

面向对象技术在网络管理中的应用研究

On the Using of Object-Oriented Technology in the Field of Network Management

杨锐 白英彩

(上海交通大学金桥网络工程中心 上海200030)

摘要 The model and description of network resources is one of the key problems that NM must solve. In the promoted solutions, the object oriented method gained many attention. In this paper, we analysis the using of object oriented technology in the field of NM before, and point out its drawback. At last, we gave our solution for it.

关键词 Object-oriented, Network management, MIB

网络资源的建模与描述是网络管理所必须解决的关键问题之一,如果不能够较好地描述网络中的资源,网络管理系统又如何对其进行管理?因此,许多研究机构和学者对此进行了深入的研究。网络管理的信息模型就是针对这个问题进行研究的。在提出的模型和描述方法中,应用面向对象的方式对网络资源进行定义是一个比较引人注目的方向。

以往面向对象技术主要是应用于软件领域^[1,2],但从对面向对象技术的分析可以看出,面向对象技术关键之处在于它用一种崭新的方式对系统进行描述。我们认为,面向对象技术应也很适合于其它非软件领域系统的描述和建模。在这方面一个成功的例子是美国国防部 RASSP(Rapid Prototyping of Application-Specific Signal Processors)项目的 OO-VHDL(Object Oriented VHSIC Hardware Description Language)语言。VHDL 是一种描述数字式硬件的语言,在1987年被 IEEE 确立为标准。将面向对象技术引入 VHDL 后,显著地缩短了抽象模型的建立时间,对降低系统建模的复杂性有很好的效果^[3]。我们同样相信面向对象技术在网络管理领域的应用也会取得良好的效果。

一、面向对象技术在网管中的早期应用

目前两个主要的网络管理解决方案是 ISO 的一系列网络管理标准和 IETF 的 SNMP^[4]和 SNMPv2^[11]。ISO 网络管理的管理对象是网络资源,指网

络系统中的设备(如路由器、主机)、服务(如连接, QoS)等。在描述这些网络资源时,ISO 使用了被管对象的概念。被管对象是从管理的角度对网络资源的抽象,一个被管对象反映了网络资源的某个特性,而不是这个网络资源本身。ISO 的面向对象的思想主要也是体现在封装、抽象和继承三个方面。

· 封装

被管对象有一些属性,一个属性有其相关的值,一个属性的值在这个被管对象的边界可被外部看到,往往反映出被管对象的行为。被管对象还定义了操作和通知(notification),和被管对象的属性同时被封装在被管对象中,只在被管对象的边界才被外部管理所看到,而其内部功能和结构对外部是透明的,除非被管对象所定义的操作允许访问某些内部信息。外部管理可通过向被管对象发送一个请求以获得属性的值,这同样需调用被管对象提供的操作。ISO 要求对属性的操作必须通过对此属性的被管对象的操作来实现,而不应直接操作于属性本身。被管对象能够对某些属性的值进行限制以保证内部数据的完整和一致。被管对象所定义的操作都有严格的一致性来保证这一点,这就使被管对象的完整性得到保护。

· 抽象

在 ISO 管理中,一些被管对象可被抽象为一个被管对象类。一个被管对象类被定义为一些包

杨锐 博士,主要研究方向为网络管理,高速网络技术及人工智能技术在网络管理中的应用等。白英彩 教授,博士生导师,主要研究领域为网络管理,网络安全,CATV 等。著书甚丰。

(packages)的集合。每一个对象类中都包括一组属性、操作、通知和相关的行为。一个被管对象是一个被管对象类的实例。

为了使被管对象类及其相关特性的定义更规范方便,ISO 还在文[7]中定义了一些模板(template)。

对一个被管对象类的定义包括:

-这个被管对象类在继承层次中的位置。

-一组属性、操作、通告和行为的强制包(mandatory package) 是指这个被管对象类的所有实例被管对象必须支持强制包中的所有内容。

-一组属性、操作、通告和行为的条件包(conditional package),以及相应的条件 是指当这个被管对象类的一个实例被管对象满足类中定义的条件时,必须支持条件包中的所有内容。

