

一种适合于分布式虚拟环境的层次式协议结构^{*})

徐丹 陈继明 潘金贵

(南京大学计算机软件新技术国家重点实验室 南京 210093)

摘要 随着分布式虚拟环境中信息交互要求的不断提高,目前主要采用的基于兴趣表达式的兴趣过滤机制已经不能满足需求,XML 路由器技术的提出为进一步提高分布式虚拟环境的可扩展性注入了新的动力。本文在对基于兴趣表达式的兴趣管理协议结构进行分析的基础上,以虚拟博物馆为例给出了适合动态信息的任务表达式格式,并根据其潜在的层次式结构提出基于 XML 封装协议的思想,使其能够更好地运用于基于 XML 路由器的主动网络。

关键词 分布式虚拟环境,兴趣表达式,任务表达式,XML

A Hierarchical Protocol Structure in Distributed Virtual Environment

XU Dan CHEN Ji-Ming PAN Jin-Gui

(State Key Lab for Novel Software Technology, Department of Computer Science and Technology, Nanjing University, Nanjing 210093)

Abstract With the demands of information interaction increasing in distributed virtual environment (DVE), interest filter based on interest expression can't satisfy requirements now, the proposal of XML router technology gives new drive to improving the scalability of DVE. Followed by analyzing the protocol structure of interest management based on interest expression, this paper introduces task expression of dynamical information by illustrating virtual museum, and proposes the schema of XML-based encapsulating protocol according as its potential hierarchical structure, which will be adapted to XML router-based active network better.

Keywords Distributed virtual environment, Interest representation, Task representation, XML

分布式虚拟环境^[1] (Distributed Virtual Environment, DVE)提供了一个共享的虚拟空间,并且能够通过计算机网络支持对象的仿真以及多个分布用户的交互和通讯。数据信息是 DVE 中的基本信息,如下载的虚拟场景以及场景中虚拟人的交互等都属于数据信息。DVE 中的数据大致可以分为静态数据和动态数据两类^[2],其中静态数据主要包括 3D 图形数据、2D 图像、视频、声音等组成虚拟场景的数据,动态数据则主要是虚拟空间中虚拟人的交互信息。

在 DVE 中,有限的网络带宽一直是制约 DVE 性能的一个主要因素,尤其随着对虚拟现实的逼真度要求不断提高,需要在网络上传输的信息量也越来越大,网络延迟问题也日益严重,主动路由技术成为减少网络通信量,提高系统可扩展性的一种很有研究价值的技术。目前的主动路由技术主要采用基于兴趣表达式的方式对协议进行封装传输,但在匹配和过滤信息时需要在内存中指定的位置获取相应的信息,因此对于动态交互信息的表示缺乏灵活性,不能随意地加入新的属性,也没有较强的可扩展性,无法适应动态信息多变的形式。

本文提出了一种适合于分布式虚拟环境中动态数据的任务表达式交互协议结构,该结构采用 XML 来封装协议内容^[3,4],使其以 XML 形式在网络上传输,极大增强了 DVE 中通信的效率和灵活性,同时以虚拟博物馆为例给出了基于 XML 的协议封装形式,很好地实现了由基于兴趣表达式转换到基于 XML 的通信协议封装形式。

1 主动兴趣管理

1.1 协议结构

构建一个完整的主动兴趣管理系统,需要有实现兴趣区域订购的协议、网络上数据传输的协议等。这些协议包括 Subscription Routing Protocol(SRP)、Real time datagram Delivery Protocol(RDP)、Stream Transport Multicast Protocol(STMP)、Geometry Transport Protocol, GTP^[5]。其中 SRP 和 RDP 底层都是以 UDP 作为支撑,而 STMP 和 GTP 的底层由 TCP 支撑,如图 1 所示。

SRP 协议用以构造组播树、传播路由信息、维护路由信息、维护组播树、退出组播树。RDP 协议用以发布数据,它在 SRP 协议建立的路由信息基础上进行数据包的转发。由于它们是基于 UDP 协议的,因此无法保证可靠性,只有利用重传机制来满足特殊的可靠性需求。STMP 协议用于可靠的数据传输,它是为了满足今后网络会议和桌面共享应用对数据传输可靠性的要求而设计的。GTP 规定了几何数据服务器如何与主机进行通讯,主机通过 GTP 协议从数据服务器下载数据。SMTP 和 GTP 都基于 TCP 协议,因此它们都能够保证可靠性传输。

在此协议结构平台上,目前实现上述协议功能的消息一般采用兴趣表达式来表示,如 SRP 中的 SUBSCRIBE 消息^[5]等。当消息到达主动路由器时,路由器提取去除 UDP 或 TCP 首部的兴趣表达式信息。如果是订购消息,则在主动路

