Chapter 5

Methodology for selection and evaluation of policies: Application for routing management.

Current networks introduce more and more sophisticated services and this is the reason why it is necessary to provide with new management schemes adapting to the changeable network conditions and to the higher and higher QoS requirements.

In this Thesis, a powerful scheme that allows managing networks based on heterogeneous environments both at the hardware level and the software level is proposed. In those networks the edge nodes status, the core nodes status and the server's status change constantly. In general, the status of all the different network components is considered as dynamic.

A global vision hiding all implementation details of every particular network and considering the business goals that every company establishes for its network is necessary in the interconnection of heterogeneous environments. Some examples of business goals

can be: A telephony network will give priority to voice communications while a book sale company maybe can be more interested in electronic trade operations.

This chapter proposes a methodology to evaluate and select the management policies that must be applied in the network considering a lot of different factors, as for example, the network status, the class of service, business goals, etc.

This methodology is applicable in any functional area of the system. In this chapter, its use concerning routings with QoS constraints is proposed. In spite of the fact that several researches have been carried out concerning routing based on QoS restrictions, for example [Lee95], [Chen98], the problem about routing among different network environments (with different QoS levels) has not been solved. This research is focused specifically on a proposal to improve the routing management using a policy server that guarantee different Quality of Service levels for intra and inter-domain heterogeneous environments.

This chapter ends with a proposal about a policy-based algorithm for path selection.

5.1 Policy-based routing

As we mention before, there are different proposals to improve the QoS-based routing in high-scale networks. We are going to analyze how to solve the QoS constrains of routing using policies. In this way, the IETF designed a framework of QoS-based routing in the Internet [Crawley98].

Within the existing architectures proposed for policy-based routing, this Thesis proposes an innovative QoS policy-based routing management system that it is scalable enough to be implemented in interconnected heterogeneous networks and to be applied not only in routing management, it could be applied in all the functional areas of the system, it means that the system allows obtaining synergies with respect to other functionalities such as failure management, traffic management, congestion (traffic engineering), rate settings, etc.

It is also considered as a general methodology to cope with the problems of any network in an efficient way.

The PBMS proposed consider a routing based on explicit paths from a database owning updated information about the entire network. The policy server computes paths that are subject to multiple constraints, including both QoS constraints (QoS requirements and resource availability) and business constraints. The methodology proposed lies on the differentiated service technology [Blake98], [Kilkki99] in conjunction with MPLS [Faucheur02] in order to guarantee the QoS requested by the user of a specific connection.

Considering several network parameters or services as a whole to define a policy (unlike other systems where only the bandwidth is taken into account) allows obtaining a higher efficiency. For example, in the mobile communication system management, the fact of considering losses or delay is an important factor in routing.

The following are typical routing circumstances that can be easily solved by means of a PBMS

1. The network administrator does not want a protocol to import all paths into the routing table. If the routing table does not learn about certain paths, they can never be used to forward packets and they can never be redistributed into other routing protocols.
2. The network administrator does not want a routing protocol to export all the active paths it learns.
3. The network administrator wants to manipulate the path characteristics, such as priorities, characteristics to control which path is selected as the active path to reach a destination.
4. The network administrator wants to change the Class of Service for a specific service at a specific hour of the day. For example, a telephony company could establish as one business goal that users connected between 10:00 and 17:00 hrs has as priority service the IP telephony instead of other services such as videoconference and commercial transactions.

PBMS use for the routing management is compatible and can be integrated with traditional routing protocols like Open Shortest Path First (OSPF) whose starting point is a cost function [Zinin03] [Cahpin92] or mechanisms like traffic engineering that normally use optimisation algorithms that consider only one metric (bandwidth, hop count, cost).

This chapter presents contributions to PBMS taking into account several metrics for the routing process. This proposal considers the interconnection of several heterogeneous management domains. In order to solve the intra-domain routing, applicable policies to the DiffServ-MPLS Networks and other management techniques based on policies.

5.1.1 Policy-based Intra Domain Routing

A policy-based scenario, which is compatible with a DiffServ-MPLS network, is proposed to solve the routing within the same management domain. This section shows the design and implementation carried out to select an intra-domain path.

An intra-domain path makes reference to a connection among edge routers and an inter-domain path makes reference to a peer-to-peer connection. It, that is to say, a path going from a source to a destination. An inter-domain path can consist of one or more intra-domain paths. In figure 1 the blue colour shows the intra-domain path 1 and the green colour shows the intra-domain path 2. Both paths together form the inter-domain path requested at the user's connection. As figure shows, it is necessary a link between both domains (red line), it is possible to use border gateways protocols as BGP4 [Rekhter95], [Bates00]. In the implementation (see appendix II), the edge router is in charge of differentiating among intra-domain paths and inter-domain paths.

