

基于 CORBA 和 TL1 的 STP 网元集中管理系统实现

张 雪¹, 韩福丽², 倪桂强¹, 金凤林¹

(1. 解放军理工大学 指挥自动化学院, 江苏 南京 210007;

2. 武警北京指挥学院, 北京 100012)

摘 要: 设计了 STP 网元集中管理系统, 阐述了该系统的体系结构和各模块的功能以及各模块间的通信方式。通过使用 CORBA 构建技术、IDL 接口描述语言以及文本描述语言 TL1, 使得系统具有灵活性、可维护性和可扩展性的特点。该系统的实现, 使得对信令网中的 STP 设备进行集中管理和统一维护成为可能, 为进一步对信令网的检测和历史数据的采集分析提供了基础。

关键词: CORBA; TL1; STP 网元集中管理系统

中图分类号: TP311

文献标识码: A

文章编号: 1674-7720(2010)21-0079-03

Centralized management of STP based on CORBA and TL1

ZHANG Xue¹, HAN Fu Li², NI Gui Qiang¹, JIN Feng Lin¹

(1. Institute of Command Automation, PLA University of Science & Technology, Nanjing 210007, China;

2. Chinese People's Armed Police Command College of Beijing, Beijing 100012, China)

Abstract: The author designed a centralized management system of STP. The paper explains the architecture of this system and the function of every module and the communication mode in detail. The usage of CORBA and IDL and TL1 makes the system flexible and maintainable and expansible. The implement of this system makes it possible to centralized manage and maintain STP, and may be a base of the monitoring of signaling network and the gathering and analysing of historical data.

Key words: CORBA; TL1; centralized management system of STP

七号信令系统是一种国际性标准化的通用公共信道信令系统。作为电信网的三大支撑之一, 电信网的重要组成部分, 其应用十分广泛。目前, 我国已经建立了由高级信令转接点(HSTP)、低级信令转接点(LSTP)和信令点(SP)的三级结构七号信令网^[1]。为了方便对信令网中的信令转接点设备(STP)进行集中管理和统一维护, 有必要建立一个全网的 STP 管理系统, 使得管理员远程对处于不同地点的 STP 进行管理和检测成为可能。本文介绍了如何利用构建技术 CORBA、文本语言 TL1 以及高级编程语言实现 STP 网元集中管理系统。

1 CORBA 技术简介

通用对象请求代理体系结构 CORBA (Common ORB Architecture) 技术是当今主流的三大构件技术之一。所谓构件是指软件系统的基本单元, 由一个或多个在逻辑上紧密关联的对象封装而成, 也是一种开发人员不必改变软件源代码就可以跨平台运行, 并在各种软件环境下实

现重用的、具有某种功能的独立软件^[2]。构件技术能很好地解决分布式计算环境中异构数据源间的互联共享和互操作问题, 实现各种应用程序间的协同工作。此外, 利用构件的重构性还可以大大提高软件系统的开发效率, 增强系统的稳定性和开放性, 大大提高系统的可扩展性和可维护性。现在它已经成为分布软件开发的主流, 并被业界广泛接受。现有操作系统和硬件平台的任何一种组合几乎都支持 CORBA 技术, 同时 CORBA 支持大量编程语言。

ORB 是 CORBA 的基础, 是在分布环境下, CORBA 所使用的、基于对象模型的软件总线, 其基本职责是解决对象引用的请求和建立应用对象之间的联结, 通过标准接口, 使这种联结独立于所使用的硬件和软件平台, 从而保证了对平台的透明性以及操作系统、网络协议和编程语言的透明性。对象服务是为公共设施和各种应用对象提供的基本服务, 它包括命名服务、事件服务、事

技术与方法 Technique and Method

件处理服务、通知服务、交易服务、生命周期服务和安全服务等等,其体系结构如图 1 所示。CORBA 提供了开发可移植的分布式应用程序的抽象和服务,而不必顾及它们的底层细节,并对多请求响应模型的支持,透明的对象定位和调动、以及编程语言和操作系统的无关性为传统的系统集成和新的应用程序的开发提供了坚实的基础。

