

# 基于 CORBA 事件服务的主动服务技术研究与实现\*

郭长国 刘江宁 邹鹏

(国防科技大学计算机系 长沙 410073)

**摘要** 介绍了主动服务及其需求背景,文章讨论了基于 CORBA 事件服务实现主动服务的系统框架,并简要介绍了该在电信管理中,基于事件服务的主动服务的应用。

**关键词** 主动服务,推,拉,事件服务,电信

**分类号** TP311

## The Study and Implementation of Active Service Based on CORBA Event Service

Guo Changguo Liu Jiangning Zou Peng

(Department of Computer Science, NUDT, Changsha, 410073)

**Abstract** This article introduces the active service, discusses the system framework based on CORBA Event Service. And it presents the system's application to telecommunication.

**Key words** active service, push, pull, event service, telecommunication

近年来,主动服务技术受到了愈来愈多的关注。从用户的角度讲,主动服务就是在用户不发出指令的情况下把信息数据放到自己的系统中。和主动服务系统密切相关的技术是“推”(PUSH)技术。推是相对于传统的 Client/Server 结构而讲的,是由服务器把信息主动推给客户。服务器是主动的,信息传输的发起者是服务器,而不是用户的请求。相对地,我们将“请求/应答”模式的信息传递称做“拉”(PULL)。“推”最大的特点是服务器的主动性。服务器根据设定好的触发事件,一旦被触发,便根据用户所提交的“兴趣”描述来发送相应的信息。我们这里所讲的“推”是从 Client 和 Server 在数据传输中的地位来定义的,并不是用户感觉到的表象。

## 1 主动服务技术的需求背景

主动服务技术在监控系统和信息发布方面有着广泛的应用背景。

在一个集成化的企业 Intranet/Extranet 网络中,在实时系统和其它特别追求时效的系统中,决策时间的拖延将意味着巨大的风险,甚至产生灾难性后果。因此,必须针对管理者的要求及时将管理信息上报,以便决策使用。传统的“请求/应答”模式的系统则无法保证对事件的及时响应。

主动服务技术的发展也同 Internet 的发展息息相关。面对 Internet 这个巨大的信息海洋,访问者难以找到其感兴趣的内容,即使找到了自己所要的信息,也无法对这些信息变化进行自动跟踪和监测。

信息时代的发展,迫切需要主动服务技术:人们向信息中心提交自己需求,“预定”信息,一旦信息到达或者信息变更,就主动将信息“推”给客户,这样客户得到的始终是最新信息。

## 2 主动服务系统的实现途径

目前的主动服务系统多为“出版/订阅”(Publisher/Subscriber)型。接收信息的用户叫订阅者(Subscriber),发送信息的用户叫出版者(Publisher)。依据出版者和订阅者的主动性,可以分为三种:

- 以出版者为中心的主动服务系统。

\* 国家 863 项目资助  
1998 年 6 月 29 日收稿  
第一作者:郭长国,男,1973 生,硕士

出版者处于主导地位, 订阅者只有有限的主动性, 它向信息中心“预定”信息。订阅者一般处于“睡眠”状态, 主动服务器将信息“推”过来时, 才唤醒它。这是一种纯粹的解决方案, 但效率不高, 因为出版者将信息“推”过来时, 无法预测用户系统的工作情况。

- 以订阅者为中心的主动服务系统。

实际上还是传统的 Client/Server 结构。订阅者周期性地查询, 如有信息或信息更新则将信息“拉”回来, 表现给用户一种“推”的假象。过于频繁的查询可能会导致网络和服务器的饱和, 降低系统的性能, 甚至无法工作。这种系统在对实时性要求比较高的系统中是不适宜的, 但它赋予用户更大的自主性: 用户可以设置“拉”的间隔和时间, 从而提高网络的利用率。

