Directory for Network Configuration Management

3D−4

K. JAYANTHI, G. MANSFIELD, Y. KIMURA *, Y. NEMOTO †, S. NOGUCHI ‡

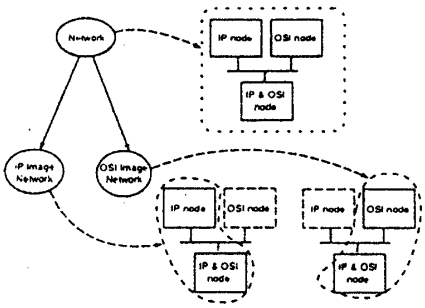
Abstract

The Network Configuration information in the X.500 Directory can be used for effective network configuration management e.g. configuration visualization, configuration analysis for reliability, performance and enhancements. The underlying concepts are outlined.

1 Introduction

Information about network configuration is important for network management in general and configuration management in particular. In recent times, networks have grown in size, making it impossible for network adminstrators to *remember* the details of the network. Further, with the globalization of the network reach the scope of the configuration information has become global. It is desirable to have access to configuration information of any part of the global network from any part of the globe. There is a movement to use the X.500 directory to provide such a framework [GYM]. In this paper, we give show how this frame work can be used for effective configuration management.

2 Directory Representation



FUNCTIONAL IMAGES OF A NETWORK

For representing networks in the Directory we use the following hierarchical model. A network is the media for transmitting information with zero or more network elements each having at least one network interface on the media. The media may be a line (physical circuit/virtual circuit), a coaxial-cable, or a collection of interconnected networks. The model allows a hierarchy of subnetworks. Network elements with multiple interfaces may act as *external gateways* to the attached network and to networks higher up in the hierarchy. Thus, a gateway may be the external gateway of several networks which are either interconnected or have a hierarchical relationship. A network may be *simple* consisting of zero or more network elements or *composite* consisting of several subnetworks. Examples of simple networks are ethernets, Optical fiber/copper cables, free space, ... Just as there are several maps of the same geographical domain (political, natural...) one can envisage several views of the same network and its components. A view (called "image" in the remainder) could pertain to a particular protocol suite (IP/OSI/...), an administrative domain or purpose. Using images, several abstractions of the same object is possible (fig. 2).

In the following we will restrict ourselves to discussion of the Physical network and its IP-image only. Also, for matters of simplicity we will keep the description of the objects simple - restricted to the attributes that relate to the discussion in the following.

3 Directory Objects

The network related directory objects and some of their attributes used to represent the configuration information in the directory are given below.

Network The network object supplies general descriptions which are common for a set of nodes and circuits comprising one network. This includes information about the type of circuits (medium, broadcast or point-to-point, etc.) and properties (speed etc.).

network
Name
ExternalGateway: list of nodes that connect this network to neighbor networks;
Type: either "composite" or bus, ring, star, mesh, point-to-point

Node

The node object describes any kind of device that is part of the network, such as simple nodes, printer, bridges.

node
Name
MachineType: e.g. main frame, work station, PC, printer;
OS: e.g. VM, UNIX, DOS; might include release

NetworkInterface

Each node object will have one or more *networkInterface* objects as subordinates. NetworkInterface objects provide information about interfaces of the node and connectivity.

*AIC Systems Lab., Sendai
†Tohoku University, Sendai
‡Nihon University, Koriyama.

networkInterface
Name: e.g. le0, COM1
Address: the protocol-independent interface address
connectedNetwork: list of networks which this networkInterface is connected to

IP network image

IP network image is one instance of network images.

networkImage
externalGateway: points to one or more nodes that act as gateway for the protocol application this image refers to

IP network interface image

The most important IP related information of a node (its IP addresses) is registered with ipNetworkInterfaceImageObjects. This picture is accurate as a node can have several IP addresses, but at most one per interface. Furthermore, it shows clearly the relationship to the neighboring IP network and the connected port.

portImage
networkInterfaceAddress: this ports address in the context the image refers to, e.g. IP number, NSAP
connectedNetwork: pointer to networkImageObject

4 Generation of network graph

The network graph is useful for network visualization. It can be used for several purposes like planning, debugging and administering. The graph is a set of vertices which may be network-vertices or node-vertices. The interconnections are represented by the edges. A Graph is represented as follows:

```
Graph: list of vertice
vertice{
        type {network/node}
        list of edges
       }
edge{
     list of two vertices
    }
```

Given a set of network elements it is possible to draw the interconnection between the elements using the information in the directory. The network elements will map to *physical communication objects* in the Directory e.g. *network object*, *node object*. The algorithm of drawing the interconnection of the objects is given below:

Algorithm *Generate interconnection graph*
input: *set of physical objects*
output: Graph

For each member of the "set of physical objects" do =
case element is NETWORK
make a network-vertice
case NETWORK-TYPE = PTP
case NETWORK-TYPE = ETHER
break;
default:
get the gateways of the network
for each gateways get the ports
for each of the ports find the
port-type =external/internal"
case external
make an edge for the network-vertice
case element is NODE
make a node-vertice
for each ports of the NODE make an edge for the node-vertice

make edge:
case edge already-exists
update the NULL vertice-ptr
case edge non-existant
create an edge
update the vertice-ptr
return edge

5 Generation of network tree

The network tree is another view of the network - it is useful for several cases e.g. in intelligent polling.

We start with the set of items as before, and the root from which the tree is to be made.

1. Locate the root element in the set.
2. Locate the interfaces of this element.
3. Locate the connected items to these interfaces.
4. If the connected item does not have any other interface, this is a leaf element; else, locate the connected item and repeat the procedure from 2.

6 Applications

The application of the network configuration are manifold. The graph generated in section 4. can be directly utilised for visualization. Further by obtaining the networkInterface address from the corresponding image object the dynamic status of the network element may be displayed.

The network tree generated in section 5. can be used to devise algorithms for intelligent polling. For example, when a gateway is down - there is no sense in attempting to find the status of network elements that lie on *the other side* of the gateway (as seen by the polling station)

Another very importatnt application is for network probing in case fo faults. Given the network graph probe-algorithms may be devised to locate the fault.

References

[GYM] G.Mansfield et.al. "Mapping Communication Networks in the Directory", JENC'93 Norway.

[HK 91] Hardcastle-Kille, S.:
X.500 and Domains;
RFC 1279, November 1991.

[IP] Johannsen, Th., G. Mansfield:
Representing IP information in the X.500 Directory;
Technical Report 92.8, AIC Systems Lab., Sendai, Japan, February 1993.

[SPP] Johannsen, Th., G. Mansfield:
The Soft Pages Project;
Technical Report 92.9, AIC Systems Lab., Sendai, Japan, February 1993.