

XML 消息传递协议的分析与研究

张珂 岳利华

(中国科学技术大学计算机科学系 合肥 230026)

摘要 在 XML 消息通信架构中,传递协议是其中的重要一环。本文在讨论了目前已有方案的可行之处和存在的问题后,提出了一种利用面向对象方法来构建模块化的传递协议,为面向复杂电子商务、电子银行应用的 XML 消息传递提供了一种有效可行的方法。

关键词 XML, XMLP, SOAP, 电子商务, 电子银行

Analysis and Research of XML Messaging Protocol

ZHANG Ke YUE Li-Hua

(Department of Computer Science, University of Science & Technology of China, Hefei 230026)

Abstract Messaging is one of the most important part of the framework of XML message communication. In this paper we will firstly introduce requirements of XML messaging protocol. Secondly some existed solutions are to be presented, and then we will analysis their merits and problems. In the last part a modular XML messaging protocol which constructed by using object-oriented methods in e-commerce and e-bank is proposed to explore a feasible model and challenges it faces.

Keywords XML, XMLP, SOAP, E-commerce, E-bank

1 XML 消息传递的需求

XML(eXtensible Markup Language,可扩展置标语言)是由 W3C(World Wide Web Consortium,互联网联合组织)^[1]于 1998 年 2 月发布的一种标准,同 HTML 一样是 SGML(Standard Generalized Markup Language,标准通用置标语言)的一个简化子集。它以一种开放的自我描述方式定义了数据结构,在描述数据内容的同时能突出对结构的描述,从而体现出数据之间的关系。随着分布式应用尤其是 Web 模型被广泛地采用,结合了 SGML 的丰富功能与 HTML 易用性的 XML,受到越来越多的软件开发者的支持和程序开发人员的喜爱,显示了它的巨大应用潜力。

虽然 XML 是 W3C 通过的标准,但公司可自行规范专属标签(tags)。所以在使用 XML 通信时,除了有词汇和方言问题外,一个更重要的问题是通信协议的问题。也许有人认为,HTTP 已经足够了。因为在实际应用中,随着应用需求日趋复杂化,仅提供传输的功能往往并不足够,特别是在电子银行和电子商务的应用领域,例如网上银行支付,需要 CA 认证和数字证书对交易双方身份的合法性进行确认,同时对转账交易信息完整性、保密性、不可否认性进行保证,同时还需要对关键交易信息进行存储直到所有双方确认交易成功,还有交易信息的加密传输问题;在实时网上购物交易系统中,交易数据会在交易双方不断交互传递以最后达成一个交易结果,这样就首先需要一种可以被传递的格式对文档编码然后被接受方理解和处理,其次需要交易机制能对交易过程和数据进行加密和认证,并能保证传递过程中每个交易是完整的,防止因为交易方的掉线或第三方入侵带来的威胁。另外,在怎样定义通信双方服务调用的规则、数据分发的规则、如何序列化

数据等方面,HTTP 不能提供对这些问题的任何帮助。要么应用程序自己支持这些协调、安全和可靠性的开销——或者干脆不支持。设想在多个企业间协调复杂的交易!显然需要一些标准的传递方式^[2]。

W3C 联盟在 WebServices 专题下有一个 XML Protocol (XMLP)工作组^[4]在这方面做了很多研究,当然他们更多的是服务于 Web Services 环境,主要致力基于 XML 的消息传递和 RPC(Remote Procedure Call,远程过程调用)的协议。XMLP 工作组对 XML 消息传递的应用情况^[5]和需求^[6]作了详细的分析,还提出过 XMLP 的抽象模型^[7]。由于应用需求的复杂化,现有的一些 XML 消息传递协议已经显示出一些缺陷,而且这些缺陷往往是由协议框架设计决定的。尽管具体的环境和应用情况的需求各有差异,我们将尽量从客观的角度来探讨可以满足基本不需求 XML 消息传递协议框架,主要着眼于功能的整合。

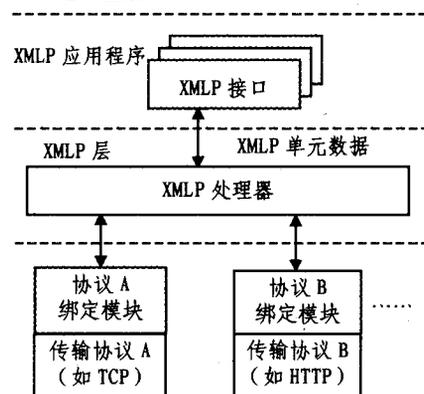


图 1 XMLP 抽象模型

2 已有的解决方案

XML 消息传递协议(简称为 XMLP)的抽象模型^[1]的工作流程如图 1 所示,这里我们先简单介绍一下 XML 消息传递协议的实现机制,再分析一些已有的解决方案和研究情况。

