

# 网管环境下连接高可用性软件总线的研究与构造

李治文<sup>1</sup>, 杨正球<sup>1</sup>, 齐峰<sup>2</sup>, 孟洛明<sup>2</sup>

(1. 北京邮电大学程控交换技术与通信网国家重点实验室, 北京 100876; 2. 北京邮电大学信息产业部网络管理重点实验室, 北京 100876)

**摘要:** 指出了网管环境下高可用性系统的不足, 提出了高可用性连接和高可用性管理域的概念. 分析了软件总线的基本概念, 提出了软件总线的连接模型、实现模型和网络实现模型. 在提出的软件总线理论上构造并实现了一种基于消息机制的支持连接高可用性的软件总线模型. 该模型系统在全国移动汇接网管项目中得到成功应用.

**关键词:** 网络管理; 高可用性; 软件总线

**中图分类号:** TN913.2      **文献标识码:** A      **文章编号:** 0372-2112(2001)10-1332-05

## The Study and Construction of High Availability Connectivity Software Bus in Network Management Environment

LI Ye-wen<sup>1</sup>, YANG Zheng-qiu<sup>1</sup>, QI Feng<sup>2</sup>, MENG Luo-ming<sup>2</sup>

(1. National Lab of Switching Technology and Telecommunications Networks, Beijing University of Posts and Telecommunications, Beijing 100876, China;

2. Ministry of Information Industry's key lab of networks management, BUPT, Beijing 100876, China)

**Abstract:** The shortcoming of the existing high availability system in network management environment (NMS) is pointed out. New concepts of high availability connectivity and high availability management field in NMS are proposed. Basic concept of software bus is analyzed. The link model, implementation model and network implementation model of software bus are presented. A high availability connectivity software bus based on message mechanism is designed and implemented. This model has been used in National Mobile Network Management System of China successfully.

**Key words:** network management; high availability connectivity; software bus

### 1 引言

网络管理系统是保证网络正常、经济、可靠运行的重要基础. 因此, 网管系统本身的可靠性早就引起了人们的注意. 网管系统的一个特点就是系统中存在需要进行大量通信的各种分布式应用, 所以保证它们之间的通信可靠性就成为保证网管系统可靠性的一个主要任务. 目前提高网管系统可靠性的方法主要是采用容错计算机系统和高可用性(HA, High Availability)<sup>[1]</sup>系统. 前一种方法可靠性高但是价格昂贵, 网管系统不需要如此高的容错要求, 一般采用 HA 系统. HA 技术是一种可靠性指标可控制的技术, 它采用冗余技术但又不同于容错系统, 它不保证器件级的容错, 而是采用大量供应的一般软硬件的集成, 以保证整体如服务方(包括软件和硬件服务方)的可靠性. 它的容错粒度比容错系统大, 具有很好的性能价格比、且较易实现, 是目前应用广泛的一种方案.

普通的 HA 服务模型是通过在服务器端提供多个功能相同的服务方, 使得系统中服务方满足冗余要求. 当服务方发生故障时, 系统快速检测到故障, 并透明地进行服务方迁移, 使应用能够继续工作, 这种模型称为多服务器模型. 从理论上分

析 HA 系统的可用度<sup>[2]</sup>, 可以发现在 HA 系统中通过采用服务方冗余技术, 可使系统的可用度大大提高, 但是没有超过网络资源的可用度. 从上面的分析可知, 普通的 HA 服务模型主要是通过保护服务方而提高系统的可靠性, 而没有保证应用之间的连接可靠性. 在网管环境下, 网管应用之间的连接不可靠性可能导致网管系统的数据不完整、网管质量降低. 针对这个问题, 本文提出高可用性连接和高可用管理域概念, 并给出了软件总线的理论框架, 包括软件总线的概念、连接模型、实现模型和网络实现模型. 在此理论上设计并实现了一种基于消息机制的支持连接高可用性的软件总线.

