Policy-Based Networking Architecture for QoS Interworking in IP management

- Scalable Architecture for Large-Scale Enterprise-Public Interoperation -

David C Blight, Takeo Hamada
Fujitsu Laboratories of America
595 Lawrence Expressway, Sunnyvale, CA 94086-3922
e-mail: dblight|thamada@fla.fujitsu.com
Ph:+1 408 530 4575 Fax:+1 408 530 4515

Keywords

Policy-Based Networking, Policy-Driven Management, Quality of Service Management, Management of Internet, IP management

Abstract

Policy-Based Networking (PBN) is gaining a wide acceptance in IP management, resulting in a more unified control and management approach toward complexity of IP management. QoS interworking in IP management based on PBN is going to provide QoS guaranteed/differentiated IP connection services. QoS interworking issues between enterprise network and public IP network are studied. The exponential growth of intranet/internet interworking itself may put PBN in jeopardy. To counter the increasing management complexity, two approaches, policy abstraction and hierarchy organization with precedence rules are proposed and studied.

1 Introduction

A new demand being placed on the internet is to provide guaranteed Quality of Service (QoS). Although we don't normally associate QoS requirements with Internet Protocol (IP) based applications as IP based applications traditionally have used a best effort approach to QoS, in an Enterprise environment QoS may be applied to services on a network in order to ensure effective return on IT investment. In addition, multimedia applications such as: Internet telephony, Video on Demand (VoD), video conferencing, groupware, distance education, and remote health care are examples of applications which may have stringent Quality of Service (QoS) requirements. QoS requirements of applications and services will lead drive the policies used to managed IP based networks, and specify Service Level Agreements (SLAs) with Internet Service Providers (ISPs)

The current internet architecture does not support QoS guarantees. The internet is traditionally based on a best effort routing principle, in which each packet of information is treated independently and fairly. Routers, which are layer 3 devices in the ISO seven layer network protocol model, are only concerned with routing packet towards their destination, based on information in the IP header. IP is a connectionless service, and no state information is maintained at the routers about connections. Most connection oriented services are implemented using higher level protocols such as TCP/IP. Here the connection state is maintained at the end hosts.

As the internet is increasingly becoming QoS oriented, not only does it have to support generic applications and application-oriented protocols, but the burden of management and maintenance of resources and resource management protocols for QoS support has to increase at the proportional rate. The overall complexity of IP management become significant. The rising popularity of policy driven management (PDM) [1], or policy-based networking (PBN) is a natural answer to it. Though the PDM was originally proposed for security management, in particular for access control, the concept is flexible and generically applicable to broader aspects of network/ system management problems.

In this paper, we study PBN architecture for QoS interworking in IP management. Although we are becoming confident that it is possible, and it will receive wider acceptance both from enterprise and public networking community, it seems that much of the architectural issues still remain unsolved, in particular scalability issues of PBN still await further researches. Our primary goals in this paper is to investigate the QoS interworking in IP management, in the light of scalability analysis of large-scale interworking of PBNs. This paper is organized as follows. Following this introduction, we explain QoS interworking in IP management in Section 2. In Section 3, physical architecture of PBN is explained. In Section 4, we focus on the issues of public-enterprise PBN interoperation. We turn our attention to scalability issues of PBN in Section 5. Following the results of analysis in the preceding sections, we present a scalable PBN architecture using policy abstraction mechanism in Section 6. In Section 7, we discuss policy divergence due to organizational hierarchy and how we can deal with them by using precedence rules. Conclusions follow in Section 8.

2 QoS Interworking in IP Management

The Quality of Service requirements may be specified for an application, service, or network. QoS specifications for applications (instance of service) and its associated traffic stream are often expressed in terms of bandwidth, delay, delay variation (jitter), error, and reliability[8]. Many of these applications can not tolerate performance below specified levels. For example, delay greater than 400 ms is unacceptable to speech applications as it introduces delay distracting to humans. High jitter or error rates may make audio unintelligible. QoS for services will include constraints for multiple applications and traffic streams, and may include higher level requirements such as availability and reliability. Network centric QoS focuses on satisfying needs of interoperating services and technologies and will focus on more business oriented constraints including cost, security, reliability, and manageability.