^{*} 本课题得到国家自然科学基金项目(60473113)、国家自然科学基金重点项目(60533080)资助。徐丹 硕士研究生,主要研究领域为分布式虚拟环境、Agent 技术;陈继明 博士研究生,主要研究领域为 XML、分布式虚拟环境;潘金贵 教授,博士生导师,主要研究领域为多媒体信息处理、多媒体远程教育系统。

由器端以一定的格式存储对应用户用于订购的兴趣表达式,例如维护一个二维列表等;如果是发布消息,则将其与发布订购区域兴趣表达式进行匹配,当且仅当所有兴趣表达式都匹配时,才可通过主动路由器,沿转发路径向下一级路由器或终端传输。

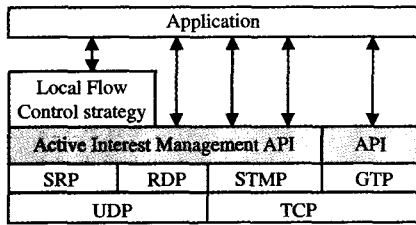


图1 主动兴趣管理协议结构图

1.2 兴趣表达式

在分布式事件模型的发布-订购模式中,比较常见的是采用由多个属性值或值域组成的结构来表示订购区域,即为一个兴趣表达式。一组属性值集合中的每一个属性都采用一个三元组来表示^[6],包括属性类型(AttType)、属性名称(AttName)和属性值(或值域)(Att(Value or Range)),如图2所示。兴趣表达式为一组相关属性的三元组集合,采用二维列表形式表示。同一个兴趣表达式中的每一个属性条件之间是一种逻辑与的关系,只有满足所有属性条件的数据才是订购范围内的数据,才能进行转发。图3给出了一个基于兴趣表达式的动态信息的描述:发送者“xudan”向房间号为“302”内的接收者“chenjiming”发送一个位置变更信息,数据类型为“Dynamic”,消息类型为“move”,消息包含的具体内容为发送者“xudan”向“south”移动了“12”步。其中 roomNum 和 receiver 是用以过滤的属性值,只有满足所有这些属性的更新消息才能最终通过主动路由器。

过滤机制一般是在主动路由器端完成,当它接收到数据包时,根据数据包中的属性决定是否转发。通过对冗余信息的过滤,很好地降低了网络传输量。对于静态消息,其属性值相对固定,兴趣表达式能够较为清晰地表达。但是对于虚拟空间中的动态交互信息,随机性比较大,不同的应用需要不同的属性个数。即使同一个属性值也有可能安排在不同的地方,与具体应用的初始设置密切相关。在基于兴趣表达式的动态数据包中属性的数量以及存放位置都是由系统事先定义好的,是针对具体应用情况的。但对相同的应用来说,想要在运行时刻改变属性的个数或者存放位置都是相当困难和低效的。因此,对于动态交互信息的表示主要存在以下的缺陷:

(1)表达力。在描述同一层次的数据属性时,通过顺序的形式来表达一系列的用户兴趣。但当要表达用户兴趣之间的某种关联,譬如嵌套等,这种表达方式就变得异常困难,即该表达方法很难适应结构化和层次化属性的描述。而在动态交互信息中,嵌套是很常见的关系。

(2)通用性。由于数据包中的属性是系统事先定义好的,针对某种特定的应用,属性相关部分的处理都是通过硬编码的方式进行组织。因此,当具体应用发生变化时,客户端、主动路由器端与数据包生成以及处理相关的代码都需要重新构建。动态交互信息是经常发生变化的,且其还可能被用于不同DVE系统的互连交互。由于开发人员的不同,各个系统定义属性的方式以及属性的位置都会有所不同,要实现系统间的互连相当困难。因此这种数据定义方式使得主动兴趣管理

中动态信息的通用性较差。

AttType	AttName	Att(Value or Range)
---------	---------	---------------------

图2 订购区域的属性结构

AttType	AttName	Att(Value or Range)
string	dataType	“Dynamic”
string	messageType	“move”
string	sender	“xudan”
string	receiver	“chenjiming”
float	roomNum	302
string	orientation	“south”
float	stepNum	12