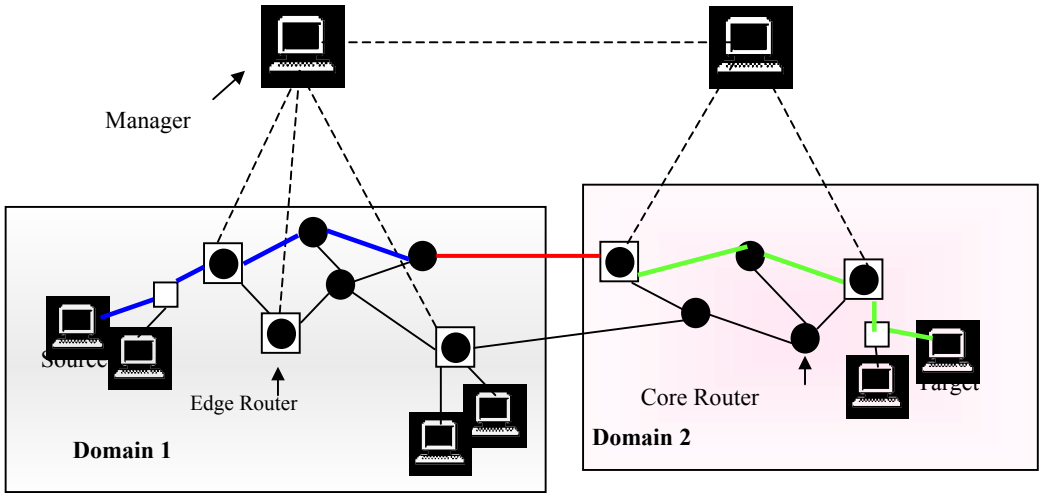

Figure 1. Network Management intra and inter- domains

Efficient QoS techniques are used in order to select the intra-domain path that the information flow must follow: Differentiated Services and the Mulit Protocol Label Switching (MPLS). DiffServs offer different QoS levels in IP networks and its main contribution with respect to other QoS technologies (IntServ, IP flow through ATM, etc.) is the fact that it differentiates service levels in the same network in a scalable way [Kilkki99]. The service level that a specific user receives indicates how their packages will be treated, taking into account that these packages arrive according to the SLS profile assigned to the connection.

DiffServs recognise traffic flows belonging to different Classes of Service using the DS byte from the packages head; in the IP version 4 this field is called Type of Service (TOS) and in version 6, Class of Traffic (CoT) [Huitema98]. Table 1 shows the parameters that the field Type of Service of IP protocol has.

Bits 0-2	Precedence
Bit 3	0= normal delay, 1=low delay
Bit 4	0= normal performance, 1=high performance
Bit 5	0= normal reliability, 1=high reliability
Bits 6-7	Reserved for a future use

Table 1. Bits of the field TOS of IPv4

In Ipv6, the DS field allows the differentiation process of traffic and the possibility of discard in case of congestion. The DS byte owns a code point DS, that is to say, a mark that specifies the per-hop behaviour (PHB) provided for every package. A PHB can specify either the package priority or can include its execution characteristics. Table 2 shows the DS field parameters.

Bit 0	Uncharacterised traffic
Bit 1	Filling traffic
Bit 2	Non-assistance data transference, e.g. NetNews
Bit 3	Reserved
Bit 4	Assisted data traffic, e.g. FTP, NFS
Bit 5	Reserved
Bit 6	Interactive traffic, e.g. Telnet
Bit 7	Traffic control on the Internet, e.g. routing

Table 2. Bits of the field DS of Ipv6

The value of every bit of the ToS field of IP4 and the DS filed of IPv6 is established by means of policies. Our system configures policy actions related to business goals, security or marketing policies for those bits that stay as reserved.

The philosophy of DiffServ networks distinguishes between edge nodes (that carry out some functions such as the admission control, supervision, traffic conditioning and accountability) and core nodes, whose behaviour depends exclusively on the type of service associated to every package. [Escribano02].

Figure 2 shows the way in which edge and core routers are configured with base on the SLS profiles in a DiffSer Network. A package classifier is used at the network ingress. It is in charge of the flow identification for every package and it assigns them the suitable policy, that is to say, the function that distinguishes among those packages that agree with the SLS profile assigned to the connection and those ones that do not. There is also a package marker connected to the policy server. The policy server uses specified parameters in the SLS profile, network status parameters coming from the Oracle database and the policies stored in the LDAP (Lightweight Directory Access Protocol) directory to indicate the package manager to write a code in the routers (which represent the policy action) in every IP package head identifying the type of service that must be applied to the package.