Currently there are at least three approaches being taken to meet QoS issues in IP networks. The first approach is often referred to as class based, and is being standardized by the IETF Integrated Services[1] (intserv) working group, involves creating distinct Classes of Service (COS) in a network, each with reserved resources. A second approach being formalized by the Differentiated Services (DiffServ)[2] working group, utilizes a resource reservation approach initiated by applications (e.g. RSVP). A third

[1] See: http://www.ietf.org/html.charters/intserv-charter.html

[2] See: http://www.ietf.org/html.charters/diffserv-charter.html

approach is to use commercially available traffic shapers and policers to enforce QoS on traffic streams. While each approach has its own strengths and weaknesses, networks will probable evolve to include aspects of each. No matter which approach is to supplying QoS to traffic streams, higher level management will be required to coordinate, arbitrate, and synchronize the approaches at the service and network levels.

The first problem in implementing QoS routing is to differentiate traffic streams, so that they may be classified and priorized according to policies. IP Traffic streams can be differentiated by the following information

- IP address: DHCP may group computers in ranges which can used to classify traffic streams. The limitation of this approach is that an assumption is made of a one to one mapping between computer and user.
- Protocol: Protocols may be identified by the protocol field in IPv4 packets (or Next Header in IPv6). The limitation of this approach is that only level 4 protocols may be identified (web traffic uses http/TCP/IP which would be difficult to identify).
- Port: Port field in TCP and UDP can be used to uniquely distinguish traffic streams. The limitation of this approach is that higher level protocols must be decoded (slower and more complex), and there is no one to one guaranteed mapping between ports and application.
- Priority: TOS (IPv4) and priority (IPv6) may be used to distinguish classes of service. This approach is suitable for traffic with a heterogeneously controlled environment, but inter domain traffic streams may require redifferentiation (as each domain may not accept the assigned classification).

Although the above information is used for state-of-the-art IP management, it is necessary to differentiate traffic by the following, in order to impose more complete accountability to network usages:

- Users: We want to identify the individuals sending or receiving traffic. Users may be differentiated as people, groups, organizations, or any other role based classifications.
- Service: Identify application, or service (WWW-browser, WWW-server, email, internet telephony).
- User Priority: An indication of relative priority assigned to the traffic stream by the application.

A consequence of this approach is that the new enhanced IP management will bring the internet to resemble an increasingly session-oriented, connection-oriented, and stateful network, much more like a telecommunication network. For example, Layer 4 routers will carry service type information, and it is plausible to expect that there will be Layer 5 routers which will carry the rest. It should be no surprise, since telecommunication networks have been traditionally designed for real-time voice traffic.

PBN can, conceptually, support QoS interworking of the above enhanced

(L5) IP management. A policy would describe priority of traffic conditions per service and per user basis for IP differentiated services. Another policy would describe scheduled reservation of network resources for IP guaranteed services.

Quality of service classification may not be consistent throughout a network. This can be exemplified by a network with a VPN provided by an ISP. Independent of the number of service classes, and the traffic differentiation employed in internally managed networks, different classes and differentiation techniques may be applied by the ISP.

3 Policy-Based Networking Physical Architecture

Several policy-based management systems have been announced lately. In particular, Directory Enabled Networks (DEN)[1] consortium (now part of the DMTF[2]), and the IETF Policy working group[3] have attracted significant attention in enterprise network management community. There are two complimentary aspects to the PBN architecture. We must have a physical architecture which can provide the necessary databases in a distributed manner to all the network elements. Secondly we also need an information schema for the policies themselves. In this section, we focus on the generic physical architecture of policy-based networking, which are common to all the existing PBN approaches. We address the information schema issue in Section 5, PBN Information Schema and Policy Abstraction.

The physical architecture of a PBN is required to provide a logically centralized database which can be accessed by network elements throughout the network. The basic architecture of a PBN is shown in Figure 1. In this model, the architecture consists of three types of entities: directories, policy servers, and network elements.