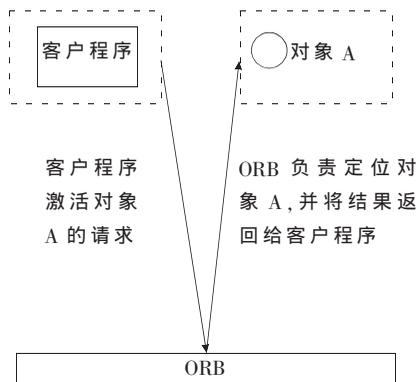


图 1 CORBA 对象间的交互

图 1 说明了客户应用和服务对象是如何交互的。客户应用不必知道服务对象位于本机上或者远程的机器上。客户应用只需要知道服务对象的对象引用(或者机器名和端口号)及如何来使用服务对象的公共接口即可。由 ORB 来处理发送请求、对象定位和返回结果等具体问题。

需要注意的是,ORB 并不是一个单独的进程。它是库函数和网络资源的集合,并且要与客户程序集成在一起,起到定位和方便使用分布对象的作用。

2 TL1 协议简介

TL1 由 Telcordia(先前的 Bellcore)定义,全称为 Transaction Language-1。它是一个基于字符的管理协议,因此,消息阅读很容易,不需要复杂的调试和协议分析器。

TL1 作为网络设备和网络管理程序间的基本接口,用于操作、管理和维护电信网络。TL1 代理位于被管设备上提供访问被管设备数据的接口,TL1 管理者使用这个接口监视和控制被管设备。数据(TL1 messages)在两者之间以二进制流的方式传送。在设备侧和管理程序侧的 TL1 实现允许用户进行以下操作:告警和网络监视、性能监控、网络安全和管理及检索网络资产数据。

TL1 定义了四种消息类型:(1)输入消息(Input Message)。输入消息是一条从 OS 或其他来源(如管理者)到网元(如代理)的命令,这条消息要求 NE 进行一些动作;(2)响应消息(Response Message)。响应消息是对应一条输入命令的详细回复(可以是一个回复或多个回复),它包含了命令是否执行成功的信息以及所需要返回的数据;(3)应答消息(Acknowledgment Message)。应答消息是一个从 NE 的简短回复,标志输入命令消息正在处理或已经被拒绝执行,主要是通知用户 NE 对所收到的命令需要较长时间的处理;(4)自发消息(Automomous Message)。当 NE 进行周期性的操作或报告异常事件发生时产生自发

消息。图 2 说明了 TL1 与 STP 的通信过程。

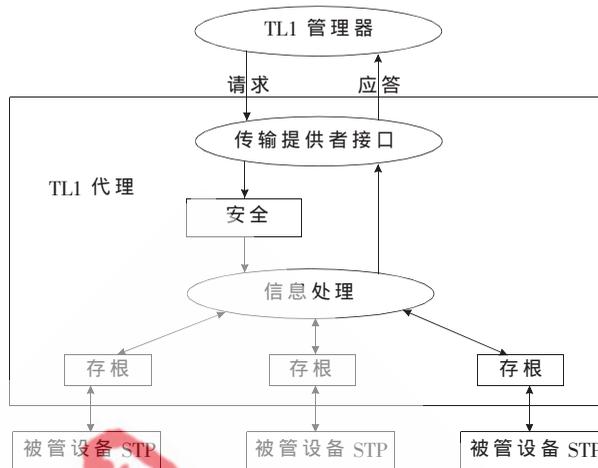


图 2 TL1 的工作流程

传输提供者接口(Transport Provider Interface)作为底层传输协议和会话层之间的一个桥梁,传输提供者接口对应于管理者和 TL1 代理之间的通信方式。使用特定的协议以满足通信需求,并可以让会话层不必理会底层所用的传输协议。