图3 兴趣表达式的一个例子

2 任务表示法

2.1 任务表达式

分布式虚拟现实场景中的静态数据,其表示相对而言比较固定,大多数表达式都能很好地将静态数据表达清晰。而动态数据则比较多变和灵活,如果采用兴趣表达式来表示的话,就会出现上述缺点。所以,针对兴趣表达式在表达力方面的不足,我们将任务表示法^[7]引入到分布式虚拟环境中。任务表示法是基于动作(action)的表达式^[7,8],接近于自然语言,表现力强,便于理解,与ACL表示形式^[9]类似,但更具有层次感,所以可以很好地表达DVE中的一些虚拟人的交互任务^[10]。具体的形式一般由一个关键字(即动作名称)开始,这个关键字指明消息的类型,属于哪类动作,例如“move”就表示这是一个移动消息,这个任务的属性值集合采用“(“和”)”来封装,中间的属性值可以用嵌套形式表现子属性值,如图4所示。“move”的属性值“orientation”具有自己的子属性,这种表示层次式的属性关系是兴趣表达式无法表达的。基于网络上传输的信息量要尽量精简的原则,任务表达式显然不会为了可理解性而采用信息量多的自然语言表示。图4所示的形式已经是最基本的信息格式了,无法再做进一步分解。

```
move (sender, receiver, roomNum, orientation (Num))
```

图4 任务表达式的一个具体实例

我们在已开发的虚拟博物馆系统中采用了任务表达式来表示动态信息,有效地提高了系统的效率和可扩展性。虚拟博物馆是一个分布式虚拟环境系统,用户可以在虚拟博物馆中走动,参观藏品,获取藏品详细参数,以及进行交谈等活动。以下列出了系统中利用任务表达式来表示的通信协议:

- 注册协议:


```
register(sender, receiver, roomNum)
registerAccept(sender, receiver, roomNum)
tellNewVisitor(sender, all, roomNum)
```
- 移动协议:


```
move(sender, receiver, roomNum, orientation(Num))
moveCross (sender, receiver, iniRoom, desRoom, orientation
```

(Num))

• 谈话协议:

textChat(sender, receiver, roomNum, content)

textChatAll(sender, all, roomNum, content)

• 查询协议:

askDetail(sender, receiver, roomNum, stash)

tellDetail(sender, receiver, roomNum, stash, content)

• 注销协议:

unregister(sender, receiver, roomNum)

replyUnregister(sender, receiver, roomNum)

当一个虚拟人用户想进入博物馆进行参观时,首先需要向服务器发送注册信息(register),服务器返回接受信息(registerAccept),并通知所有同一个房间的用户(tellNewVisitor)。注册完成后,就能在虚拟博物馆中获得相应的功能。当用户离开时,需要发送注销消息(unregister),服务器返回响应信息(replyUnregister),并通知同一房间用户有人离开(tellNewVisitor)。值得注意的是,由于不同房间的用户无需知道其它房间的用户位置等信息,采用房间号属性的方法可以首先过滤一大部分不必要的网络冗余消息开销,极大降低了网络流量。

虽然任务表达式能很好地表示虚拟现实中的动态交互信息,但是这种形式无法很好地在网络上传播,主动路由器也很难对这种任务表达式表示的协议进行主动过滤和路由,所以需要对其进行相应的封装和解析,使其更好地适用于主动网络。

2.2 任务表示形式的层次式结构

任务表达式能表达通信协议中的层次式关系,潜在的有一种层次式结构,在我们开发的虚拟博物馆系统中将数据分为静态数据和动态数据。对于动态数据的任务表达式形式,在层次式结构中将动态数据作为其父元素,以消息类型将其按照系统所定义的协议数来做出适当的划分,将其置为动态数据的子属性;每一个类型的消息都有自己的属性值集合,是执行该动作所必需的信息,其中可以包括一些用以过滤的信息,也可以是一些参数值。而消息中的属性可能还包括它自己的子属性,这种嵌套关系正是任务表达式的优势所在。同样,对于静态数据,我们也可采用层次式结构来进行描述,相对动态较为简单。将静态数据置为父节点,子属性包括大厅、房间、走廊,根据场景中物体的位置来确定层次式关系,如展台在房间里面确定展台是房间的子属性。整体的层次式结构如图5所示。