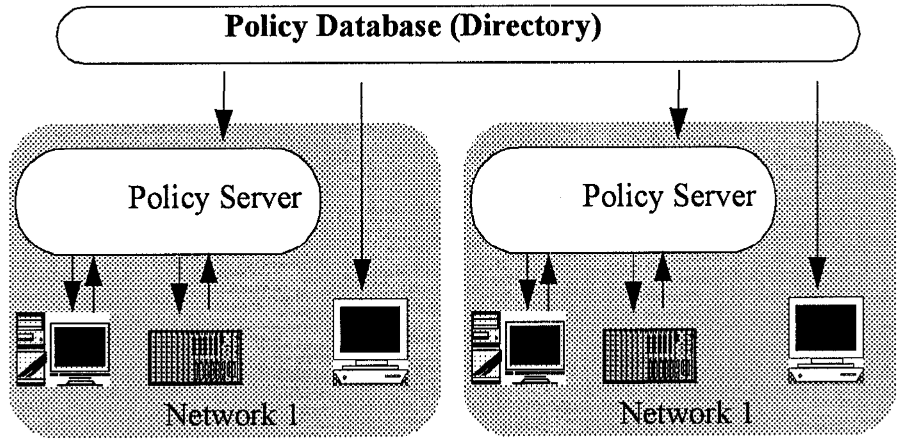

Figure 1: PBN Architecture

The primary building block of a PBN architecture is the directory service. A directory is a simple database, designed for maintaining large static information bases. The three main choices for directory service include: LDAP based servers such as Netscape[4], NDS from Novell[5], and Active Directories from Microsoft[6] (to be released

[1] See http://murchiso.com/den

[2] See http://www.dmtf.org

[3] See: http://www.ietf.org/html.charters/policy-charter.html

with NT version 5.0). Each of these solutions offers a simple distributed database, with LDAP[10] compatibility. The primary difference in integration into existing network management solutions and environments.It is important to realize that these directories services are being used as part of non PBN services such as userid databases, security databases, and application databases. PBN will utilize the same database structure, and thus, can import data from many directory enables services, and also make policy information available to other services. Although LDAP standardizes the exchange of information, the information schemas will be service specific.

The directory architecture will normally consist of a distributed collection of databases. Each database will have a single master, which is responsible for maintaining consistency. The overall directory structure will create a logically centralized database. The main limitation of directory databases for network information is that the single master architecture limits the suitability to dynamic data. The directory service is essentially distributed for read operations, and centralized for write operations. Originally developed for white page services, directories are well adapted for storing userid information, security keys, and static policies, and static network information. If the data in the directory is changing frequently, the master server will be overwhelmed. In the event of a failure in the master directory, non master directories may take over the master role, until the master recovers. Current standards efforts by the IETF are focusing on replication standards for LDAP to efficiently distribute database information[1].

The policy server in the PBN architecture acts a bridge between policy information stored in the directory, and devices in the network. Directory Enabled devices are those which read information from the directory service, others may require a policy server to interpret policy, and configured the device accordingly. COPS[11] and Diameter[12] are two proposed standards for communication between policy servers and devices.

A policy server will also be beneficial in the case where multiple configurations must be made in order to implement a policy, and the configurations must be synchronized. For example, consider the case where network traffic needs to be engineered to allow for increase high priority traffic on one link, by moving existing lower priority traffic to an alternative path. To implement this policy change, we must first redirect existing traffic to the new route, before changing traffic priorities on the current path. This two step procedure would be most easily handled by a policy server which co-ordinates the activities of multiple devices.

4 Enterprise-Public PBN Interoperation

Since QoS is an end-to-end concept, QoS interworking between enterprise and public networks is an essential part of it. When QoS interworking is realized by PBN, both in enterprise and public networks, it is ideal that the two PBNs interoperate, to support end-to-end QoS.

[4] See http://home.netscape.com

[5] See http://www.novell.com

[6] See http://www.microsoftcom

[1] See http://www.ietf.org/html.charters/ldapext-charter.html and http://www.ietf.org/html.charters/ldup-charter.html

