

B2B 电子商务协作模型研究^{*}

李双庆 程代杰 何玲

(重庆大学计算机科学与工程学院 重庆400044)

Research on B2B Electronic Business Collaboration Model

LI Shuang-Qing CHENG Dai-Jie HE Ling

(School of Computer Science and Engineering, Chongqing University, Chongqing, 400044)

Abstract B2B is an important application form in electronic business, it provides the methods to do business activities through Internet among companies. But the heterogeneous of the information systems between companies makes it difficult in interoperating between them. In this paper, a B2B electronic business collaboration model is proposed, solving the collaboration between heterogeneous business systems by defining the business information ontology and describing the hierarchy in business collaboration process.

Keywords Electronic business, Collaboration, Ontology, Business process

1. 引言

电子商务已经成为 Internet 的一种重要的应用形式,在此领域的研究也非常活跃。B2C 模式的电子商务形式单一而容易形成统一规范,以 SET 为代表的交易模型较好解决了 B2C 电子商务中交易支付安全、认证、隐私和不可抵赖等问题,而 B2B 模式的电子商务则需要面临更多复杂的问题。由于企业构建的信息系统或电子商务系统往往各不相同,因而将它们有机联系起来,实现商务协作,是一件很具挑战性的工作。目前,该领域的研究已经取得了一些进展。OMG 的 CORBA^[2]解决了分布环境下对象操作问题,W3C 的 XML^[1]为异构系统间数据传递与互操作提供了方便,UN/CEFACT 和 OASIS 也正在制定基于 XML 的商务信息交换规范 ebXML^[4]。

本文将针对 B2B 电子商务应用提出一种商务实体协作模型 EBCM(E-Business Collaboration Model),讨论该协作模型中商务知识库的结构,并给出了商务流程的形式化描述和定义方法以及以 DTD 描述的商务流程示例。

2. 电子商务协作模型

B2B 电子商务涉及两个或多个商家之间的商务协作关系。为简化起见,本文提出的 EBCM 模型以两商家间的协作为基础,重点解决两商家间协作关系建立和实施的过程。但是,该模型也适用于多商家协作的环境,可以将多商家协作分解为若干两商家间协作关系。EBCM 的总体架构如图1所示。协作模型由三个角色组成,分别是商务知识库、商务代理 BA1 和 BA2。商务知识库包含本体(ontology)库、商务库。本体库描述模型中概念和领域知识,商务库则描述商家注册信息,包括商家概要信息、商务流程和商务文档。在本文以下内容中将对此作详细论述。商务代理(Business Agent, BA)是商家在电子商务环境中的实体。

在 EBCM 下,一个完整的商务协作过程分为四个阶段,共六个步骤组成。第一阶段是注册,包含第1、2步。首先由 BA1 从 Repository 中下载概念、域知识、预定义文档或协作流程。如果预定义的文档和协作流程能够满足需求,则直接引用,并定义商家概要信息,否则,需要根据商务概念和域知识自定义文档或协作流程。将结果发送到 Repository 注册;第二阶段是发现阶段,包括第3、4步。BA2 从 Repository 中发现感兴趣的注册信息,并下载相应的商务文档和流程;第三阶段是协商阶段,包括第5步。BA2 以下载的文档和协作流程为基础,与 BA1 进行协商,并最终达成一致,形成电子合同。第四阶段是执行阶段,包括第6步。BA1 和 BA2 根据双方认同的商务流程传递电子文档,执行商务过程。该阶段可能配合有某些非电子商务的行为,如发货、运输等,这些行为本身不在 EBCM 讨论范围。

模型中的 BA 可以被集成到企业信息系统(如 ERP 或 MIS)或电子商务系统中成为一个独立的功能部件或智能代理程序。从企业信息系统的角度看,BA 是系统数据交换的网关,通常 BA 需要实现企业信息系统的数据库与商务协作数据之间的转换功能。

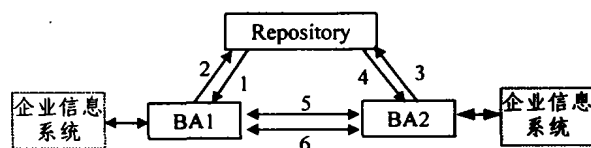


图1 B2B 电子商务协作模型

需要指出,EBCM 对商务协作中文档的交换形式不做限制,但是由于 XML 对数据的表示能力很强,目前被普遍用于不同系统间数据传递和互操作,因此建议采用 XML 作为数据交换语言。同时注册服务、注册查询和发现可采用通用的标准协议,如 LDAP 和 UDDI。此外,模型本身也不考虑网络传输和传输安全本身的实现方法,但在定义商务协作能力和协