### 2 高可用性连接概念

本文提出的高可用性连接, 是将容错技术中的高可用性概念应用到网管环境中得出的一个新概念. 它是指两个需要通信的网管应用之间的连接高可用性, 这种连接可以是逻辑的或是物理的. 在复杂的网管异质环境下, 需要通信的应用可能是工作在不同的软硬平台之上, 它们之间的通信涉及到大量复杂的细节, 如操作系统特性、系统字长度、字节高低位

方式、进程号、网络地址、网络协议和网络参数等. 应用之间通信时, 所有这一切相对它而言都应该被隐藏起来, 应用只需知道它要通信的另一方的名字即可完成通信, 通信连接的细节对它透明. 同时, 对各种异常情况造成的连接中断, 系统都能妥善处理使应用感觉连接是一直正常保持的. 这对在动态环境中的复杂且变化多样的网管应用来说, 意义是重大的. 总之, 连接高可用的系统应该满足以下透明性: 连接过程透明、位置透明、访问透明、重定位透明、失败透明、迁移透明、连接持久性透明. 另外, 系统必须采用各种措施, 在各种通信连接异常情况下, 在设计规定的范围内保证通信数据的有效性和完整性直至通信连接恢复将数据完整送达.

### 3 软件总线的基本思想

在计算机体系中, 硬件总线是一组以铜铂线为载体的信号传输线, 它是计算机各部件间及输入输出设备间传送各类信息的公共通道. 硬件总线技术和软件模块化的发展, 导致了软件总线<sup>[3]</sup>的产生. 对软件总线人们已有相当研究成果, 如由 13 个西欧国家联合完成的 Eureka 工程计划, 就设计了一种软件总线 ESF<sup>[4]</sup>, OMG (Object Management Group) 组织也有对象请求代理 ORB (Object Request Broker)<sup>[5]</sup>. 但它们在介绍软件总线时, 只是表述了软件总线的目标, 即分布式异构环境下的一种透明通信的集成工具, 或称之为透明请求和响应的对象总线. 而软件总线的较完整的定义及框架在公开的文献中一直没有很系统地给出. 北京邮电大学程控交换技术与通信网国家重点实验室和信息产业部网络管理重点实验室在承担全国性大型网管建设任务过程中, 为实现连接高可用的目标, 为保证网管建设的平滑过渡, 最大限度地继承现有技术和包容已有系统, 对软件总线进行了大量的研究, 形成了一套软件总线理论, 并据此设计了支持连接高可用性的软件总线. 下面先介绍我们给出的软件总线理论.

#### 3.1 软件总线的概念

设计一个软件系统的关键是设计该软件系统的系统结构, 而设计系统结构的关键问题是如何处理组成系统的单元和单元之间的关系. 单元之间的关系是系统结构中的核心问题. 由于单元之间的关系十分复杂, 所以处理单元之间关系的技术也比较复杂并且处于不断发展的状态之中.

传统的处理单元之间关系的方法是定义软件单元之间的接口, 由于单元之间关系的复杂性, 导致应用单元之间接口或非常复杂, 或适应性较差. 软件总线的基本思想是避开单元之间关系的复杂性, 不去直接定义软件单元之间的接口, 而是引入一个“连接件”. 连接件的概念如图 1 所示.

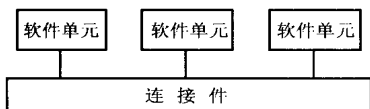


图 1 连接件的概念

受计算机系统硬件总线技术的启发, 如果能象硬件总线处理硬件插板之间关系一样来处理软件系统单元之间的关系, 那么软件系统的适应性和扩展性将会大大地提高, 且技术

复杂度也相应得到控制. 因此, 如果从总线的角度来看连接件, 那么连接件就可被视为“软件总线”. 在软件总线的概念基础之上, 就可以定义软件单元和软件总线之间的接口. 采用总线接口方式带来的好处是可以将处理复杂的软件单元间关系的工作转化为处理软件单元和软件总线之间相对简单关系的工作. 这种处理单元之间关系的方法也可称为软件总线, 其概念如图 2 所示.



图 2 软件总线的概念

#### 3.2 软件总线的连接模型

由于软件总线不能象硬件总线那样仅提供一个数据交互的通道, 因此, 必须安排相应的软件实体来实现总线的功能, 由此得到软件总线实现的概念.