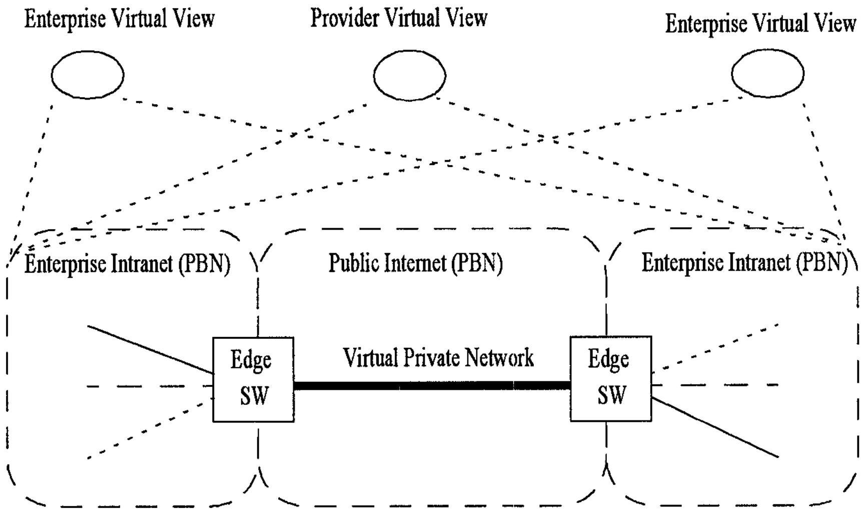

Figure 2: Enterprise-Public PBN Interopera-

Figure 2 illustrates the concept of enterprise-public PBN interoperation. Two enterprise domains are owned by a corporation, and the two domains are connected by a VPN, offering end-to-end QoS differentiated IP services. Both enterprise intranets and public internet are operated by PBNs, though they belong to separate administrative domains. Although use of PBN in public internet is not common yet, we expect that PBN will start enjoying wider acceptance in the public internet as well, due to its clear technical merits. Edge switches are of particular interests, as they are primarily responsible for supporting end-to-end QoS from enterprise point of view. Edge switches/routers may also provide other supporting functions such as firewall, wireless support etc. Regarding network/service management, edge switches need to support the following functions, in the order of the level of integration:

- Management view translation
- Policy interoperation
- SLA management

Higher the level of integration, more flexibility in interoperation is expected. In the rest of this section, we explain more details of the above edge management functions.

4.1 Management View Translation

Each domain has its own view on the network resources, in particular on those resources beyond the domain boundary. For example, users in enterprise domain may or may not see the VPN in the public internet, but it must be visible as a separate resource from the management system in the enterprise domains. The way it looks to the enterprise domain, however, depends on the exposed management views from the public internet domain to the enterprise domains. For example, a VPN may look like a simple wire, or perhaps a wire with more sophisticated call control interfaces for third-party call control.

The same is also true for the public internet domain. An enterprise domain may look like an endpoint, or perhaps an endpoint with more sophisticated management interfaces such as those used for QoS negotiation. In either case, two domains, both enterprise and public, have specific view (virtual view) on resources in other administrative domains in the network. These virtual views are exposed from one domain to the other, or in other words, these management views are translated at the edge node.

A virtual view may be set up based on a long-term contract. In this case, the virtual view is static, and once it is set up, only the network manager of the respective domain can change it. A virtual view, however, can be made more dynamic, per user or per session basis. It will give more flexibility and greater use of network resources to the end-users in the enterprise domain. There must be a standard view on the network resources. Unfortunately, there seems no industry standard in sight with this regard, possibly except for TINA network resource information model [2][3]and related reference point interfaces.[1]

4.2 Policy Interoperation

Two PBNs can interoperate, if one policy of one domain is interpreted and understood by the other domain. Although it must be assumed that two domains share a common management information model, for the current application domain, QoS interworking in IP management, it would not be too difficult to reach a consensus. For example, a user's traffic stream differentiation information discussed in Section 2 can be formulated as a policy in the enterprise domain. The policy in turn is exposed to the public domain, to initialize the user's traffic differentiation policy in the public domain, eventually selecting an optimal COS for the public internet connection under the policy.

As it can be observed in the above example, policy interoperation is only a part of a larger issue, management view translation. Since it is usually not the case that the two domains, the enterprise and the public, have the same provisions and management requirements, 100% policy interoperation would not be achieved. For example, a user's policy may request a QoS guarantee for a particular service type, but the VPN bandwidth, which is based on a contractual agreement, may not be sufficient to support the guarantee.