-在每个包的结构中包括:可在被管对象的边界被外部看到的属性、可应用在被管对象上的操作、被管对象所表现出的行为、被管对象可能发出的通告。

所谓包是一组特性的集合,如属性、通告、操作和行为等,一个被管对象的性质都必须在包中定义。包是被管对象类定义的一个重要模块。ISO 也定义了它的一些性质,如一个具体的被管对象只能支持一套包等。

从上面可以看出,对被管对象的抽象大大降低了描述网络资源的复杂性,另外 ISO 更进一步在抽象的基础上提供了描述网络资源所用的模板,更加规范化了描述,这也是面向对象技术的目标之一。

·继承

ISO 定义一个被管对象类可以从另一个被管对象类中派生出来,派生出的类是其父类的子类。子类继承父类的特性,同时也可定义自己新的特性。ISO 的定义还支持多重继承的概念,即一个子类可以从多个父类中继承属性、操作等特性。ISO 还定义了一个超类 TOP,它是最高层的父类。下图给出了一个 ISO 网络管理信息模型继承层次的一个例子。

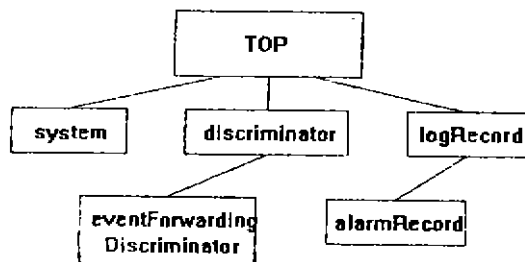


图1 ISO 网络管理信息模型继承层次例子

Internet 的网络管理标准 SNMP 及 SNMPv2 对网络资源的描述方式借用了 ISO 的方式,也采用了被管对象的概念,两者在这方面比较相似,都是用 ASN.1 形式语言^[12]的一个子集描述的,但 SNMP 并未引用 ISO 的面向对象的思想来定义被管对象。

二、传统网络资源定义方式的不足

虽然上述的网络资源定义方式采用了面向对象的思想,有不少的长处,但仍有以下一些不足。

1) 缺少对网络资源的较高层次的描述。ISO 和 SNMP 对网络资源描述方式的基本单位都是被管对象,一个被管对象并不对应网络中的一个资源,而只是描述了网络资源的某个性质,一个网络资源往往需用许多被管对象来描述。网络中被管对象的一个概念上的信息库被称作 MIB。这种方式的主要问题在于:一方面它对网络的定义过于繁琐和原始,缺乏对网络管理信息细节的封装,没有更高一层的网络管理信息的定义。MIB 的定义显得缺乏结构性、层次性。另一方面,在网络资源与网络被管对象之间没有一个清晰的关系。这些被管对象信息或 MIB 信息的含义到底反映了网络资源怎样的状态,以及如何利用这些信息实现某个特定的网络管理任务,如故障检测、性能分析等都是模糊的。这造成了网络管理应用程序的匮乏,虽然现在网络管理系统不少,但大多数网络管理系统只停留在“MIB 浏览器”阶段,并没有能够提供进一步的网络管理功能,网络管理员所得到的只是关于各种网络资源的大量繁杂的 MIB 信息。

2) 缺乏对网络资源操作的描述。目前对网络管理资源的定义也很少定义对网络资源的操作,SNMP 的方案在这方面比较明显,因此网络管理系统不知应该或能够对网络资源施加怎样的操作。没有统一的标准,这使网络管理系统很难得到进一步的发展。

三、一个改进方案

针对上面所述的状况,我们同样采用了许多面向对象的思想,提出了一个改进方法。我们所描述的对象是较高层次的,是网络中的资源,如路由器,主机,网络逻辑连接等,并将对它们的操作也封装在其描述中。这些描述都是从管理的角度进行的,其描述的具体过程可以用下面的流程简略表示。

1) 对网络资源进行分类

这是相当重要的一步,可以使我们的工作过程清晰。我们将网络中的资源分类为一棵树形结构(图2)。

我们采用类似 MIB 定义的方式,对于这棵树的每个节点都赋予它一个全局唯一的标识符,如在上图中,交换机的标识符就可以是 0.0.2。这样,在相应的网络管理系统中,用这种标识符描述所管理网络的构成,能够做到清晰简明。

