

基于 CORBA 的网络故障管理系统的设计与实现

陈炯¹ 曹争

(东南大学计算机科学与工程系, 南京, 210096)

摘要: 随着 web 技术和分布式计算技术的发展, 基于 web 的网络管理和分布式网络管理成为网络管理的发展趋势。本文详细介绍了一种分布式的网络故障管理系统的设计模型, 并就何开发基于 CORBA 技术的网络故障管理系统-----FMSBC(Fault Management System Based on CORBA)系统作了有益的尝试。最后, 本文对系统的进一步改进作了说明。

关键词: CORBA 故障管理 JAVA

中图分类号: TP393

The Design and Implement of CORBA-Based Network Fault Management System

Chen Ji ong Cao Zheng

(Southeast University, Computer Science Dept., 210096 Nanjing, P.R.China)

【Abstract】 With the development of technologies of web and distributed computing, network management in web and distributed network management become the trend. A model of distributing network management system is given in the paper. The paper also introduces the implement of FMSBC (Fault Management System Based on CORBA) which is a distributed network fault management system and some advice to develop the system.

【Key words】 CORBA Fault Management Java

¹作者简介: 陈炯, 硕士研究生, 主要研究方向为网络管理; 曹争, 工学硕士, 东南大学计算机系副教授, 主要研究方向包括网络管理、网络体系结构、开放分布式处理等。

综合性的网络故障管理在网络管理中占有很重要的地位。传统的网络故障管理大部分是集中式管理。随着网络应用的普及，网络规模的扩大，网络故障管理所处理的信息来源已经不再局限于传统的网络互联设备。对众多网络设备的故障信息进行集中处理不仅效率很低，而且会加大主机的负载。伴随着分布式技术的不断应用，特别是CORBA技术的广泛应用，网络故障管理都趋向于分布式管理或者是分布式与集中式相结合的管理方式。FMSBC（FAULT MANAGEMENT SYSTEM BASED ON CORBA）是由CERNET华东（北）网络中心开发的基于CORBA的网络故障管理系统。FMSBC系统实现了对网络故障信息进行分布式管理，包括故障信息的收集、分类、过滤、入库、转发等功能。FMSBC系统提供WEB浏览和命令行两种运行方式。

1. FMSBC 系统结构和功能设计

FMSBC系统定义了一种通用的网络故障管理的层次化结构，如图1所示：



图1. 网络故障管理层次化结构

系统结构共分为三层。trap收集层是属于网络管理平台的一部分；信息处理层是整个故障管理的核心层，与信息报告层同属于网络管理应用程序。三层的主要功能定义如下：

trap收集层 是整个系统最底层，属于网络管理平台的一部分，主要是负责收集由被管理设备发出的trap信息，保存原始的故障信息。整个trap收集层包含一个demo进程，该进程不断地监听网络上的各种trap，一旦接收到trap，就将其存入本地的一个trap列表中。

信息处理层 是故障管理系统的核心部分，它的功能包括对trap信息进行分类、过滤等。按照rfc1215标准所定义的trap类型，一般的trap类型段表示了比较精确的类型，它的取值有七种：

- 0: 冷启动（coldstart），表示协议实体重新初始化，代理配置或协议实体的实现可能被修改。
- 1: 热启动（warmstart），发送协议的实体重新初始化，但是没有被修改。
- 2: 链接关闭（linkdown），代理中的通信链接不可提供使用。
- 3: 链接开启（linkup），代理中的通信链接开始工作。
- 4: 身份验证失败（Authentication Failure），发送协议实体收到了身份验证失败的信息。
- 5: 邻居丢失（egp Neighborloss），EGP端断连。
- 6: 企业特有（enterprise specific），发生了企业特有的事件。特定的trap类型在FMSBC中另外有定义。

信息处理层按照设定的规则对所获取的trap进行过滤处理（并非所有的trap信息都是故障信息），然后按照上述的trap类型将trap进行分类并将trap信息封装成为相应的卡片。FMSBC系统采用了故障卡片的管理方式，用卡片的形式传递所有的故障信息，故障卡片隶属于某一个故障事件。当一个事件对象的一个事件源（如一台网络设备或是一条线路）发生故障时，这个事件源一般会向故障管理系统发送多条trap信息，相应的会产生多张卡片。将由同一个事件源发送的trap产生的卡片归并入同

一个故障事件，有助网络故障的分析和解决。故障事件分为已决事件和待决事件两种。凡是事件状态不是关闭的事件都是待决事件，统一存放在待决事件表（`unsolve-table`）中。故障事件包括以下几个方面的内容：