消息处理(Message Processing)用于处理从客户端收到的输入消息,它解析和处理 TL1 引擎收到的输入消息,并且发送输出的响应消息和自发消息到网元。

管理者只需要建立一个连接到被管理者,之后可以通过连接发送字符消息对被管理者进行操作。管理者只需要对接收到的字符消息进行解析就可以了。

3 CORBA 在 STP 管理中的实现

借助于 CORBA 创建某个应用 X,该应用包括两个部分,即客户方程序和服务方程序。首先,定义一个接口,接口中定义了服务端提供的操作。服务器程序的功能是实现应用中定义的接口,为接口提供一个实现对象,该对象也称为服务对象(Servant)为客户方提供服务。当服务对象被请求时,该对象可以实现被请求的功能。通过一个 CORBA 对象来提供这一功能,并用 IDL 接口来描述访问语法。X_impl 类是提供这一功能的 C++ 语言级对象类,通过创建一个 X_impl 对象实例并将其注册到对象适配器上或发布一个对象引用,便使 C++ 语言级对象变成了可被远程访问的分布对象。客户程序通过对象适配器或服务对象发布的对象引用就可以访问该对象,该访问过程就好像服务对象在本地一样。

(1)接口。使用 IDL 语言来描述服务对象的公共接口。IDL 语言描述服务方对外提供,客户方可以访问的共有操作。

(2)对象适配器。对象适配器(Object Adapter)是 ORB 的一个组成部分,它提供产生和解释对象引用、激活对象实现和访问对象状态相关的各种服务。

(3)客户方和服务方的框架代码。客户方框架代码

技术与方法 Technique and Method

(stub)负责将对服务对象的本地接口请求转换为网络请求。服务方的框架代码(skeleton)帮助对象适配器将相应的接口传到服务对象(Servant)的具体实现。

(4)服务方程序。服务方程序(Server)是一个拥有一个或多个 CORBA 对象的应用程序,用来实现接口中所声明对象。服务方程序负责创建提供接口服务的实例 X_impl 类,将该命令注册到对象适配器中并创建对象引用。对象引用包括了对象 ID 和它在网络中的位置信息。

(5)客户方程序。客户方程序通过对象引用中的信息来定位将请求发送给对象,获得相应的服务^[3]。

4 STP 网元集中管理系统的实现

4.1 网元管理系统(EMS)对 STP 的管理

图 3 显示了 EMS 与 STP 的通信方式。EMS 使用 TL1 通过 TCP/IP 协议与 STP 进行通信。EMS 对 STP 的管理分为主动方式、被动方式和数据库方式。(1)主动方式。EMS 可以向 STP 发送指定的 TL1 命令,STP 接收到 TL1 命令后,解析该命令并向 EMS 发送相应的返回字符串。EMS 通过解析该返回字符串就可获得与所发送命令相关 STP 的属性和状态;(2)被动方式。EMS 可以通过设定时间间隔,使得 STP 定期向 EMS 报告其所处的状态。EMS 还可以设置 STP 需要上报的告警和事件类型,使得 STP 可以主动向 EMS 上报相应的告警和过滤;(3)数据库方式。EMS 设定 STP 更新数据库的时间间隔,可以通过访问该数据库中的内容获得 STP 的相关属性。但这种方式只能查看 STP 的属性,不能对 STP 进行相关的管理。