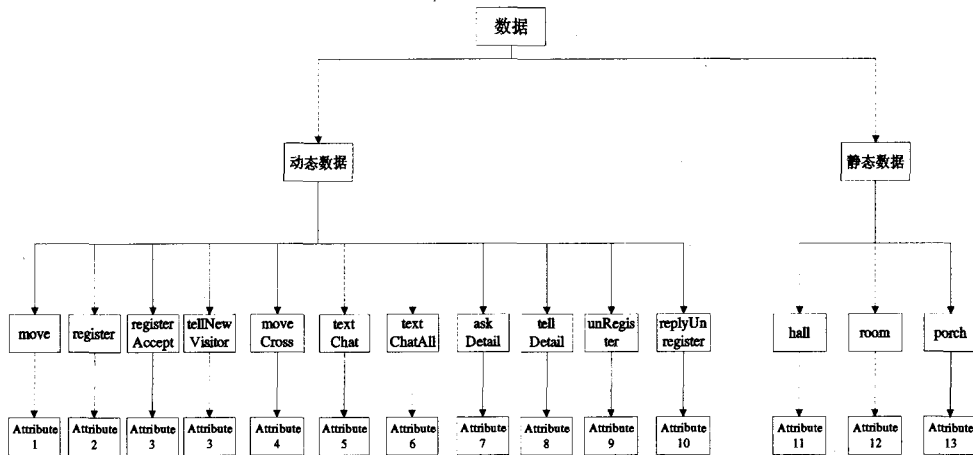


图5 任务表达式的层次式结构

基于上述整体的层次式结构,在描述一个参观者参观博物馆中展品的场景时,静态数据方面,描述房间302中1号展台上的化石,可以将房间302作为静态数据的子属性,1号展台作为房间302的子属性,化石在展台上,所以将其置为1号展台的子属性,该结构一共有5层。动态数据方面,描述图3用兴趣表达式表达的“move”消息,将“move”消息置为动态数据的子属性,对于发送者“xudan”,接收者“chenjiming”,房间号302,方向“south”都是“move”的子属性,置于“move”下一层,往南移动步数12为“south”的子属性,所以在“south”的下一层节点,整个结构也同样是5层(如图6所示)。

改,因此我们将采用XML对协议进行封装。

3 基于XML封装层次式协议

针对这种层次式关系结构,需要一种很好的封装方法。这种方法既要体现属性值之间的层次式关系,又要便于在网络上传输和路由。由于XML是一种自描述的、灵活的、可扩展的文本格式,允许将文档组织成层次式结构(包含根元素和子元素),元素还可以任意层次嵌套,很好地表示虚拟现实协议属性值的层次式关系;同时描述文件的多样性和灵活性使得定义的协议可以根据不同的应用进行属性的添加或更

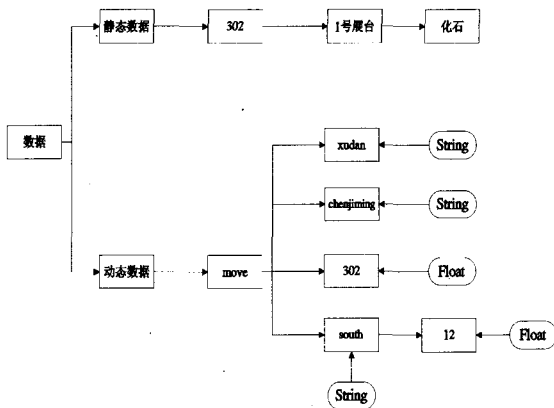


图6 具体实例的层次式结构

根据协议的层次式结构和XML文档的特点,我们将“动态”(Dynamic)或“静态”(Static)用<datatype>和</datatype>封装,将其置于根元素<message>的属性位置。而对于“move”等代表消息的类型的特征字,我们采用<messagetype>和</

message)进行封装,并使其作为“datatype”的属性也就是“message”的属性的属性。对于“sender”等属性,采用<messageparameter>和</messageparameter>进行封装,置于“message”的子属性。整个协议结构被组织成一个多层次树形结构,如图7所示。

主动路由器根据对应的 XPath 表达式能够迅速地找到用以过滤的属性值,如根据上面的例子为“roomNum”和“receiver”。“receiver”已能确定接收者,但是对于虚拟博物馆来说,位于不同房间的虚拟人只对本房间的其它参观者移动信息感兴趣,其它房间的人的位置的移动对于不在同一个房间的人来说是无效的,因为他们的位置和变化的影响都是不可见的。为了更有效地过滤数据信息,我们添加的“roomNum”属性虽然增加了少量网络流量,但可以过滤相当大一部分不必要的信息。