4.3 SLA management

Service Level Agreement (SLA) [4] is a contractual agreement between an end-user and an operator, or between two operators, which specify QoS level to be maintained and monitoring conditions. SLA is usually an off-line agreement rather than on-line agreement that can be dynamically set up per user or per session. SLA management can be seen as a special case of policy interoperation, where some QoS parameters are extracted from a corresponding policy, then they are formatted into an interoperable form (SLA), which is then passed on to the public internet operator. There needs an industry standard to assure interoperability, and TMF [5][6] is working toward the goal.

5 Scalability of Policy-Based Networking

[1] Information on TINA-C and its specifications are available at http://www.tinac.com.

In creating a scalable PBN we need to have an understanding of the number of policies required in the network as function of network size. As a starting point, if policies were applied in a flat manner, with specific policies to each user and application, we would observe a unmanageable number of policies. We will generally assume that the size of a network is proportional to the number of users. Let N be the network size, and U be the number of users.:

$$N \sim U$$

As a rule of thumb, we estimate the number of policies (P) in a flat PBN domain as being proportional to the number of users multiplied by the number of services (S).

$$P \sim U \times S(N)$$

The number of services on a network is often dependent upon the size of the network, and also on the size of the connected networks. As an example: as more people are connected to global networks, more services will be requested, including ones utilized by only small portion of the network user base. The situation is more drastic when an network is connected to the Internet, where new applications appear daily, and new application and services will be used by any number of users within the enterprise. This model would reflect a exponential number of policies based on network size. We would require policies for every users and service they use. While this approach sound ridiculous, it is the approach employed by access list based management which is appearing in first generation PBN networks.

The use of Role based policies can reduce the number of required policies, but reducing the number of distinct users (groups of users are replaced by a role). While this reduces the number of required policies, it would still need to manage the large number of services.:

$$P \sim S \times \ln U$$

The same relation will also hold for networks grouping users based on IP addresses, assuming IP address management has been allocated IP addresses in a hierarchical manner (using subnet masks to match hierarchy). This is not always possible in multi-homed networks. In networks with out hierarchical IP addresses, the flat policy estimate will likely hold.

To maintain scalability we need to classify the services into a fixed set of classes, independent of network size. This is critical as the lowest level of network infrastructure can only support a fixed amount of priorized classes c (generally 8 -16). If c represents the number of classes of service, we have achieved a scalable number of policies.

$$P \sim c \ln U$$

If we ignore user identification, and focus only on services for differentiation, we would have a scalable solution.

$$P \sim c$$

In the current IETF QoS proposals, user identification is accomplished primarily through IP address, and number of classes limited by hardware limitations. LAN supporting IEEE 802.1p and q will typically have up to 8 queues, IP based routers will typically have 16 queues, and ATM based networking equipment may have 32,000 queues (per VC queueing).To ensure end to end QoS, we must generally look at the smallest number of queues, 8.

There is an obvious need for a more scalable solution. There are a few other kinds of scalability issues, namely physical scalability and control path scalability, which we need to consider along with the above management complexity at the same time. In the rest of this section, we discuss these scalability issues in large-scale PBN architectures.

5.1 Physical Scalability

Physical scalability of PBN is a primary concern when the network size becomes large. We briefly take a look at engineering mechanism of PBN, to study its physical scalability. When a PBN policy is activated, the policy is interpreted, then it is put into effect by using management interfaces of network elements involved with the policy. Policies are stored in a directory server, and they are accessed through standard directory access protocol, e. g. LDAP.

There are two mechanisms known, in which the directory server and the network elements interact differently. In active networks, network elements are active, and they are able to pull policies from a directory server spontaneously. In passive networks, network elements are passive, and the directory server must push policies onto network elements, often using conventional management protocols such as SNMP. These two models, however, are not exclusive, and it is possible that a network have both active and passive elements in its domain. The difference between the two models does not seem to affect physical scalability of PBN, and it is rather a balance of traffic load and intelligence in the network, between the server and the network elements.

Physical scalability was already well understood in X.500 series specifications. A set of cooperative distributed Directory System Agents (DSA) can partially replicate or cache policies, maintaining scalability and integrity of the policy set.