事件对象，事件源：唯一确定被管对象（如一条线路或网络设备）；

事件ID：在事件库中唯一地标识一个事件；

事件开始时间：事件源发生故障的时间；

事件结束时间：事件源恢复正常的时间；

事件卡片数：由trap信息产生的相应的卡片的数目；

事件状态：在FMSBC系统中，定义的事件状态由高到低分为：紧急，警报，错误，警告，注意，关闭。事件状态的定义主要体现了网络故障等待处理的优先级。

事件类型：在FMSBC系统中，定义的事件类型分为：线路，安全，环境，路由，系统，配置六种。

卡片的内容包括以下几个方面：

事件ID：标识卡片所属的事件在事件库中的标号；

卡片类型：将事件的六种类型细分出来的若干子类型；

卡片级别：对应事件状态的六种处理优先级；

卡片时戳：卡片产生的时间；

卡片ID：卡片在所属的故障事件中的标号；

卡片创建者：标识创建卡片的管理员或事件源；

卡片诊断信息：比较详细的故障原因；

卡片处理方法：保存解决故障的方法，以便于今后类似的故障的解决。

从上面的相关定义，我们可以看出故障事件完全依赖于卡片，故障事件的许多成员参数会随着新卡片的产生而发生相应的改变。因此卡片式故障管理的核心内容是如何将已获取的trap信息封装成卡片信息。由于各种不同设备发送的trap的格式不相同，结合CERNET华东（北）网络中心网络设备和线路的实际情况，FMSBC系统分别处理四种格式的trap，包括STD(.1.3.6.1.6.3.1.1.5)，GEN(.1.3.6.1.2.1)，CISCO1(.1.3.6.1.4.1.9.0)，CISCO2(.1.3.6.1.4.1.9.9)。信息处理层从trap的第一条variable的前几个字段获取trap的格式类型，然后分别按照相应的方法取出后续的几个字段或者是几条variable，产生相应的卡片。当从trap信息中获取的事件源同待决事件表中的每个待决事件的事件源都不相匹配的时候，说明一个新的事件源发生故障，此时就要产生一个新的故障事件，加入待决事件表中。

信息报告层的主要功能是报告故障信息。FMSBC系统提供以下几种报告形式：

- 1 线路故障数据统计报告
- 2 当前网络故障监测报告
- 3 预测报告

线路数据统计报告反映了网络线路故障的历史情况，包括故障率、可用率、解决方法等等，为管理员提供经验资料。当前网络故障监测报告反映当前已产生故障的事件源的相关信息，包括故障开始时间、故障事件类型、故障事件级别等。预测报告给出将来某时刻网络可能出现的问题，是管理员对网络进行重新规划，调整的依据。通过卡片信息对线路的历史故障情况进行统计是很有必要的。它不仅能基本反映一条线路的网络质量，而且也为以后的预测提供依据。

2. FMSBC 系统的实现

2.1 FMSBC 系统的技术特点

FMSBC系统采用了servlet和CORBA分布式结构相结合的技术。与传统的管理技术相比较，FMSBC系统有如下几个特点：

- 1 用户界面友好，易于使用，用户仅需要掌握基本的浏览器使用方法。
- 2 Servlet动态的生成html文档，与浏览器之间使用http协议进行传输，同java applet相比较运行速度大大提高。
- 3 基于CORBA技术便于分布式管理，可以在多个服务器之间进行负载均衡。
- 4 运用CORBA的Event Channel实现采集服务器主动的向终端发送卡片信息。

2.2 FMSBC 系统的运行模式

FMSBC 系统是基于 CORBA 规范的分布式故障管理系统，所以 FMSBC 系统按照 CORBA 规范实现了请求-----代理-----服务（REQUEST-----BROKER-----RESPONSE）的运行模式，如图 2 所示。

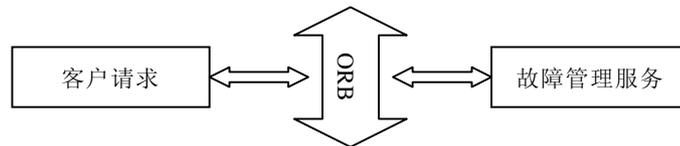


图 2. FMSBC 系统的运行模式

在 FMSBC 系统运行过程中，客户端首先将对象服务的名称以及相应的参数传递给对象请求代理---ORB(Object Request Broker)，然后由 ORB 将参数提交给相应的 CORBA 对象服务。服务完成后，将按照提交的逆过程将服务结果返还给客户端。FMSBC 系统采用这种典型的 CORBA 体系结构有以下几个优点：