软件单元和软件总线功能实体之间的接口称为软件总线接口. 为了提高软件总线接口的适应能力, 可以采用 OSI 参考模型中间层服务模型作为软件总线接口的连接模型. 为了减少软件总线接口实现的复杂性, 可以对 OSI 参考模型中间层服务模型进行简化, 只使用其中的请求 (Request) 和应答 (Response) 作为软件总线接口的连接模型, 如图 3 所示. 由于请求和应答是两个实体间的关系, 因此, 软件总线功能实体只是起到了一个请求和应答的转接站的功能.

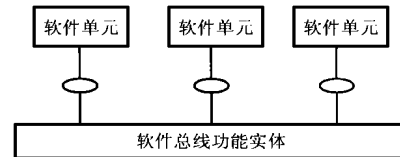


图 3 软件总线的连接模型. ○: 软件总线接口

#### 3.3 软件总线的实现模型

OSI 参考模型中间层服务请求和应答是两个实体之间的关系. 为了支持分布处理, 即为了实现连接到软件总线的软件单元间的互通, 软件总线功能实体应能提供通信方面的透明处理功能, 即支持软件单元间按名字进行访问. 为了做到这一点, 软件总线功能实体本身应采用分布处理, 其主要做法是:

(1) 在软件单元端安排一个软件总线功能实体的子功能实体, 即软件总线功能实体软件单元端处理模块 (ES, Element Side), 简称单元端模块.

(2) 在软件总线功能实体端安排一个软件总线功能实体的子功能实体, 即软件总线功能实体软件总线端处理模块 (BS, Bus Side), 简称总线端模块.

在安排了单元端模块和总线端模块后, 软件单元和软件总线功能实体, 采用请求/应答方式的接口方式就可以进一步减少复杂性而采用 API 接口. 单元端模块和总线端模块采用 API 接口配置方式的软件总线实现模型如图 4 所示.

#### 3.4 软件总线的网络实现模型

软件总线功能实体采用单元端模块和总线端模块结构的优点不仅是为软件单元上面的应用程序提供了 API, 方便了

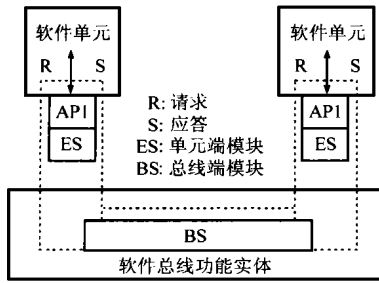


图 4 软件总线的实现概念

应用程序的开发和支持了应用程序的互通,它带来的最大好处是软件总线可以在网络环境下实现,而且这种网络环境下的实现对软件单元上面的应用程序是透明的,即软件单元上面的应用程序不知道 ES 和 BS 是否是在网络环境下.只要这个网络环境提供对各种硬件和软件平台的支持,就能支持软件单元上面的应用程序在多种硬件和软件平台上跨平台的互连.这样的互连和在多厂商环境下网管系统的互连是一个同样的问题.软件总线网络实现的概念如图 5 所示.软件总线的网络实现可以是全部的网络实现和部分网络实现相结合.即有一部分单元端模块和总线端模块是通过网络连接的,也还有一部分单元端模块和总线端模块可以不通过网络连接.

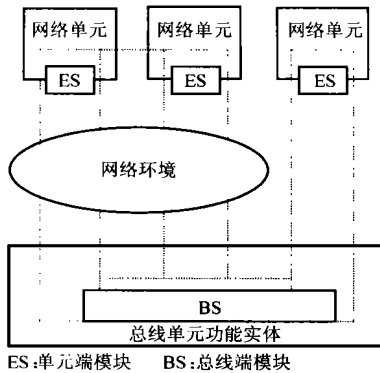


图 5 软件总线网络实现的概念

### 3.5 面向对象技术的引入

虽然面向对象技术还有许多不完善的地方,如理论基础不完善、方法论支持不足等,但其具有的数据抽象、继承、封装等技术,已经给软件的设计和开发带来了很大益处,特别是封装技术在网管系统中的使用,为多厂商设备环境下网管系统的互通提供了较好的技术支持.在软件总线中引入面向对象技术后,将会提高软件总线的灵活性、开放性和可维护性.