2.1 XML 消息传递协议的实现

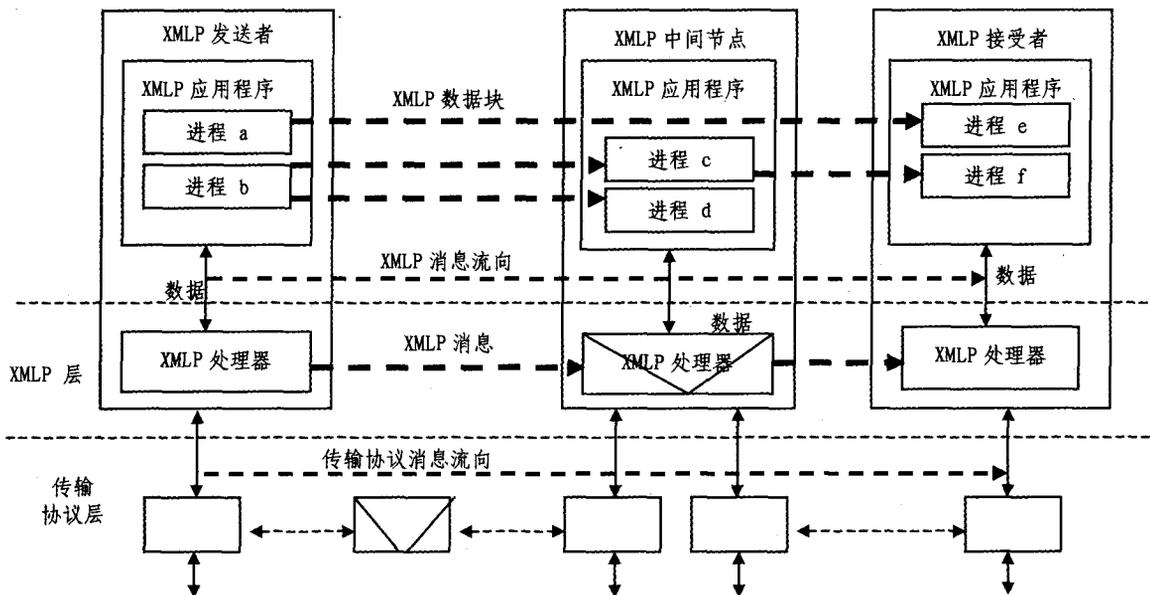


图 2 XMLP 通讯模型

XMLP 通讯模型典型的代表是 SOAP(Simple Object Access Protocol, 简单对象访问协议), 下面的例子^[3] 阐明了 SOAP 中一个简单的通信信息, 包括了两个不是 SOAP 定义而是应用程序定义的元素: 头块元素 alertcontrol 和体块元素 alert。头块元素包括两个参数: priority 和 expires。体块元素包括的是实际传送的信息。

```

<env:Envelope xmlns:env="http://www.w3.org/2002/12/soap-envelope">
  <env:Header>
    <n:alertcontrol xmlns:n="http://example.org/alertcontrol">
      <n:priority>1</n:priority>
      <n:expires>2001-06-22T14:00:00-05:00</n:expires>
    </n:alertcontrol>
  </env:Header>
  <env:Body>
    <m:alert xmlns:m="http://example.org/alert">
      <m:msg>Pick up Mary at school at 2pm</m:msg>
    </m:alert>
  </env:Body>
</env:Envelope>
    
```

2.2 现有的 XML 消息传递方案

作为代表方案的 SOAP 采用了 HTTP 作为底层通讯协议, RPC 作为一致性的调用途径, XML 作为数据传送的格式, 允许服务提供者和服务客户经过防火墙进行通讯。RPC 的描述可能不大准确, 因为 SOAP 一开始构思就是要实现平台与环境的无关性和独立性, 每一个通过网络的远调用都可以通过 SOAP 封装起来。SOAP 的两个主要设计目标是简单性和可扩展性。这就意味着有一些传统消息系统或分布式对象系统中的某些性质将不是 SOAP 规范的一部分。SOAP 在商业尤其是 Web 服务方面得到广泛的应用。