- 1 实现客户端与服务器的分离，双方可以按照接口定义自由升级。
- 1 提供软件总线的机制，只要软件和程序的接口符合规范就可以方便的集成到 CORBA 系统中，实现软件总线上的即插即用。
- 1 实现分层设计的原则和实现方式，CORBA 的具体对象服务是逐步分层定义实现的，系统的复杂功能由内核向外延伸的。

2.3 FMSBC 系统的系统结构

整个 FMSBC 系统的结构模块如图 3 所示分为四个主要的部分，包括命令行运行方式模块（图中 A），web 运行方式模块（图中 B），CORBA 对象服务模块（图中 C），trapd 和 ticket server 模块（图中 D）。其中模块 A 和模块 B 是客户端的应用程序，模块 C 和模块 D 是服务器端的程序。各个模块介绍如下。

2.3.1 命令行运行方式模块

在命令行运行方式下，FMSBC 系统现在提供创建卡片（new ticket）和接收卡片（ticket receiver）这两个最基本功能的命令行运行方式。在命令行运行方式中，先要对输入的各项参数进行有效性检查；检查确认后，通过 ORB 向对象服务提交请求。服务器完成服务请求后，通过 ORB 将处理后的结果返还给客户。在命令行方式下启动 ticket receiver 是采用事件服务的方式，应用程序必须与事件通道（event channel）建立联接。在无消息到达的时候，ticket receiver 将一直处于休眠；当有消息到达的时候，ticket receiver 将被 event channel 唤醒并开始接收数据。

2.3.2 WEB 运行方式模块

web 运行方式采用的是三层结构模式。其中用于用户使用的浏览器属于 client 端。驻在 web server 上的 servlet 是中间层。用户通过浏览器获取相关的数据，并通过浏览器提交请求服务的命令。用

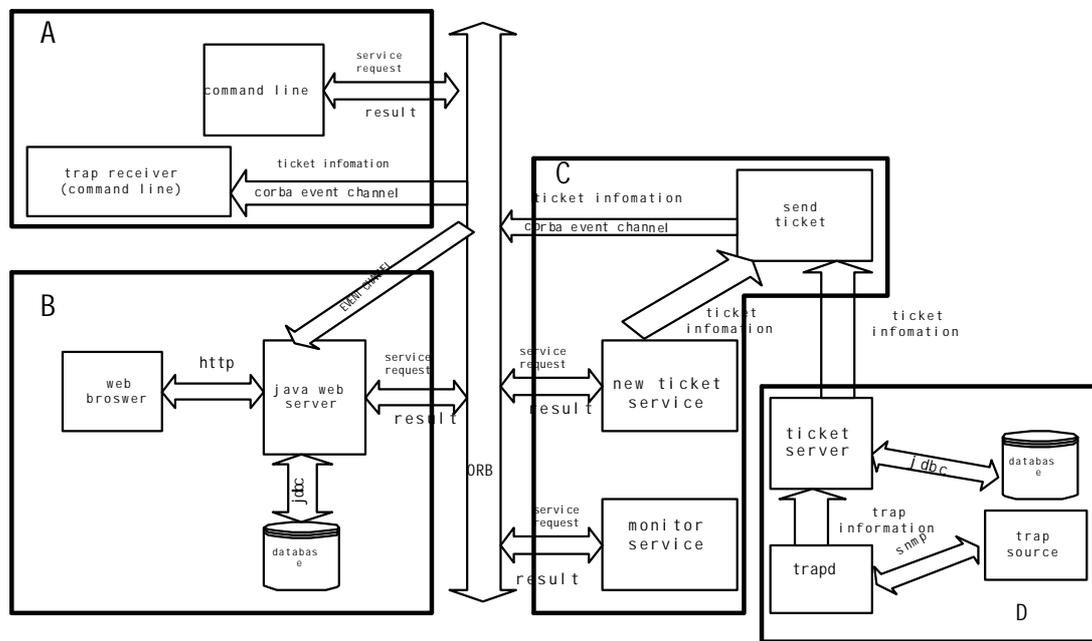


图 3. FMSBC 的系统结构

户的请求命令由 servlet 来接收和进行必要的处理，然后 servlet 再按照 CORBA 对象服务的接口将参数提交给 CORBA 服务器端，进行最终的处理。处理完的数据结果返回给 servlet 的时候，servlet 将数据封装在 html 文档中提供给用户。为了在 servlet 中能够调用 CORBA 的服务，必须要求在 servlet 中实现 CORBA ORB 和 BOA 的初始化。

2.3.3 CORBA 提供服务模块

整个系统对外提供的服务都由这部分统一来完成。其中 new ticket 和 user monitor 服务提供的方式是 CORBA 的 name service (名录服务)，而 send receiver 服务是 CORBA 的 event service (事件服务)。