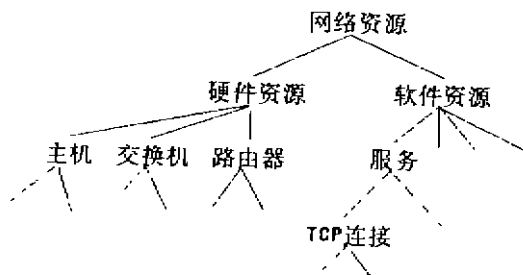


图2 网络资源树形分类

2) 抽象

对网络资源树中的节点进行抽象。对网络资源树的描述已经蕴含了各网络资源之间继承的关系。我们利用这些信息,从上至下对网络资源树中的每一个节点,从可在网络资源的边界被外部看到的属性、可应用在网络资源上的操作、网络资源所表现出的行为及网络资源可能发出的通告这几方面对其进行抽象描述,生成一个描述框架。要注意的是,这个网络资源从父类继承下来的一些性质不要再进行描述,以保持父类和子类的一致性。

3) 将各描述框架转换为对 SNMP MIB 的描述和操作

这一步也是必要的,一方面因为 SNMP 目前的流行,几乎所有网络管理系统都基于 SNMP,SNMP 已经成为 Internet 的标准之一,TCP/IP 网络的管理都要求支持 SNMP。另一方面,鉴于 TCP/IP 网络的主导和流行,将来 SNMP 定会成为网络管理的事实标准。所以我们此处也应提供对 SNMP 的支持。

经过上述几步工作,我们就可得出一个基于 SNMP 的,高层次的模块化的对网络资源的描述。

下面是我们用类 C++ 语言给出的一个简单例子。

```
class NetworkDevice{
```

```
private:
    SysInfo; //关于网络设备的系统信息,包括这个设备的
              全称,版本,软件操作系统名,设备的物理位置,
              设备的管理名称,MIB 树中此设备的位置等信息。
              对应于 MIB-II 中的 System 组。
public:
    访问网络设备系统信息的函数;
};

class Interface{
private:
    InterfaceSysInfo; //关于接口的系统信息,如接口在系
                      统中的编号,接口类型等。
    StatisticalInfo; //关于接口的统计信息,如接口进入当
                      前状态后经历的时间,发出的广播分组数,收到的错
                      误分组数等。
    OperationInfo; //接口的运行状态信息。
    QoS; //接口的 QoS 信息及控制函数。
public:
    接口访问及控制函数;
}; //以上参数可由 MIB-II 的 Interface 组来定义。

class Router:public NetworkDevice{ //路由器从网络设备
    继承
private:
    SysInfo; //关于路由器的系统信息。
    RouterInterface[InterfaceNum]; //一个接口数组描述
                                    路由器的各接口信息。
    RouterQoS; //路由器的 QoS 信息及控制函数。
public:
    路由器访问及控制函数;
};
```

结束语 我们认为,利用面向对象技术对网络资源进行描述是一个很有效的方式,但以前的应用在层次上和对操作的封装上都显得不足,在某种程度上已经阻碍了网络管理的发展,是 ISO 或 IETF 工作组应解决的问题。我们相信,采用高层次的对网络资源的描述,可以大大降低管理网络系统的复杂性,使网络管理系统结构更加清晰。这对网络管理系统的进一步发展,如推出更多的网络管理控制应用程序,会有不小的帮助。

参考文献

- [1] G. Booch, Object Oriented Design, Benjamin/Cummings Publishing, Redwood City, Calif., 1991
- [2] Hesham El-Rewini 等, Object Technology, IEEE Computer, Oct. 1995
- [3] Sowmitri Swamy 等, OO-VHDL: Object-Oriented Extensions to VHDL, 同[2]
- [4] ISO/IEC 10165-1, Aug. 1991
- [5] ISO/IEC 10165-2, Aug. 1991
- [6] ISO/IEC 10165-4, Aug. 1991
- [7] ISO/IEC 10165-5, Feb. 1992
- [8] ISO/IEC 10165-6, Feb. 1992
- [9] M. Rose 等, RFC 1155, May 1990
- [10] M. Rose, 等, RFC 1213, March 1991
- [11] J. Case 等, RFC 1442
- [12] International Standard 8824, Dec. 1987