model for relevancy calculation.

图3 计算相关度的贝叶斯网络模型

对于两个术语,其术语相关度的最终计算公式为

$$R = \omega_1 \times R_1 + \omega_2 \times R_2, \quad (3)$$

其中 $\omega_1 + \omega_2 = 1, \omega_1 \geq 0, \omega_2 \geq 0$.

式(3)中, ω_1 和 ω_2 为 R_1 和 R_2 的权重. 当两个构件库面向的应用领域相近,拥有的相同构件较多时,便可以增加语义方面的权重. 反之,则应该更多地借助语法方面的信息来计算相关度.

1.2.4 DCLITTA 中同义词词典的构造与更新

DCLITTA 中的检索流程利用同义词词典对检索请求进行重构. 而同义词词典的计算依据构件信息和术语信息,会随着时间发生变化. 为了构造和更新同义词词典,我们需要在构件库之间复制构件信息和术语信息.

DCLITTA 中对同义词词典进行更新采用的算法流程如图4所示. 其中步骤1)和2)会循环进行,记录更新信息,当到达一定时间阈值后,控制从步骤3)开始运行.

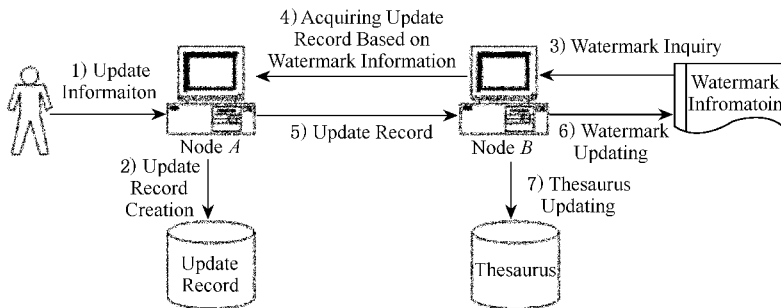


Fig. 4 The update process of thesaurus.

图4 同义词词典更新流程

2 系统实现及关键技术

2.1 北京大学构件库互联体系支持系统简介

在国家“八六三”项目“基于 Internet,以构件库

为核心的软件开发平台”中,基于本文中的互联技术,我们设计实现了北京大学构件库互联体系支持系统. 系统采用 EJB 技术和 J2EE 规范实现业务逻辑,对外提供了 Web 形式和 Web Services 形式的访问接口.

目前,系统的中心节点架构在北京大学、北京中关村软件园^①和上海软件构件中心^②两地的构件库已经加入了互联体系,实现了两地构件资源的共享。下一步,我们将逐步扩大互联体系的规模,将广州软件园等地的构件库纳入到互联体系中。

在构件库互联体系支持系统中,为了清除各种复杂网络情况给通信造成的障碍,系统采用 JXTA^③(Juxtapose)作为 P2P 的通信协议。JXTA 独立于 HTTP、TCP 等具体的传输协议,可以有效地应对如防火墙、网络地址转换(NAT)等多种网络配置。

2.2 系统实现关键技术点

2.2.1 结点的失效管理

在某些情况下,结点会由于网络故障等原因而无法与其他结点进行通信。此时,通信的源结点便会在通信之前先执行一个 ping 操作,检查通信目标结点是否处于可通信状态。如果无法与目标结点通信,源结点会通知目录服务器目标结点当前不可达。当一个结点在单位时间段内的不可达次数超过阈值时,互连体系便认为该结点失效,使其临时脱离互联体系。不可达次数的阈值和单位时间段的长度均可以通过配置文件进行设定。

2.2.2 检索结果收集的管理

检索结果收集要判断是否所有在检索步长范围内的结点都已返回了结果,从而尽快对用户做出响应。为此,需要计算检索步长范围内结点的个数。这里使用 C 来表示检索步长范围内结点的个数,那么对于分发步长为 n 的检索条件,可以得到

$$C = \sum_{i=1}^n C_i, \quad (4)$$

其中 C_i 为 $TTL = i$ 结点的个数。根据分发的流程,进一步可以得到

$$C_i = \sum_{j=1}^{C_{i+1}} N_j, \quad (5)$$

这里 N_j 表示某个 $TTL = i + 1$ 结点向外分发检索条件的结点个数。 N_j 可以由拓扑结构及检索分发控制变量决定。由此便可以计算出 C 的大小,进而判断是否已从所有在检索步长范围内的结点收集到结果。

为了避免因网络传输故障等问题导致用户等待时间过长,系统还根据不同的检索步长分别设定了相应的等待时间阈值。如果超过时间阈值,即使还有结点没有返回结果,系统也将不再等待而将已收集结果返回。