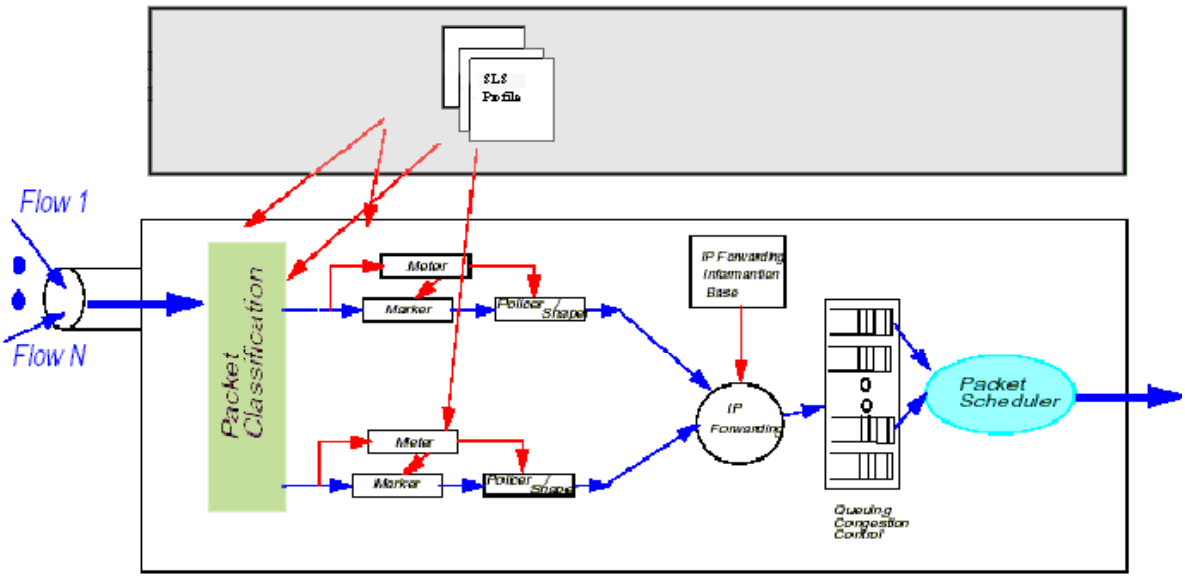

Figure 2. Configuration of DiffServ elements using SLS profiles

A queue manager is used inside the network, which, depending on the PHB, will give priority treatment to some packages as contrasted to others in the queues of node transmission. The queue manager generally consists of a discard priority mechanism whose task is to decide which packages it has to discard in case of congestion at the queues, and a

service priority mechanism that has to decide which package is the following to be transmitted. In this way, the final service offered to every traffic flow depends on the PHBs sequence assigned. All packages found to agree with the traffic descriptor are marked as creditors to a high priority treatment in every jump. The rest of them are marked as out of profile and therefore, they are creditors to a worse network treatment. When these packages reach core nodes, they are assigned a treatment depending on the fact that they are in or out of the profile.

The definition of policies for the core routers specifies the kind of behaviour (priorities, bandwidth, losses limit, delay, jitter, etc.) that is assigned to every type of traffic. This behaviour will be applied in the queues corresponding to the different package marking kinds. Due to the circumstance that policies determine the differentiation level assigned to every class of service, the definition of low-level policies in a differentiated services network consists of a set of rules that determine the behaviour of the network and the devices independently from the details included in every device.

Because of packets are marked just at the edge routers, Diffserv cannot solve the congestion inside the domain. For example, a lot of flows in the same class can be routed through the same link, thus cause congestion there. Policies can route the flows to the paths that have the capacity to accept the flows, or to reject a flow if there is not resource for it.

MPLS labels can be used to establish explicit paths [Rosen01] in order to design and implement the routing information within the management domain. Figure 3 shows a MPLS scenario that indicates the edge router and core routers.

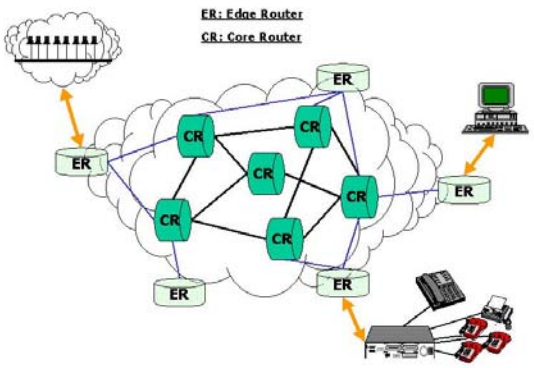

Figure 3. MPLS Scenario

The MPLS scenario is compatible with DiffServ and with policy-based routing. Policies select the path and the MPLS labels do the packet forwarding along the path. The application design to establish intra-domain paths considers that only the communication among elements that can be connected to an edge router can be established. Edge routers routing tables are stored dynamically in the relational database Oracle. The routing information is periodically updated by a monitoring system based on intelligent agents [Barba02] [Reyes02-1].