支持传递 XML 消息的通讯协议当然不止 SOAP 一种, 还包括了 XML-RPC, WDDX, ebXML 和 JMS 等等。W3C 组织的 Eric Prud'hommeaux 和 Ken Macleod 调查了这些协议, 并给出一个非常好的总结^[8]。XML-RPC^[9] 提供了一个非

协议的实现主要包括几个方面: 封装, 定义了一个描述消息中的内容是什么, 是谁发送的, 谁应当接受并处理它以及如何处理它们的框架; 编码规则, 用于表示应用程序需要使用的数据类型的实例; 交互协定, 表示双方通信约定, 如请求和应答的协定; 传输协议绑定, 使用底层协议交换信息。图 2 描述了 XML 消息的通讯模型。

常简单使用在 HTTP 上传递 XML 的 RPC 机制。WDDX (Web Distributed Data Exchange) 是由 Allaire 公司开发的, 提供了一个在 HTTP 之上交换复杂数据结构的机制。WDDX 声明的目标是“提供一个更类似 Web 的方法在不同的网络实体间传送结构化数据对象, 而不需要将开发 Web 应用的编程方法从面向页面改变到面向对象。”但是 WDDX 序列化的方法是基于结构的而不是基于对象的。可以看出, XML-RPC, SOAP 和 WDDX 都是基本的在 HTTP 上序列化和传递 XML 编码数据的技术^[2], 也是相对简单和现实的解决方案。

ebXML 是一项倡议, 参与者包括很多大公司和和官方标准协会。ebXML 是一个规范集, 这些规范共同实现了模块化电子商务框架。ebXML 的构想是实现一个全球电子市场, 不同规模和不同地区的企业可以通过交换基于 XML 的消息来合作和进行商业活动^[10]。ebXML 消息传递支持在多方交易处理中必需的高层语义。这些语义包括 一对一以及一对多路由模型, 对多方回路文档交换的支持, 以及根据消息头属性的服务质量确定。ebXML 与传输协议无关, 甚至可以用 SOAP。

Java 消息服务(Java Message Service, JMS) API 是 J2EE 平台的构成元素。JMS 1.0.2 定义了两种类型的消息传递域(它们是相互独立的), 即点对点发布/订阅。尽管 JMS 不是专门为传递 XML 设计, 但是在实际应用中由于它对消息交换高层语义的支持使得它也可以传递 XML。

2.3 存在的问题

首先是依赖 RPC 的序列化问题, 例如 SOAP, XML-RPC 提供了一个序列化的方案, 提供了最适合支持通常在分布式对象运算中常见的同步机制的传输协议。尽管 SOAP 之上可提供更高的协议, 但是其依赖于的 RPC 机制会带来一些天生的缺陷, 如难以实现高度的交互性, 在实现扩展协议编程接

口时也会有困难,另外在安全上也有问题。

其次,现有的一些协议由于保持简单性而对复杂的爸爸应用需求支持不足,而且扩展性也不好,使得协议的适应性不强。这是协议在整体框架设计带来的问题。

另外,不同的应用需求对协议的要求是不同的,对于很多 Web 服务应用,可能只需要 XML、HTTP 再加上自定义的消息传递协议。实际上,W3C 的工作组也没有在其 Web Services 架构中使用 SOAP。

3 面向对象的 XML 消息传递协议

为了避免一些已经存在缺陷和适应 XML 消息传递应用需求的复杂化,我们认为协议设计要着重考虑以下几个方面:

1、序列化的实现应当更高层

由于直接使用 RPC 机制会带来一些问题,如难以实现高度的交互性,在实现扩展协议编程接口时会有困难,在安全上有问题。为了可交互序列化机制应使用高层协议实现,而不应依赖于面向 RPC 的实现。

2、协议应当面向对象

由于序列化的方法是基于结构而不是基于对象所以不能被用来交换具有复杂关系的对象实例,因此应采用面向对象更适合通用地表达商务逻辑,所以应采用面向对象方法来弥补这样的缺陷,以便能更通用地帮助协议实现模块化,以及提高模块的可重用性。

3、协议应当简单化,并有良好的可扩展性

在像 Web 环境这样的松散结构下,要求开发的简易性、系统的可扩展性,这也是对 XML 消息传递协议的要求。从这个角度来说,SOAP 是一个典型代表。SOAP 本身不解决高层的分布式对象问题,例如对象引用、对象激活、分布式垃圾收集、成批传送消息、生命周期管理等。