使用名录服务的时候，CORBA 对象可以通过“名称”进行注册和查找定位。客户端通过名录对象服务的名称来调用对象服务。客户端按照接口定义通过 IDL 桩将接口参数进行映射，映射后的参数传递给 ORB，再由 ORB 将参数传递给对象服务，对象服务的 IDL 桩在将参数映射成为可对象服务可识别的参数。最后按照上述过程的逆向，从对象服务端将数据结果传递给客户。

Send ticket 的实现是使用得事件服务。FMSBC 系统使用事件服务是为了实现 CORBA 的对象服务端主动将数据发送给客户端。在系统实际运行的时候，系统中建立一个事件通道 (Event Channel)。对象服务端和客户端分别绑定到事件通道上。当对象服务端有数据产生时，对象服务端将数据主动的发送给事件通道，事件通道又将主动去唤醒客户端去接收数据。通过数据通道这种方式，客户端就可以实时的接收到新产生的卡片。

2.3.4 trapd 与 ticket server 模块

Trapd 作为 FMSBC 系统的 trap 采集层，主要的功能是接收从网络上的设备和线路发送过来的 trap。Trapd 模块包含一个 deamo 进程。当 deamo 进程接收到网络设备发送过来的 trap 后，会对 trap

进行转换，然后将转换后的 trap 传递 ticket server。

Ticket server 作为 FMSBC 系统的信息处理层，是整个系统的核心部分。FMSBC 系统的管理方式是卡片式管理，系统中消息的传递都是以卡片的形式完成的。卡片封装是 ticket server 的一项主要工作。Ticket server 要对 trap pdu 中的相关数据项分析，获取卡片的各个数据项并封装成故障卡片。各个网络设备厂商将网络故障信息用多 trap 来描述示的方法不统一，ticket server 要对这些 trap 分别进行处理。

2.4 技术难点

系统实现中主要的技术难点是 servlet 中 CORBA client 的嵌入。在基于 web 的运行方式中，servlet 作为 CORBA client 端要向提供服务的 CORBA 对象服务提交服务请求。因此，在 servlet 中必须进行 CORBA 的 BOA 和 ORB 的初始化。由于 servlet 是近几年来才出现的比较新的技术，所以 servlet 与 CORBA 的互通方面的研究目前做的很少。要实现在 servlet 中的 CORBA client 的嵌入，首先就是要 web server 所带的 servlet 引擎要能够支持 CORBA；其次就是在 servlet 引擎中有关 CORBA 类库路径的设置。不同公司的 CORBA 平台产品所使用的类库是各不相同的，在 servlet 引擎中不可能包含所有的这些类库文件。Servlet 要与 CORBA 互通，必须相应的 CORBA 类库包外挂到 servlet 引擎中，指定相应的路径。

3. FMSBC 系统的改进方案

FMSBC 系统今后还可以在以下几个方面进行扩充：

首先是 servlet push 的实现，即让 servlet 再接收到新的卡片信息后主动向浏览器发送信息。这样可以在 web 运行方式下，真正做到实时数据更新。

其次是增加故障卡片之间的关联。在发生故障的时候，系统自动分析一些相关的故障卡片，提交管理员一些可能的故障原因，使整个系统的管理更加的智能化。

进一步的改进，可以将管理员对故障解决的方法作一个分类总结，当以后发生类似的故障的时候，系统可以主动向管理员提交一些参考的解决方案。

4. 结束语

随着网络规模的扩大，网络结构的复杂，网络故障管理在网络管理中变得越来越重要。基于 WEB 的故障管理和分布式故障管理成为网络故障管理的主流，FMSBC 系统在以上两个方面作了很好的尝试。FMSBC 系统有很好的可扩展性，可移植性。目前，FMSBC 系统已集成到 CERNET 华东（北）的 MUDMAN 网络管理系统中，并已经正常运行。

参考文献

- 1 Programmer's Guide--VisiBroker for Java Inprise Corporation, 100 Enterprise Way Scotts Valley, CA 95066-3249
- 2 Gatekeeper's Guide-- VisiBroker for Java Inprise Corporation, 100 Enterprise Way Scotts Valley, CA 95066-3249
- 3 AdventNetSnmpV3 API Reference <http://www.adventnet.com>
- 4 汪云 编著.CORBA技术及其应用.南京:东南大学出版社,1999
- 5 夏亮, 韩松洋, 赵毅 Agent应用程序的开发 《计算机应用》增刊 成都: 科学出版社 1999
- 6 岑贤道, 安常青 网络管理协议及应用开发 北京: 清华大学出版社, 1998
- 7 Stephen R Schach .Software Engineering with Java.McGraw-Hill, 1999