- 出版者、订阅者相结合的主动服务系统。

将上述两种进行结合: 预定结束后, 订阅者就处于“睡眠”状态, 如果有信息或信息更新, 出版者通知客户, 而不是将信息全推过来。由用户来决定何时从出版者处将信息拉回。这样一方面不会丢失信息, 另一方面也可以给用户充分的自主性

### 3 CORBA 事件服务

1991 年和 1995 年, 国际对象管理组织(Object Management Group, OMG) 先后发布了基于分布对象技术的公共对象请求代理结构(Common Object Request Broker Architecture, CORBA) 1.0 和 2.0, 为分布异构环境下各类应用系统的集成提供了良好的可遵循的规范和技术标准<sup>[1]</sup>, 有力地推动了客户/服务器计算机系统向更深的方向发展。事件服务(Event Service) 规范<sup>[2]</sup> 是其 15 个基本服务之一。

一个主动服务系统的关键是其对推和拉的支持。事件服务对主动服务技术提供了强有力的支持。事件服务允许对象动态登录和退出。它在对象之间产生一个透明的“生产者/消费者”模式的通信信道。生产者的事件发送到消费者而生产者和消费者不必显式的知道对方。事件服务按照对象在事件中的不同角色把它们分成三种: 提供者(Supplier) 和消费者(Consumer) 和事件通道。如图 1 所示:



图1 事件服务结构图  
Fig.1 Event service components

在提供者和消费者之间有两种方式来进行通讯: “推”和“拉”。在“推”模式中, 提供者是数据传输和通讯的主动者, 它将事件数据送给消费者; 在“拉”模式中, 消费者是主动的, 它从提供者请求事件数据。事件通道是一个中间对象, 它允许多个提供者和多个消费者异步地进行通讯。对提供者而言, 它是事件的消费者; 对消费者而言, 它是事件的提供者。

事件通道通过代理(Proxy) 对象来负责通信。事件消费者和事件提供者在通信之前, 都必须获得一个代理。消费者获得一个提供者代理, 提供者获得一个消费者代理, 代理之间来完成信息的交换。

事件服务定义了一系列的接口和操作来实现通信。接口可以分成两类: 通道代理的管理接口和代理接口。前者包括 ConsumerAdmin、SupplierAdmin 和 Event Channel; 后者包括 ProxyPushConsumer、ProxyPullConsumer 和 ProxyPushSupplier、ProxyPullSupplier。事件的发布和接收则通过实现 Push-Supplier、PullSupplier 和 PushConsumer、PullConsumer 的接口来完成。