3 相关工作

从 20 世纪 90 年代起研究人员就开始探讨如何利用网络实现构件库间资源的共享。NHSE^[6]使用虚拟库的方式进行资源共享。基于 BIDM 及其扩展协议^[10],各构件库物理结点将构件的描述信息和描述模型传给虚拟库结点,通过虚拟库结点,用户可以根据相应的描述模型检索实际构件库物理结点上的构件资源。DCH^[11]将网络上多个构件库中的构件索引信息提交到一个管理中心结点,并使用 LDAP 来对管理中心结点上的信息进行操作和控制,提供统一的构件描述。用户通过管理中心结点检索构件。DROO^[12]是一个管理面向对象构件的分布式构件库系统,它采用超文本链接的方式来管理构件及其之间的关系并交互构件信息,每个结点都需要维护所有构件的 Class-URL。UDDI^[13]中使用信息复制技术,在所有结点间复制信息,在一个结点上修改(插入、删除、更新)了数据,每个结点都要更新对应的信息副本。这样就使得分布在 Internet 上的所有结点都拥有相同的资源,用户可以在每个结点直接进行本地检索。Uranus^[14]方法中每个结点拥有自己的分类结构,用户在各个结点进行检索,并将自己期望得到的构件复制到自己的私有结点中。

上述技术只是简单地将构件资源进行集中式地管理,并没有考虑构件库之间在描述、检索模型等方面的差异。因此,同本文中的 DCLITTA 相比,上述技术未能充分体现构件资源共享的优势。

近些年来,也有研究工作利用对检索条件进行重构来完成构件资源共享。OML^[15]中使用中间转换模块和本体技术相结合的方法实现构件资源共享。各构件库结点使用本体来管理描述模型,利用中间转换模块将用户的检索条件进行转换。统一检索机制^[16]的方法是根据术语的同义词词典为多种分类模式建立统一的视图,用户可以根据这些统一视图来检索多个构件库中的构件。虽然 OML 和统一检索机制也提供了检索条件的重构机制,但在重构的过程中,这些技术都需要人工的干预。

DCLITTA 与以往的相关工作所不同的另一点是结点间的体系结构。DCLITTA 采用的是混合型的对等网络体系结构。这种形式比以往的服务器模式

① <http://www.bsw.net.cn>

② <http://www.sstc.org.cn>

③ <http://www.jxta.org>

更为灵活、稳定,更适合开放、动态的 Internet 网络环境。

4 结束语

为了帮助复用者在 Internet 上的多个构件库中获取构件,本文提出了一种构件库互联技术 DCLITTA,从而将多个构件库中的构件资源加以共享并为复用者所利用。针对构件库的特点,DCLITTA 采用混合型的对等网络体系结构将多个构件库结点组织在一起,并通过 LDAP 来管理体系的树形拓扑结构。根据描述构件的术语在语法和语义方面的关联程度,DCLITTA 构造了同义词词典,并利用它对复用者的检索请求进行重构,从而帮助复用者透明地完成面向多库的检索。在国家“八六三”项目中,作者实现了 DCLITTA 的支持系统,并投入实际的使用。目前系统运行稳定,累计构件访问次数达到 58 万次,下载次数超过 1 万次。通过统计分析,我们发现系统的平均查准率达到了 42%,与不进行检索条件重构直接进行分发的方法相比,有了 14% 的提高;系统中进行查询的平均响应时间不超过 5s。

本文仅对对等网络体系结构结合检索条件重构的方法进行了初步地探讨,下一步的工作需要在对等网络的具体组织形式以及检索条件的重构算法等环节上进行进一步的改善、优化。另外,需要将更多的构件库结点纳入到支持系统中,从而在更加复杂的环境下考察 DCLITTA 的实用性和稳定性。进一步地,可以借助搜索引擎等相关技术来实现构件库结点的自动发现。