5.2 Control Path Scalability

Policies need to be propagated, either by push or pull, to the network elements in the PBN domain. The protocol being used in this propagation step, therefore, must be scalable. It is possible that the control path scalability becomes of grave concern, when the number of network elements are large and traffic between network elements and the directory server are busy and voluminous. In the QoS interworking issue, however, we do not expect that the control path scalability is a critical factor, since expected traffic between directory servers and network elements (switches/routers) is relatively low in volume. Typically, a user specific QoS differentiation policy is pushed onto network elements along the path across the domains at the beginning of a session. Once QoS provisioning is done, there is little need of interactions for the session between the network elements and the directory server.

A few observations to be made on this directory-based control are; (1) management and control are unified, and a single directory access protocol (predominantly LDAP) is expected to serve as a universal vehicle for most of the management/control needs in the network, (2) the directory server now acts as the single point of control in the network, shifting the weight of distributed network management onto the directory server.

5.3 Management Scalability

By management complexity, we imply the complexity of network management proportional to the number of policies. This is the main theme of this paper. In the following part of this paper, we focus on the management salability of large-scale enterprise-public interworking PBNs, presenting a few of our ideas to cope with the problem.

6 PBN Information Model and Policy Abstraction

As we discussed in the previous section, scalability is posing the greatest challenge to wider deployment and end-to-end applicability of PBN and QoS management in large IP networks. The issue, however, is a well-known one when policies are viewed as active entities, executables of distributed software. To manage the complexity, a set of policies can be organized by two orthogonal manners. The first approach, which is presented in this section, is to use abstraction hierarchy. Policies are ordered by the degree of abstraction, and are classified in such a way that concrete (specific) policies are generated from a fewer number of abstract policies, by parameterizing the abstract policies, or by rule-based translation from the abstract policies to concrete policies. The idea of policy abstraction was introduced as policy hierarchy by Moffett and Sloman [7]. We illustrate this concept using an example from QoS routing policies.

6.1 Complexity of QoS Routing Policies

The primary difficulties in implementing a QoS policy network is the differentiating of traffic streams by the groups specified in the policies. While policies could be strict groupings of users, and QoS, parameters, this would not be a feasible policy system for a large enterprise network. A backbone router may have to contend with 30,000+ active connections simultaneously, and can not afford the overhead of simple groupings for policy specification.

Specific Policies for QoS routing should contain the following information:

- Identity of traffic streams. Information used to distinguish traffic streams.
- Minimum and maximum QoS. Contains specific limits on the resources which the traffic stream will require.
- Priority. In the event of conflicts for traffic streams requesting similar resource, the priority information will be used to resolve these conflicts.

- Identity of policy source. Feedback from the network management system will use this information to report conditions regarding this policies.
- Enabling conditions: A set of conditions which must be met before policy is applied. An example would be time of day restrictions.

6.2 Policy Abstraction

Abstract policies will have more generalized content, referring to groups of traffic streams and classes of services.

- Requesters. A set of users/applications.
- Resources. Network resources.
- Identity of policy source.
- Enabling conditions.
- QoS specification
- Priority specification

While both the specific and abstract policies have the same general form, the difference is in the quantification of resources and requirements. Abstract policies will deal with categories of information, whereas specific policies need to be unambiguous. The specific policies need to be sufficiently straightforward, that no interpretation of specification will require external information or knowledge.