## 4 支持连接高可用的软件总线

根据软件总线理论的研究成果和网管环境下连接高可用的需求,本文构造了网管环境下基于消息机制的支持连接高可用的软件总线,简称 HA 软件总线.下面介绍其结构特点.

### 4.1 系统结构

图 6 描述了建立在 HA 软件总线网之上的 HA 软件总线系统结构.图 6 中,HA 软件总线由一个软件总线网构成,它为不同类型的网络 and 应用程序之间增加了一个统一的与具体网

络软硬件平台无关的层次,以此屏蔽掉了不同网络应用的底层细节,为不同平台上的应用提供了统一的网络接口.图 6 中,软件总线和直接注册在它上面的应用构成一个高可用管理域,整个系统由若干个高可用管理域构成.建立在 HA 软件总线网上的网管应用之间的通信都是通过软件总线进行的,应用之间建立逻辑的、HA 连接.实际的连接是由应用到软件总线和软件总线之间的连接构成.应用之间的连接高可用性由应用到软件总线的高可用性连接和软件总线本身的高可用性来保证.应用之间相互传送的是由消息头和消息体两部分组成的消息.

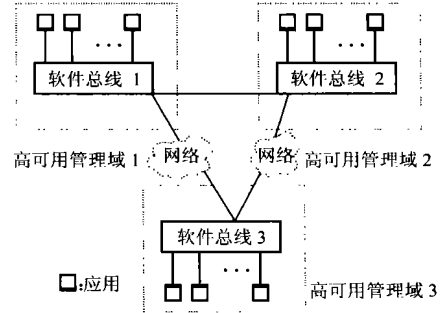


图 6 HA 软件总线系统结构

### 4.2 消息的 BNF 范式

因为任何交互行为都是通信实体之间的消息传递,因此需要定义一种具有普遍意义的灵活的消息结构.本文给出消息的 BNF 范式的定义如下:

```

< message> ::= < msg_ head> [ < parameterList> ]
< msg_ head> ::=
    < msgInvokeId> < msgType> < sourceName>
    < targetName> < parameterNumber>
    < encodeType> < escape>
< parameterList> ::=
    < parameterNameList> < escape>
    < parameterTypeList> * < escape>
    < parameterValueList> * < escape>
< nameList> ::= < identifier>
    { < escape> < identifier> } *
< parameterNameList> ::= < nameList>
< parameterTypeList> ::= < nameList>
< parameterValueList> ::= < nameList>
< msgInvokeId> ::= < identifier>
< msgType> ::= < identifier>
< sourceName> ::= < identifier>
< targetName> ::= < identifier>
< parameterNumber> ::= < identifier>
< encodeType> ::= < identifier>
< escape> ::= < character>

```

其中:

- (1) identifier 和 character 是基本的词法单位.
- (2) 消息分为固定部分和可变部分.其中固定部分对应于

msg. head, 可变部分对应于 parameterList. 在固定部分中指出了消息的类型, 消息的序列号, 消息的源和目的名, 消息按何种方式进行编码, 消息所含的参数个数, 消息的内部分割符. 消息的可变部分描述了消息的参数. 每个参数由名字、类型和值三部分组成. 为了保证系统的灵活性, 消息的类型和消息的参数类型可由应用程序自己定义.

#### 4.3 HA 软件总线的要素及特点

软件总线工作示意图如图 7 所示.