^{*}项目资助:重庆市科技攻关项目(1999.6065)。李双庆 博士研究生,主要研究方向为电子商务,多代理技术,分布式计算。程代杰 教授,博士生导师,主要研究方向为电子商务,计算机网络,并行处理。何玲 讲师。

作各方确立协作关系时,可以定义网络传输方式和安全模式要求,由所承载的系统提供相应服务。

3. 商务知识库结构

EBCM 的商务知识库(Repository),如图2所示,由本体库、商务库共同构成。本体是概念化的明确表示和描述,能够使某一领域的概念有共同理解,提高交流和协作的效率以及不同系统的互操作性。概念化 C 可以表示为 $C = \langle D, W, R \rangle$, 其中 D 为一个领域, W 为该领域的相关状态的集合, R 是领域空间 $\langle D, W \rangle$ 上概念关系的集合^[7]。因此在 EBCM 中本体库为商务协作流程定义提供了概念基础,使协作各方在商务概念上统一,也为协作各方扩展商务领域或应用的概念提供方法。

EBCM 将 ontology 库^[3]分为核心 ontology (core ontology)、域 ontology (domain ontology) 和应用 ontology (application ontology)。核心 ontology 定义商务协作中的基本概念,例如空间、时间、对象、事件和行为等,它独立于特定的商务应用和领域;域 ontology 描述特定商务领域(例如汽车或农产品)的基本术语,它依赖于核心 ontology;应用 ontology 描述特定应用中(如汽车制造)的基本术语,它依赖于核心 ontology 和域 ontology。

以下是以 DTD 方式给出的本体库定义:

```

<!-- Ontology-definitions -->
<!ELEMENT ontology-definitions ((class-def | slot-def) * )>
<!-- Class-expressions -->
<!ENTITY % class-expr " ( class | slot-constraint | AND | OR | NOT) " >
<!ELEMENT AND ((%class-expr); (%class-expr); +)>
<!ELEMENT OR ((%class-expr); (%class-expr); +)>
<!ELEMENT NOT (%class-expr);>
<!-- Class-definition -->
<!ELEMENT class-def (class, documentation?, subclass-of?, slot-constraint * )>
<!ATTLIST class-def type ( primitive | defined ) "primitive" >
<!-- Class-name -->
<!ELEMENT class EMPTY>
<!ATTLIST classname CDATA #REQUIRED>
<!ELEMENT documentation ( #PCDATA)>
<!ELEMENT subclass-of (%class-expr); +>
<!-- Slot-definition -->
<!ELEMENT slot-def (slot, documentation?, subslot-of?, domain?, range?, inverse?, properties?)>
<!-- Slot-name -->
<!ELEMENT slot EMPTY>
<!ATTLIST slot name CDATA #REQUIRED>
<!ELEMENT subslot-of (slot) +>
<!ELEMENT domain (%class-expr); +>
<!ELEMENT range (%class-expr); +>
<!ELEMENT inverse (slot)>
<!-- Slot-properties -->
<!ELEMENT properties( transitive | symmetric | other) * )>
<!ELEMENT transitive EMPTY>
<!ELEMENT symmetric EMPTY>
<!ELEMENT other ( #PCDATA)>
<!-- Slot-constraint -->
<!ELEMENT slot-constraint (slot, ( has-value | value-type | cardinality | max-cardinality | min-cardinality) + )>
<!ELEMENT has-value (%class-expr); +>
<!ELEMENT value-type (%class-expr); +>
<!ELEMENT cardinality (number, %class-expr); +>
<!ELEMENT max-cardinality (number, %class-expr); +>
<!ELEMENT min-cardinality (number, %class-expr); +>
<!ELEMENT number ( #PCDATA)>

```

EBCM 的商务库包含商务对象、商务流程定义以及商务实体注册的协作信息概要。商务对象描述了商务协作中的一些高级信息单元,它可以通过 ontology 库来定义。商务流程定义商务协作中的各种流程,其中一些是公共的商务流程,可以直接下载应用,也可以针对特定应用定制商务流程。协作信息概要是协作实体注册的信息,其中包含有该实体的基本信息、商务角色、协作能力以及可选的协作方式,该信息被用于