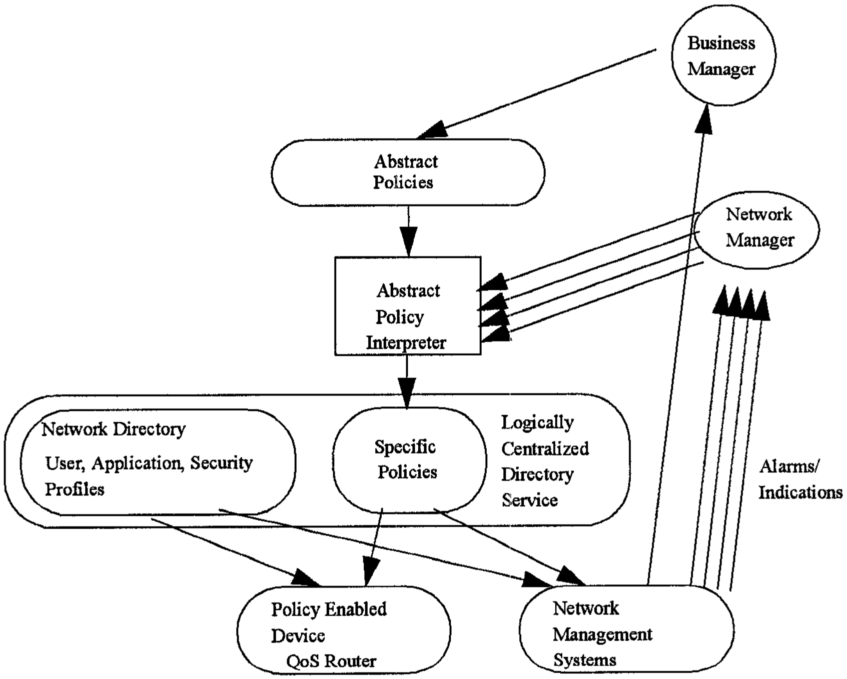
Business Manager
Abstract Policies
Network Manager
Abstract Policy Interpreter
Network Directory
User, Application, Security Profiles
Specific Policies
Logically Centralized Directory Service
Alarms/ Indications
Policy Enabled Device QoS Router
Network Management Systems

Figure 3: Abstract Policies for QoS and Network Man-

In Figure 3, the policies in the directory service, are made available to two systems: The Policy enabled devices, and network management systems.

Traditional network management systems are complex systems capable of network monitoring, configuration, and control. One of the greatest difficulties in implementing such systems is dealing with the large amount of data which must collected and analyzed. Large network management systems are designed to support different levels of hierarchy in management, and also geographically separate networks.

To make a management useful, selective filtering of network measurements must be implemented, otherwise network managers would be overwhelmed by non-critical alarms.

In our proposed architecture, we envision using the policies which govern the network as a specification of filtering capabilities for the network management system.

- Consistent interface for network managers
- Support for different levels of management

As it can be expected from Figure 3, there is an additional benefit of using policy abstraction. When abstraction policy interpreter is being made dynamic, i. e. specific policies are generated on-the-fly, it is possible to act alarm/monitoring conditions proactively, such that seemingly spurious alarms would be cut-out at policy-level. In other words, those non-critical alarms will not be generated from PBN controlled network elements.

7 Policy Divergence and Precedence Rules

The other approach toward the management complexity of PBN is through an aggregation hierarchy. This approach may be more accessible from enterprise point of view, since enterprise bodies are hierarchically organized in nature. Corporate networks will be managed by several levels of hierarchy, some with physical objects directly under their control, and others with only with goals under their control. As an example, the president of a company will have the highest level control of a network, and may specify policies which are related to business objectives of the corporation. At his level of administration he does not have direct influence over, and hardware or software resources. On the other hand, a director of the networking department will have ultimate responsibility for the entire network operation, but is not directly involved with LAN management. He however, will have management control over the backbone infrastructure.

One of the major issues of large-scale PBN is that it becomes increasingly difficult to detect policy conflicts within large policy sets, and it also becomes difficult to predict possible outcomes consequenced by conflicting policies [1][9]. We call this phenomenon as *Policy Divergence*. Although it may be difficult to completely remove policy conflicts, it is still possible to resolve the conflicts locally, to set up appropriate precedence rules. Take an example from the QoS interworking problem. Suppose an employee at VirtualWorld corp. would want to access an web server at FanstasticView corp using a 10Mbps guaranteed CBR connection. Though he may have obtained a permission from his department head and his department policy allow it, the corporate policy may still restrict him to set up a connection outgoing from the corporate edge node at 5Mbps maximum.

Though this is a simple example, the case may become more complicated when the user has separate policies for different applications. For example, he may observe that his ftp connection is granted whereas an access to a video source on a web server may be denied, due to different impacts of different QoS provisioning. In usual organization hierarchy, a corporate policy overrides departmental policy. Figure 4 illustrates organization hierarchy and precedence rules to resolve potential policy conflicts. VirtualWorld corp. has a headquarters in Tokyo, and two departments in London and San Jose, respectively. We suppose that policies are conflict-free at respective locations. For example, policies at London department is conflict-free, if they are used independently from other locations. Problem arises when these policies are linked, in order to cover the larger enterprise domain.