图7为图4描述的兴趣表达式的 XML 文档封装形式。封装形式在相应的封装标识后面跟上一个属性值,例如<message type value=“move”>,这是为了使得格式上更加具有层次感,能够很方便地看出属性间的层次式嵌套关系。对于想要利用网络上现有的资源时,它提供了一个“link”命令,只要在后面跟上网络地址就能直接引用,如文档中的<sender link=“http://www.cs.nju.edu.cn/~xudan”>。整个文档内容描述非常清晰,无需再多做解释。对于静态数据也能根据其层次式关系利用 XML 进行合理的封装,在此就不描述了。

```
<?xml version='1.0'?>
<!DOCTYPE task_rep SYSTEM "task_rep.dtd">
<message>
  <datatype value = "Dynamic">
    <message type value = "move">
      <messageparameter>
        <sender link="http://www.cs.nju.edu.cn/~xudan">
          xudan
        </sender>
      </messageparameter>
      <messageparameter>
        <receiver link="http://www.cs.nju.edu.cn/~chen">
          chenjiming
        </receiver>
      </messageparameter>
      <messageparameter>
        <roomNum>
          302
        </roomNum>
      </messageparameter>
      <messageparameter>
        <orientation value = "south">
          <stepNum>
            12
          </stepNum>
        </orientation>
      </messageparameter>
      <messageparameter>
        <language>
          Task representation
        </language>
      </messageparameter>
    </message type>
  </datatype>
</message>
```

图7 move 的 XML 封装形式

同时,XML 路由器硬件的出现和基于 XML 路由算法的相继产生又为交互协议的 XML 封装提供了有力的支持。正如 XML 特性中描述的一样,它是与 XML 路由器和 XML 路

由算法相适应的,主动路由器根据 XML 文档相应的 XPath 进行消息过滤。XML 封装后的协议结构不仅弥补了兴趣表达式表示的缺陷,而且使得主动兴趣管理完全过渡到 XML 领域。将 XML 路由器硬件应用到 DVE 场景中,拓展了 DVE 的发展前景,DVE 系统中主动兴趣管理的效率得到了很大的提高,相应的网络利用率也得到了很大的改善。

传输协议采用了 XML 形式的封装,而系统应用层能够执行的协议是基于任务表达式形式的,需要通过一个解析器来完成相应的封装和解析的过程。这个开销相对于现在的主机性能来说是相当小的,完全在可以接收的范围内,且利用主机的一小部分开销来换取网络上庞大的数据流量是相当值得的。目前在 DVE 系统应用中,网络上需要传输的流量越来越大,时延的问题也日显突出,对于协作的项目只有在一定阈值以内才能很好地完成。因此,如何改善网络流量,使其突破瓶颈才是最重要的。目前我们已经开发了简单的 XML 静态和动态解析器,用以在发送端将任务表达式封装成 XML 文档形式,在接收端将 XML 文档还原为任务表达式形式,以便用于虚拟环境的应用层处理。

结束语 本文针对分布式虚拟环境中兴趣表达式形式对于动态信息的表达力和灵活性的不足,提出了以任务表达式来封装动态交互协议的方法,并且根据任务表达式潜在的层次式结构,引入 XML 文档封装形式,使其便于在网络上传输和进行主动路由,与相应的 XML 路由器和路由算法相匹配。同时结合虚拟博物馆系统给出层次式结构图和 XML 封装文档,开发了一个应用于包装和解析的 XML 解析器,体现出了 XML 在分布式虚拟环境中的优越性和广阔前景。下一步的工作着重于将解析器不断地完善,并将其移植到我们已开发的 AIMNET 分布式平台下,使其与 XML 路由算法和主动路由网路衔接起来,具有更好的可扩展性。

参考文献

- 1 潘志庚,姜晓红,等. 分布式虚拟环境综述. 软件学报,2000,11(4):461~467
- 2 翟磊. 基于 XML 路由的主动兴趣管理研究:[学位论文]. 南京大学,2006
- 3 Leung Elvis Wai Chung, Li Qing. XML-based Agent Communication in a Distributed Learning Environment. In: International Conference on Web-Based Learning, 2004. 136~146
- 4 Grosfof B N, Labrou Y. An Approach to using XML and a Rule-based Content Language with an Agent Communication Language. In: Proceedings of the IJCAI-99 Workshop on Agent Communication Languages (ACL-99), May 1999
- 5 林诚凯,李惠,等. 分布式虚拟环境 AIMNET 的关键技术概述. 系统仿真学报,2004
- 6 孙元浩. 可扩展主动兴趣管理:[学位论文]. 南京大学,2003
- 7 Ciger J. Collaboration with Agents in VR environments: [PhD Thesis]. Swiss Federal Institute of Technology, 2005
- 8 Vosinakis S, Panayiotopoulos T. A Task Definition Language for Virtual Agents. Journal of WSCG, 2003, 11:512~519
- 9 Huang Zhisheng, Elirens A, Visser C. 3D Agent-based Virtual Communities. ACM Press, 2002. 137~144
- 10 Kallmann M. Object Interaction in Real-Time Virtual Environments: [PhD Thesis]. Swiss Federal Institute of Technology, 2001