我们按照 COSS 规范实现了事件服务, 我们事件服务结构如图 2。

在我们实现的事件服务中, 提供者管理对象和消费者管理对象各维护着两条“推”和“拉”代理的队

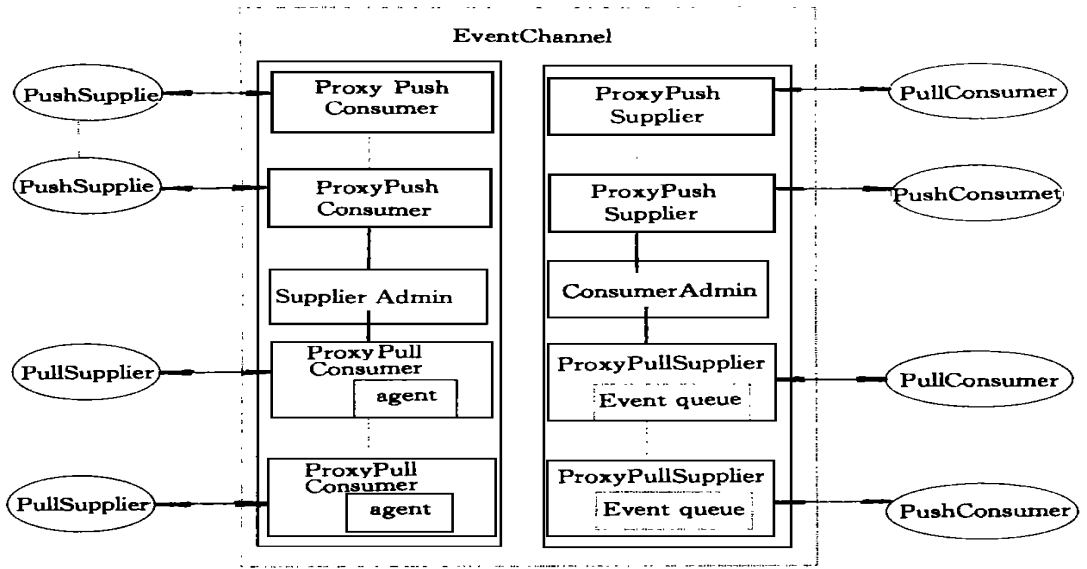


图 2 事件服务总体结构图

Fig. 2 Structure of event service

列, 每个代理对应一个提供者或消费者。对于“推”的提供者代理(ProxyPushSupplier), 可以将事件实时推送到它对应的消费者(PushConsumer), 因而不需要保留事件; 对于“拉”的提供者代理(ProxyPullSupplier), 只有它对应的消费者(PullConsumer) 请求事件时, 才能将事件提交, 因而它必须有一个事件队列来缓存事件。

图 3 是一个具体的交互实例, 展示了其接口的应用。

事件提供者首先获得获得一个提供管理者(SupplierAdmin) 对象, 然后获得一个事件消费者代理(ProxyPushConsumer) 并同代理消费者建立连接。同样, 事件消费者也要同一个提供者代理(ProxyPushSupplier) 建立连接。连接完成后, 提供者可以开始由通道传输事件。为传一个事件, 提供者查询 ConsumerAdmin 上的所有代理, 在所有 ProxyPushConsumer 之上简单地调用 push 方法, 事件通道然后在所有在此 proxy 上登记的消费者之上调用 push, 这时所有的 PushConsumer 都会受到事件; 对所有的 ProxyPullConsumer, 则将事件保存在各自的事件队列中, 如果 PullConsumer 要获得事件, 它必须调用 ProxyPullConsumer 的 pull 方法才能将事件从事件队列中拉走。

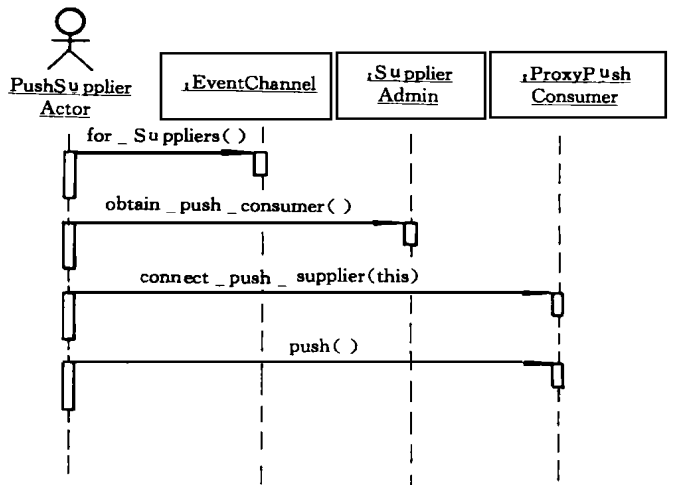


图 3 事件服务交互图

Fig. 3 A push event scenario

由于事件通道这个中间对象的存在, 使得事件服务可以非常方便的支持推/拉模式的混合通讯。这对于实现主动服务系统是至关重要的。可见, 事件服务在通讯模式上对主动服务系统提供了非常好的支持。事件服务在发送事件之前, 连接的建立非常的复杂, 这主要是为了使其可以应用的不同的领域, 因为不同的领域可能需要不同的通讯模式。事实上, 事件服务可以支持四种不同的通讯模式。对许多简单的