发现商务协作伙伴,也是协作各方协商确定电子合同的基础。

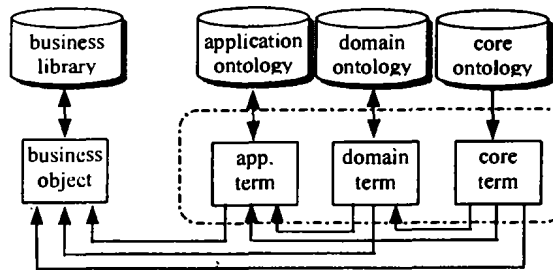


图2 Repository 结构

4. 商务协作流程

商务协作流程是指商务实体在达成协作关系之后,按照一定的规则实现商务协作的过程。EBCM 商务协作流程中涉及若干要素,以下是它们的形式化描述:

定义1 商务元素集合 $BE = AO \cup DO \cup CO$, 其中 AO 为应用本体库, DO 为域本体库, CO 为核心本体库。

定义2 商务对象集合 $BO = \{bo | bo = \langle be \rangle, be \in BE\}$ 。

定义3 商业文档集合 $BD = \{bd | bd = \langle bo, be \rangle, 其中 $bo \in BO, bo \neq \phi, be \in BE\}$ 。$

定义4 文档流集合 $DF = \{df | df = \langle bd1, bd2, bd3, \dots \rangle, 其中 $bd1, bd2, bd3, \dots \in BD\}$ 。$

定义5 事务集合 $T = \{t | t = \langle df, p \rangle, df \in DF, p$ 为文档流传输协议。

定义6 两方协作集合 $BC = \{bc | bc = \langle T', SIG, S \rangle, 其中 $T' \subset T, SIG$ 为触发事务的商务信号集, S 为事务执行顺序和状态变迁图。$

定义7 多方协作集合 $MC = \{mc | mc = \langle BC', G \rangle, BC' \subset BC, G$ 为协作关系图。

根据以上定义, EBCM 中的商务协作流程可以是两方协作或者是多方协作。多方协作可以看作是两方协作的扩展,由若干两方协作根据协作关系图的描述组合而成。每个协作中的原子操作是事务,事务的执行顺序和状态变迁由商务信号触发。在每个事务中涉及协作两方若干商业文档的传递,文档流传输协议规定了各方传递文档的类型和时序。

EBCM 定义有若干种商业文档,每个商务文档由一个或多个商务对象以及零个或多个若干商务元素构成,其中商务对象通过一个或多个商务元素定义。商务元素为 ontology 库中定义的概念、术语或域知识。

可以看出, EBCM 模型给出了一种层次化定义商务流程的方法。商务流程的外特性通过一系列商务文档的传递来表现。商务文档本身的规范目前有很多种, UN/CEFACT 的 ebXML^[5,6] 是其中影响最广泛的规范,目前该规范仍然在不断完善中。

以下是以 DTD 方式描述的 EBCM 商务协作流程定义:

```

<! ELEMENT MultiPartyCollaboration ( Documentation * , BusinessPartnerRole * )>
<! ATTLIST MultiPartyCollaboration
  name CDATA #REQUIRED
  nameID ID #IMPLIED
>
<! ELEMENT BinaryCollaboration ( Documentation * , InitiatingRole, RespondingRole, ( Documentation * | Start | Transition | Success | Failure | BusinessTransactionActivity | CollaborationActivity | Fork | Join) * )>
<! ATTLIST BinaryCollaboration

```

基于 SDL 语言代码自动生成技术研究

吴琦 熊光泽

(电子科技大学计算机科学与工程学院 成都610054)

Research on Automatic Code Generation Based on SDL

WU Qi XIONG Guang-Ze

(College of Computer Science and Engineering, UEST, Chengdu 610054)

Abstract As one of the key technology of CASE tools, code auto-generation has a wide application future. However, at present, some of problems limit its application in the practical project, such as executive efficiency of code generation, the combination with the hardware and software and etc. In this paper, the main factors of code auto-generation are introduced in details. The main parts of the code auto-generation based on SDL and the main factors which will effect the ultimately code performance are analyzed. The improved methods aiming at the different software and hardware platform and application performance are presented.