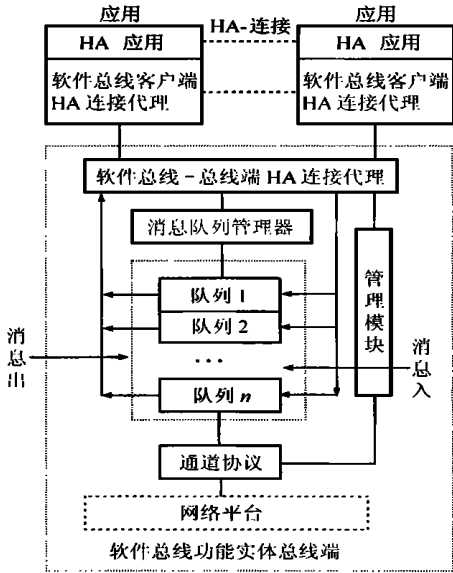


图 7 软件总线工作示意图

图 7 中, 软件总线的单元端模块和总线端模块被设计成 C/S (Client/Server) 结构, 这种 C/S 关系是一种相对变化的关系. 客户方是相对于提供服务的软件总线模块而言的. 在客户方, 应用建立在软件总线客户端-HA 连接代理之上. 客户端-HA 连接代理就是前面介绍的软件总线单元端模块, 它是软件总线的一部分, 以 API 的形式驻留在客户端, 用编程接口的方式为应用提供服务. 在本系统中, 它向应用提供了按名通信机制服务和消息处理服务.

图 7 中的软件总线功能实体总线端相对于客户而言是服务方, 服务方对应多个客户应用. 在服务方, 它有多个核心模块, 分别是软件总线-总线端 HA 连接代理 (BSHAA, high availability agent of BS)、消息队列管理器 (MQM, message queue manager)、应用的消息接收队列 (MQ, message queue)、管理模块 (MM, management module) 和通道协议 (CP, channel protocol) 等. BSHAA 执行多项功能, 包括安全管理、处理应用注册、取消注册、消息路由、发送消息、接收消息、广播服务、事件通知和管理连接等. 管理连接包括检查连接的可用性、总线主动向应用发起的连接恢复和保持总线之间的连接等. MQ 是指注册应用的接收消息队列, 它保存其它应用发送给它的消息. 传送的消息类型有信令消息和用户消息两种. 用户消息通过软总线被透明的存储和传送, 信令消息仅仅被软总线模块使用. MQM 管理 MQ, 并负责将消息提交给 BSHAA 发送给应用. BSHAA 依靠消息头记录中的路由信息将消息发往不同目的地的软件总

线 MQ 队列或是分布式环境下的异地 MQ 队列. MM 提供对各软件功能实体的综合管理功能, 包括查询和监督等. CP 是 BSHAA 与下面的特定平台的接口.

HA 软件总线具备消息中间件 (MOM, Message Oriented Middleware)<sup>[6]</sup> 的功能, 但又与 MOM 不同. 它支持网管复杂环境的要求, 拥有以下特性.

⑧ 支持广播、组播和事件通知 (event report), 适应管理者/代理互操作模式.

⑨ 对应用完全屏蔽了消息队列和队列管理器. 应用采用按名通信机制相互通信.

⑩ 消息队列对不同的操作系统和网络软硬件平台始终保持一致. 它应用于应用软件所处理的数据流中.

⑪ 相对于交互的高层网管应用而言, 应用之间建立逻辑的、高可用性的连接.

⑫ 软总线能够管理和应用之间的连接, 使其保持高可用性, 同时还具备与移动进程保持高可用性连接的能力.

⑬ 软总线采用分布式处理方式, 具有较强的容错性. 总线具备重启后主动恢复和应用的连接并保证消息完整性的能力.

#### 4.4 HA 连接的支持

系统中的应用启动后都必须向软总线注册. 注册内容包括全局唯一的应用进程名、对应的物理地址、接口参数等. BSHAA 要验证该注册的合法性, 对于认证合法的连接, BSHAA 要通知 MQM, 后者为验证合法的连接建立 MQ, 同时更新初始化时已建立的映射关系表. 该映射关系表内容有: 软件总线标识、应用进程名、消息队列名和应用接口物理参数等. 应用注册完毕后, 可以发送消息. BSHAA 接收消息并从消息头中获知该消息的目的地, 经查询映射关系表后, 把消息发给对应的 MQ. MQM 管理 MQ, 当它发现某 MQ 中有消息时, 向 BSHAA 查询与之对应的接收应用的通信连接的可用性, 在获得确认后, 将该 MQ 中的消息交付给 BSHAA 发送. 如此完成应用之间的一次单向通信. 为保证数据的完整性, 消息的发送和接收采用了发送单元端、总线端模块、接收单元端三方缓存技术. 在总线端模块, 消息可以存储在永久介质中如文件系统和数据库中, 同时被系统赋予一个永久的 ID (persistent ID) 号, 以此获得对应用崩溃和连接中断的容错性.