系统来讲,它可能显得过于灵活了<sup>[4]</sup>。

## 4 基于事件服务的主动服务框架

图4是一个基于事件服务的主动服务系统框架。该系统框架有三部分组成:出版管理器、信息出版者和信息订阅者。

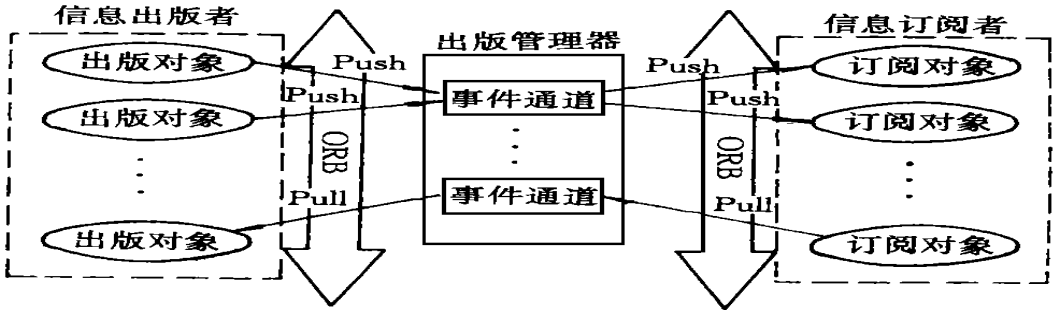


图4 主动服务系统框架

Fig. 4 Framework of Active Service

出版管理器管理若干通道,它的主要负责创建通道,一个通道代表某个信息类或被监控事件;它接收信息出版者提交的信息或信息描述,进行整理分类,将其归入不同的通道,并依照信息描述,向其索取信息。管理者还向信息订阅者发布信息描述,或依照信息订阅者提交的信息预定表,将各类最新信息推送给不同的订阅者。

信息出版者可以直接将自己的信息提交给出版服务器,也可以只提交信息描述,出版服务器会按照提交的信息描述来索取信息。

信息订阅者可以用预定的方式提交自己的兴趣描述,出版服务器会依照兴趣描述将最新的信息随时推送给订阅者。信息订阅者也可以随时查询出版服务器来获取自己感兴趣的信息。

该系统框架基于CORBA分布对象计算标准,使该系统框架具有较好的开放性和分布性,可以很好的实现同现有系统的集成和互操作。建构在事件服务之上,使得信息的出版者和信息的订阅者彼此互不影响,并且交互方式可以灵活采用推或拉任何一种模式。这样就赋予了信息出版者和信息订阅者极大的灵活性和自主权——它们可以自主决定用推还是用拉或推拉结合来实现主动服务。事件服务允许提供者和消费者动态的登录和退出,这样就使得该系统在规模上具有很好的伸缩性。

## 5 小结

自主开发的遵循CORBA 2.0规范的分布对象计算环境——StarBus,实现了事件服务,开发了电信网络运行维护管理的推拉结合的主动服务系统。在该系统中,涉及远程通讯(经邮电通讯网络)和异构平台之间的互操作问题。本地网管采用SCO UNIX平台、Sybase数据库;省级网管采用NT平台、Informix数据库;客户端采用Windows平台,整个系统采用C++ORB和JavaORB互连,是一个典型的分布异构环境。在该系统的管理服务器上运行了两类事件通道:紧急事件告警通道和工单下达通道。同时在管理服务器上还运行相关事件的处理对象。该系统已在多家电信部门得到应用,发挥了显著作用。实践表明上述基于分布对象技术的主动服务系统框架为分布式监控管理系统的开发提供了强有力支持。

## 参考文献

- 1 OMG. The Common Object Request Broker: Architecture and Specification, 2.2 ed., 1998
- 2 OMG. CORBA Services: Common Object Services Specification, 1995
- 3 Orfali R, Harkey D, Jeri Edwards. Instant CORBA. John Wiley & Sons, INC, 1996
- 4 Schmidt D C, Vinoski S. Object Interconnections—The OMG Events Service (Column 9). SIGS C++ Report, 1997