基于以上的分析,我们认为 XML 消息传递协议应是一个简单的、扩展性良好的面向对象的解决方案,并能在更高层实现序列化。

4 面向对象的 XML 消息传递协议的体系结构

考虑到可扩展性和编程模型的简易性,我们给出一个面向对象的 XML 消息传递协议的体系结构,如图 3 所示。面向对象指使用离散的对象来构建软件系统,当 XML 依赖于协议对象的组合、聚集时,协议框架也有更大的灵活性,可以支持更复杂的应用需求和有更好的扩展性。在 XMLP 层(即面向对象的 XML 消息传递协议层)所组装的模块封装了适当的基本功能。

图 3 中的 XMLP 层的模块可以扩展,如在某些环境下节点可能需要负责路由,那就应该将路由功能加入作为一个基本模块。在设计模块化协议的时候需要考查一般化的 XML 消息传递的功能需求(本文前面已提到),还有服务模型和通讯模型(如图 2),来得到协议对象模型。在实现上就要遵循模块化通信系统的实现策略和协议对象的设计原则,其中后者是难点和重点。一般来说,一个协议对象只完成一个专门的协议行为,例如封装对象应当只代表一个特定封装算法的一种实现,而不是多种可供选择的方法。协议对象应当是自治的、通用的,并且对象间应当是弱耦合的。而在协议对象之上还可以实现各种服务对象,适应各种应用需求,可以层次化地扩展服务功能,保持良好的扩展性。

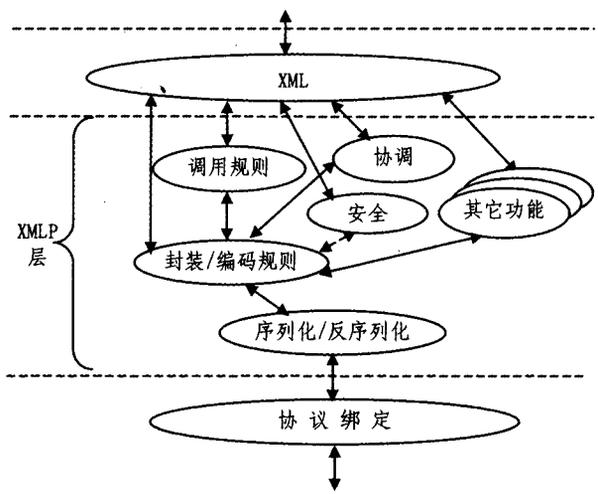


图 3 模块化的 XML 协议体系结构

从可扩展性和可复用性的角度来看,由于使用对象技术,新的协议功能和机制可以被加入,而不会影响已存在的协议功能和机制。甚至可以通过组装已存在的协议功能,提供新的服务。

另外,需要强调的一点是在一个层次化的通信系统中,一定要分清哪些功能应该在 XMLP 层实现,哪些应该在 XMLP 层的上层或下层实现,保持协议的简单性很重要。

结束语 由于 XML 在电子商务、电子银行的应用越来越广泛,本文从协议应用的角度,分析了目前在电子银行和电子商务等应用领域,XML 协议涉及的具体问题,尽量从较高的角度来介绍协议应用的现状,以及分析一些典型协议的可行之处和存在的问题,并尝试使用面向对象的建模技术,将模块化网络协议设计为可构建、可重用和可扩展的架构,提出了面向对象的 XML 消息传递协议的解决方向和体系结构,这也是我们以后研究工作的基础。

参考文献

- 1 W3Cs website: www.w3.org
- 2 Van Huizen G. Messaging: The transport part of the XML puzzle, July 2000. <http://www-900.ibm.com/developerWorks/cn/xml/xml-messaging/index-eng.shtml>
- 3 W3C SOAP Version 1.2. <http://www.w3.org/TR/soap12/>
- 4 <http://www.w3.org/2000/xp/Group/>
- 5 SOAP Version 1.2 Usage Scenarios. W3C Working Draft, 26 June 2002. <http://www.w3.org/TR/2002/WD-xmlp-scenarios-20020626/>
- 6 XML Protocol (XMLP) Requirements. W3C Working Draft, 26 June 2002. <http://www.w3.org/TR/2002/WD-xmlp-reqs-20020626>
- 7 XML Protocol: Abstract Model. W3C Working Draft, 9 July 2001, <http://www.w3.org/TR/2001/WD-xmlp-am-20010709/>
- 8 Prud'hommeaux E, Macleod K. XML Protocol Comparisons. March 2000. <http://www.w3.org/2000/03/29-XML-protocol-matrix>
- 9 XML-RPCs website. <http://www.xmlrpc.com/>
- 10 ebXML's website. <http://www.ebxml.org/>
- 11 Sommers F. Why use SOAP?. <http://www.artima.com/webservices/articles/whysoap.html>