参 考 文 献

- [1] V Basili, L Briand, W Melo. How reuse influences productivity in object-oriented systems[J]. Communications of the ACM, 1996, 39(10): 104-116
- [2] H Mili, F Mili, A Mili. Reusing software: Issues and research directions[J]. IEEE Trans on Software Engineering, 1995, 21(6): 528-562
- [3] W B Frakes, T P Pole. An empirical study of representation methods for reusable software[J]. IEEE Trans on Software Engineering, 1994, 20(8): 617-630
- [4] S V Browne, J W Moore. Reuse library interoperability and the World Wide Web[C]. In: Proc of the 19th Int'l Conf on Software Engineering. New York: ACM Press, 1997. 684-691
- [5] Jia Xiaohui, Chen Dehua, Yan Mei, et al. Research on matching model and algorithm for faceted-based software component query[J]. Journal of Computer Research and Development, 2004, 41(10): 1634-1638 (in Chinese)
(贾晓辉,陈德华,严梅,等.基于刻画描述的构件查询匹配模型及算法研究[J].计算机研究与发展,2004,41(10): 1634-1638)
- [6] S Browne, J Dongarra, S Green, et al. The national HPC software exchange[J]. IEEE Computational Science and Engineering, 1995, 2(2): 62-69
- [7] J Alkerdine, L Melville, I Sommerville. Dependability properties of P2P architectures[C]. In: Proc of the 2nd Int'l Conf on Peer-to-Peer Computing. Los Alamitos, CA: IEEE Computer Society Press, 2002. 173-174
- [8] D Lucrecio, A Prado F, E S de Almeida. A survey on software components search and retrieval[C]. In: Proc of the 30th Euromicro Conf. Los Alamitos, CA: IEEE Computer Society Press, 2004. 152-159
- [9] Pan Ying, Wang Lei, Zhang Lu, et al. Relevancy based semantic interoperation of reuse repositories[C]. In: Proc of the 12th ACM SIGSOFT Symp on Foundations of Software Engineering. New York: ACM Press, 2004. 211-220
- [10] IEEE. IEEE Std 1420.1. IEEE Standard for Information Technology-Software Reuse-Data Model for Reusable Library Interoperability: Basic Interoperability Data Model (BIDM) [S]. 1995
- [11] J X Ci, W T Tsai. Distributed component hub for reusable software components management[C]. In: Proc of the 24th Int'l Computer Software and Applications Conf. Los Alamitos, CA: IEEE Computer Society Press, 2000. 429-435
- [12] M Ohtsuki, N Yoshida, A Makinouchi. A distributed repository for object-oriented software components[C]. In: Proc of the 3rd Asia-Pacific Software Engineering Conf. Los Alamitos, CA: IEEE Computer Society Press, 1996. 439-446
- [13] T Bellwood, L Clément, D Ehnebuske, et al. UDDI Version 3.0 Published Specification[OL]. <http://uddi.org/pubs/uddi-v3.htm>, 2004-10
- [14] Y C Sun, M L Kao, C L Lei. A fully distributed approach to repositories of reusable software components[J]. Journal of Information Science and Engineering, 2000, 17(1): 147-158
- [15] RMM Braga, M Mattoso, CML Werner. The use of mediation and ontology technologies for software component information retrieval[J]. ACM SIGSOFT Software Engineering Notes, 2001, 26(3): 19-28
- [16] Ma Liang, Xie Bing, Yang Fuqing. The unified facet-based method to retrieve component in multi-library[J]. Acta Electronica Sinica, 2002, 30(12A): 2149-2152 (in Chinese)
(马亮,谢冰,杨芙清.多构件库统一刻画检索机制[J].电子学报,2002,30(12A): 2149-2152)



Li Yan , born in 1982. Ph. D. candidate. His main research interests include software reuse and software component library.

李 琰 ,1982 年生 ,博士研究生 ,主要研究方向为软件复用、软件构件库。



Li Tian , born in 1982. Master candidate. His main research interests include software component library and P2P network.

李 田 ,1982 年生 ,硕士研究生 ,主要研究方向为软件构件库、对等网络。



Xie Bing , born in 1970. Ph. D. and associate professor. Senior member of China Computer Federation. His main research interests include software engineering and formal methods.

谢 冰 ,1970 年生 ,博士 ,副教授 ,中国计算机学会高级会员 ,主要研究方向为软件工程、形式化方法。



Zhang Lu , born in 1973. Ph. D. and associate professor. Senior member of China Computer Federation. His main research interests include software engineering and program comprehension.

张 路 ,1973 年生 ,博士 ,副教授 ,中国计算机学会高级会员 ,主要研究方向为软件工程、程序理解。



Sun Jiasu , born in 1946. Professor and Ph. D. supervisor. Senior member of China Computer Federation. His main research interests include programming language and software engineering.

孙家 琰 ,1946 年生 ,教授 ,博士生导师 ,中国计算机学会高级会员 ,主要研究方向为编程语言、软件工程。

Research Background

This research belongs to the National High-Tech Research and Development (863) Project “ the Component Library Centric Software Development Platform based on the Internet ” and is also supported by the National Grand Fundamental Research 973 Program , the National Key Technology R&D Program and the National Natural Science Foundation of China. In the past few years , in the domain of software reuse , we have developed and deployed several software component libraries in China based on the technical achievement of the Jade Bird project. These component libraries contain a group of closely integrated software tools and provide the technical infrastructure for the component based software development (CBSD). However , these libraries are isolated. Users in one component library could not make use of resources in other libraries which may hamper the success of software reuse. As an important part of the 863 project , the research in this paper mainly helps to solve the problem of resource sharing among different component libraries on the Internet. It connects component libraries by way of using the P2P network architecture and provides an efficient retrieval mechanism which conceals the differences of component description models in various libraries by using the query refactoring technique. Based on the research in this paper , we designed and implemented the supporting system and put it into practical use in the component libraries in Beijing and Shanghai software parks.