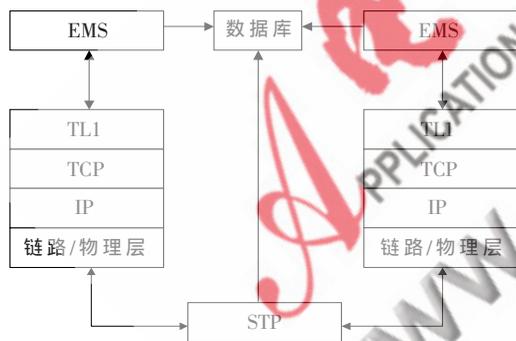


图 3 EMS 与 STP 的通信方式

4.2 STP 网元集中管理系统的实现

图 4 显示了 STP 网元集中管理系统的层次结构。该系统总共分四层。第一层是 STP 网元集中管理系统;第二层是适配器;第三层是 EMS;第四层是信令转接点设备。网元管理系统(EMS)北向接口为 EMS 与 STP 网元集中管理系统传递信息的接口,STP 网元集中管理系统通过该接口可以实现对 EMS 下的 STP 的管理。网元集中管理系统通过适配器与若干个 EMS 相连,通过 EMS 管理该 EMS 下的若干个 STP。STP 网元集中管理系统对 STP 的管理分为与网元管理系统对 STP 的方式类似。只是 STP 网元集中管理系统首先需要通过适配器将相应的 TL1 命令发送到 STP 所属的 EMS,再由 EMS 对指定

的 STP 进行管理。此外,适配器需要根据 STP 返回的 TL1 字符串更新相关的数据库。STP 网元集中管理系统也可以通过查询该数据库获得相应的 STP 属性。该数据库与 EMS 所需更新的数据库是两个相互独立的数据库。该数据库包含了整个信令网中的所有 STP 信息。

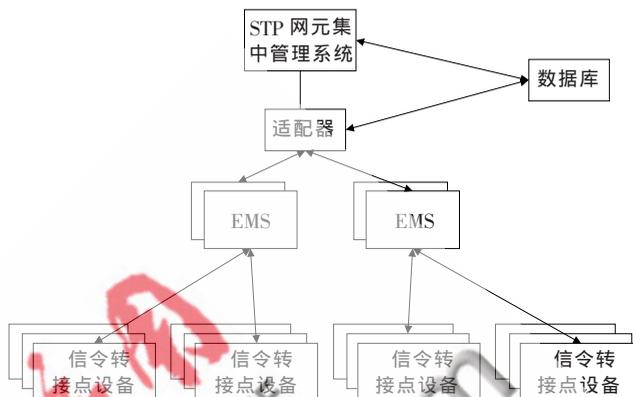


图 4 STP 网元集中管理系统的层次结构

采用 CORBA 和 TL1 技术来实现 STP 网元集中管理系统具有明显的优势,使 STP 网元集中管理系统具有以下特点:(1)灵活性。由于 CORBA 具有跨平台性、跨操作系统和跨语言的特性,因此在开发平台和运行平台以及编程语言的选取上有极大的灵活性。(2)可维护性。由于接口的屏蔽作用,使得服务方可以在实现服务的情况下随意改动服务的实现方式而不影响客户端的实现。(3)可扩展性。由于采用了接口语言 IDL,只需要通过修改 IDL 代码,并在服务方程序中添加相应的功能代码,服务方就可以实现其功能范围的扩展而不影响现存服务方程序和客户方程序。

STP 网元集中管理系统的开发使得 STP 管理员可以实现远程对信令网的管理和维护,减轻了信令网管理和维护的代价以及管理员的负担。而且,为进一步对信令网的检测和历史数据的采集、分析等提供了基础。

参考文献

- [1] 桂海源,骆亚国.No.7 信令系统[M].北京:北京邮电大学出版社,2009.
- [2] 乐江源,赖小华.基于 CORBA 技术的电工电子虚拟实验室的开发[J].井冈山学院学报,2007,28(10):19-21.
- [3] 朱其亮,郑斌.CORBA 原理及应用[M].北京:北京邮电大学出版社,2001.

(收稿日期:2010-04-12)

作者简介:

张雪,女,1986 年生,硕士,主要研究方向:软件工程。
 韩福丽,女,1962 年生,教授,主要研究方向:网络管理。
 倪桂强,男,1966 年生,教授,主要研究方向:网络管理、软件工程。