系统在工作过程中, 可能发生连接中断的情况. 例如, 应用主动迁移、通信终端、进程或通信网络物理连接的崩溃等都可能造成连接中断. 此时, 总线将保存 MQ 中的消息并可继续接收与此连接有关的其它方发来的消息. BSHAA 会设法恢复连接或重新选择路由或等待应用重新发起连接 (包括应用迁移完成和应用备份重启后发起的恢复连接). 对于后一种情况, 由于注册名字的全局唯一性, 总线通过查询映射关系表可获知该连接是否是某个中断连接的延续. 如果是, BSHAA 将发送事件通知给 MQM 和原来与该连接相关的总线单元端模块, 通知它们连接已恢复. 当连接恢复正常后, 系统会如前述过程继续工作.

#### 5 结束语

本文设计的高可用性软件总线模型已经成功地运用到全

国移动汇接网管系统的开发中. 该系统基于软件总线实现应用进程和通信进程分开<sup>[7]</sup>, 很好地满足了网管系统的要求, 从而证明了设计的合理性和实用性.

#### 参考文献:

- [ 1 ] Michael R Lyu. Software Fault Tolerance [M]. John Wiley & Sons Ltd. England, 1995.
- [ 2 ] 熊翱. 智能网中网络容错的设计与实现 [D]. 北京: 北京邮电大学图书馆, 1998.
- [ 3 ] M S Verrall. The software bus——its objective: the mutual integration of distributed software engineering tools [A]. Proceedings of 5th Conference on Software Engineering Environments [C]. Aberystwyth, March 1991.
- [ 4 ] M S Verrall. ESF: The software bus [A]. IEE Colloquium on Published [C]. 1991, Page(s): 4/1- 4/3.
- [ 5 ] Object Management Group. The Common Object Request Broker: Architecture and Specification (CORBA) [S], Version 2.0, 1995.
- [ 6 ] 黎杰. 中间件的概念及特点 [J]. 计算机工程与应用, 1999, 35 (增刊): 50- 54.
- [ 7 ] Zhengqiu Yang, Luoming Meng, Junwu Li. Design of a distributed computing platform BUPTDCP [A]. Proceeding APCG OECC99 [C]. Beijing, 1999, 1236- 1239.

#### 作者简介:



管理.

李治文 男. 1966 年出生. 1988 年毕业于江西大学数学系, 获理学学士学位. 1988~ 1996 年在江西电子仪器厂工作, 主要从事电能自动校验装置方面的工作. 1996 年进入北京邮电大学计算机学院攻读硕士学位, 1999 年获硕士学位, 同年进入北京邮电大学交换技术与通信网国家重点实验室攻读博士学位. 研究方向为通信软件与网络



杨正球 男. 1967 年 7 月出生. 副研究员, 中国电子学会会员. 1988 年 7 月毕业于北京大学计算机系, 获理学学士学位. 1991 年 3 月, 毕业于北京邮电大学, 获工学硕士学位. 1991 年至今在北京邮电大学工作, 主要从事程控软件支援系统, 智能网和 TMN 的研究工作, 先后参加了三十多项科研项目, 项目范围覆盖国家七五, 八五和九五攻关和 863 科研项目. 其中“VPN 业务管理的研究”和“电信管理网基础软件测试系统”获信息产业部科技进步一等奖; “SDH 传送网网管体系结构和信息模型研究”和“用户接入网管理功能与管理接口技术规范”获信息产业部科技进步二等奖; “基于 TMN 接口的 ATM 网网络管理所使用的管理目标定义”获北京市科技进步二等奖. 在国内外学术期刊及会议上共发表文章近二十篇, 出版译著/专著各一本.