Under this assumption, policy divergence is introduced by linking different departments into a single enterprise domain. By attaching precedence rules to interdepartmental links, therefore, it is possible to resolve potential conflicts in the linked domain. When corporate policies prepared at Tokyo headquarters are linked with policies at London department, it is assumed that corporate policies override, if conflict arises. The precedence rule (B < A) associated with the link tells that the set of policies at Tokyo (A) override the set of policies at London (B).

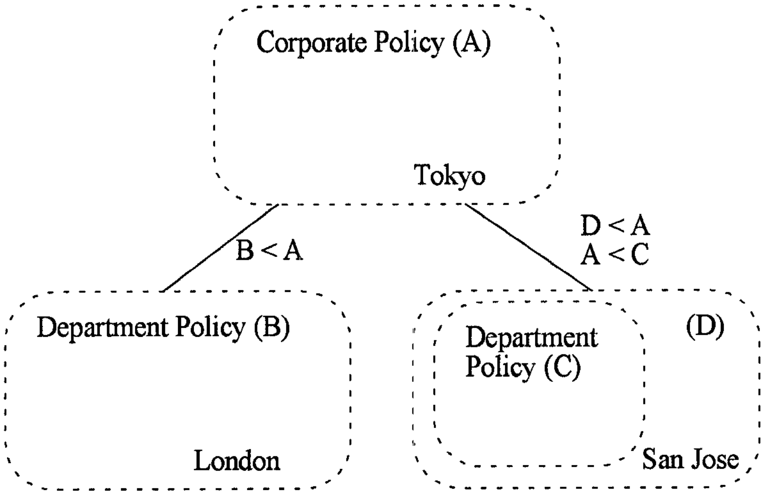

Figure 4: Organization Hierarchy and Precedence

It is, however, not all the case that the corporate policies override local policies. In the figure, there are two sets of policies at San Jose department, C and D. Although corporate policies (A) override one set of department policies (D), the other set (C) override corporate policies, in order to conform to local governmental regulations.

8 Conclusions

In this paper, we studied Policy-Based Networking (PBN), and applied it to QoS interworking problem in IP management. We have found that the management scalability is becoming the major challenge of large-scale PBN operation. In this paper, we presented two ideas to handle the management complexity, namely policy abstraction and organizational hierarchy with precedence rules. We believe that a scalable PBN architecture based on these two architectural ideas are essential for the future development of PBN systems, and all future PBN management tools will take advantage of the scalable PBN architecture.

Reference

[1] M. Sloman, "Policy Driven Management for Distributed Systems," Journ. of Network and Systems Management, Vol. 2, No. 4, pp.333-360, 1994.

[2] N. Natarajan, "TINA Network Resource Information Model," Journ. of Network and Systems Management, to appear in Vol. 6, No. 3, 1998.

[3] TINA-C, *Network Resource Information Model Specification*, Version 2.2, Nov. 1997.

[4] ITU-T, *Terms and definitions related to the quality of telecommunication services*, ITU-T Recommendation E.801, Oct. 1996.

[5] NMF, *NMF Technology Map*, NMF GB 909 Issue 1.0, Mar. 1998.

[6] NMF, *NMF Telecom Operations Map*, NMF GB910 Draft 0.2b, Apr. 1998.

[7] J. D. Moffett, M. S. Sloman, "Policy Hierarchies for Distributed Systems Management," IEEE Journ. of Selected Areas of Communications, Vol. 11, No. 9, pp.1404-1414, Dec. 1993.

[8] A. Campbell, K. Hahrstedt, "Building QoS into Distributed Systems", Chapman and Hall, 1997.

[9] Emil Lupu, "A Role Based Framework for Distributed Systems management", PhD. Thesis, Imperial College of Science, Technology, and medicine, University of London, 1998.

[10] M.Wahl, T. Howes, S. Kille, " Lightweight Directory Access Protocol (v3) ", RFC 2251.

[11] J Boyle, "The COPS (Common Open Policy Service) Protocol", ietf draft: draft-ietf-rap-cops-03.txt.

[12] Calhoun, Rubens, "DIAMETER", Internet-Draft, April 1998.