Keywords SDL, Code generation, Embedded system

1. 引言

目前嵌入式实时系统的复杂度越来越高,自动代码生成技术已成为缩短嵌入式系统开发时间和降低开发成本的主要方法之一,也是近十年软件开发工具研究的热门课题^[1-6]。所谓自动代码生成,即把形式化描述的系统需求转化为特定软硬件平台上基于某一目标语言的系统实现。代码自动生成技术研究虽然已经取得一定成果,也被应用到一些商业工具中,但生成代码的执行效率、与实际软硬件平台的结合等问题限制了其在实际工程中的使用。根据转化程度不同自动代码生成又分为完全的自动代码生成和部分的自动代码生成,本文就基于 SDL 的完全自动代码生成的关键技术及其对最终代码的影响进行初步探讨。

2. 自动代码生成技术的难点分析

自动代码生成技术一般由以下基本要素组成:

```

name CDATA #REQUIRED
nameID ID #IMPLIED
pattern CDATA #IMPLIED
beginsWhen CDATA #IMPLIED
endsWhen CDATA #IMPLIED
preCondition CDATA #IMPLIED
postCondition CDATA #IMPLIED
timeToPerform CDATA #IMPLIED
)
(! ELEMENT BusinessTransaction ( Documentation *,
RequestingBusinessActivity, RespondingBusinessActivity))
(!ATTLIST BusinessTransaction
name CDATA #REQUIRED
nameID ID #IMPLIED
: :
)
(! ELEMENT BusinessDocument ( ConditionExpression?,
Documentation *))
(!ATTLIST BusinessDocument
name CDATA #REQUIRED
nameID ID #IMPLIED
specificationLocation CDATA #IMPLIED
specificationElement CDATA #IMPLIED
)
: :

```

总结 本文讨论了 B2B 电子商务应用领域中商务协作的问题,提出了一种 B2B 电子商务协作模型 EBCM,它通过一个商务知识库来协调协作各方的关系,其中本体库解决了

1)形式化描述语言:是描述系统需求时使用的一种抽象的系统需求分析语言,其优点就是进行系统需求设计时不必考虑系统目标平台的具体细节。用于嵌入式系统的有 SDL^[6]、UML^[7]、Z^[8]、SCR^[9]等等。其中,通用建模语言(UML)虽使用广泛,但目前尚只能支持自动生成代码框架;SDL 语言可以与 UML 配合使用,弥补了 UML 的这一缺点,因此目前利用 SDL 语言进行通信系统需求描述已相当普遍,并逐渐使用于其他嵌入式实时系统。

2)嵌入式系统硬件平台:从体系结构总体上又可分为多机系统和单机系统。多机系统,即由多台可独立运行的嵌入式计算机构成的嵌入式系统,多机系统中的嵌入式计算机可以是同构的也可以是异构的,它们之间通过嵌入式通信网络互联;单机系统,即由一台独立运行的嵌入式计算机构成的嵌入式系统,单机系统又可分为多 CPU 和单 CPU 系统。

3)嵌入式系统软件平台:是指嵌入式实时系统上的系统软件,一般包括板级支持包(BSP)、嵌入式实时操作系统

异构系统间概念、术语和域知识统一与共享问题,商务库实现商务实体及其商务协作能力的注册、发现以及协作关系建立机制。同时还给出了商务协作流程层次化的定义方法,并以 DTD 形式进行了描述,为商务协作流程的实现提供了参考。该模型不仅是一种理论上的思想,同时还具有较好的实用性。

参考文献

- 1 Specification of the XML 1.0. <http://www.w3.org/TR/REC-xml>. 2001
- 2 The Common Object Request Broker: Architecture and Specification. <http://www.omg.org/> 2002
- 3 Uschold M, Gruninger M. Ontologies: Principles, Methods and Applications. The KnowledgeEngineering Review. 1996, 11(2): 93~115
- 4 ebXML Technical Architecture Specification. Version 1.0.4. <http://www.ebxml.org/specs>, 2001
- 5 ebXML Business Process Specification Schema. Version 1.0. <http://www.w3.org/TR/REC-xml> 2001
- 6 ebXML Collaboration-Protocol Profile and Agreement Specification. Version 1.0. <http://www.w3.org/TR/REC-xml> 2001
- 7 Guarino N. Formal Ontology and Information Systems. In: Proc. of FOLS'98, 1998. 3~15