The following figure shows the routing tables' format that every edge router owns.

RouteID	CoS	Label	Next_CR	Destino
1	1	3794237498	CR33	ER2
2	1		CR12	ER3
3	1		CR20	ER4
4	1	3247923749	CR12	ER5
5	2		CR33	ER1
6	2		CR15	ER2
7	3		CR12	ER3
8	3		CR20	ER4
9	4		CR12	ER5
10	4		CR33	ER1
11	1		CR33	ER2

Tabla 3. Routing tables format

The meaning of the different columns is the following:

1. RouteID. It is an identifier of the necessary path in the dynamic path selection and creation process.
2. CoS. It is the class of service offered in this route.
3. Label. It is the MPLS identifier. In case this field is empty, that fact means that the path is available and it is not being used.
4. Next_CR. It is the identifier of the following core router through which the path goes. All packages belonging to this session will be sent to the core router specified in this column in the following hop. There will be as many values as contiguous core routers to the edge router.
5. Destination. It is the edge router destination identifier to which all packages of this connection go.

Events produced in the network make tables increase or decrease. In this sense, other agents or applications can add paths dynamically in case the capacity and the network quality increase or eliminate them in case it is degraded.

The number of the connection in every path allows the application to mark the paths in the very moment they are released as free. In case there is not this table, the application should seek in all edge routers tables until it finds which path the connection is using. The following table shows the connections in process:

ConnectionRef	Destino
3794237498	ER2
3247923749	ER3
3464646544	ER4
0131548546	ER5
646498794	ER1
465798798	ER2
145678797	ER3

Table 4. Connections in process

Finally, the table 4 was defined to store the accounting of the system because it makes the rate setting and failure recuperation operations easier and contemplates an activity column together with the network ingress and egress nodes.

ConnectionRef	User Reference	QoS	Active	Start_time	End_time	Edge router	Edge router
145678797	2	1	False			ER1	ER3

Table 5. Table for logs

The figure 4 represents a scheme for the implementation process to assign a path to a connection.

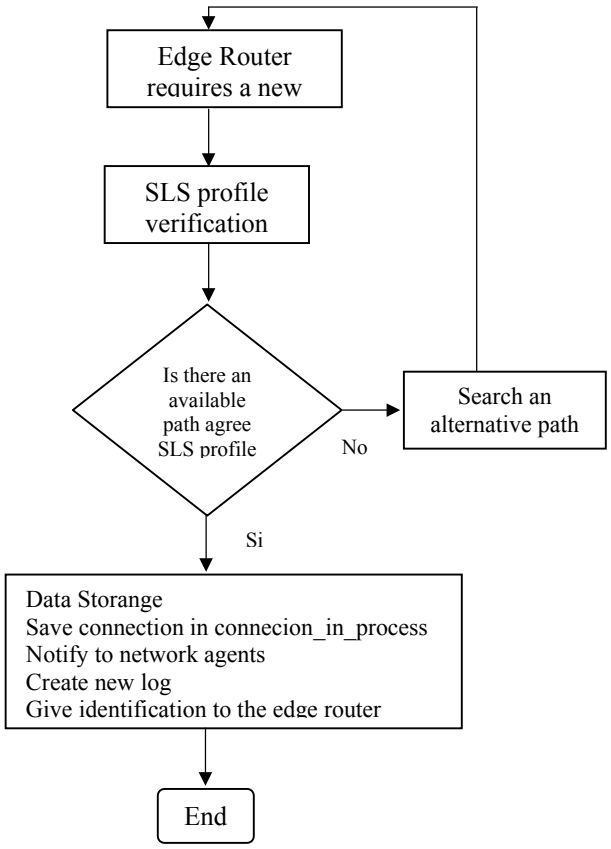

Figure 4. Implementation Process

In order to release a path, we must carry out the opposite process to the assignation one. In this case, the edge router already knows the identifier of the connection that it wants to

close and so, it sends a signal to destruct it. The different stages to close the connection are the following ones:

- Release the path from the paths corresponding to the edge router in process.
- Undo the corresponding connection entry in process.
- Close Log entry.
- Notify the network agents the re-computation of the available resources in the network.
- Notify the edge router the elimination of the MPLS label from its routing table. This action is necessary because a connection can be closed from any place in the network and that place does not have to be necessarily from the edge router that the connection established. As an example we can mention the case of a session close by the network administrator according to security reasons.

The application contemplates a last application that allows reconfiguring the network elements. This operation behaviour is analogue to the creation one in all aspects but concerning the user's parameters check. In case the edge router paths had to be re-written due to some emerging event, then the connection would carry out a new path selection with the consequent modifications and notifications in the path tables and connections in process.

The following figure shows the relation among the different elements that take part in the implementation of the policy-based intra-domain routing.

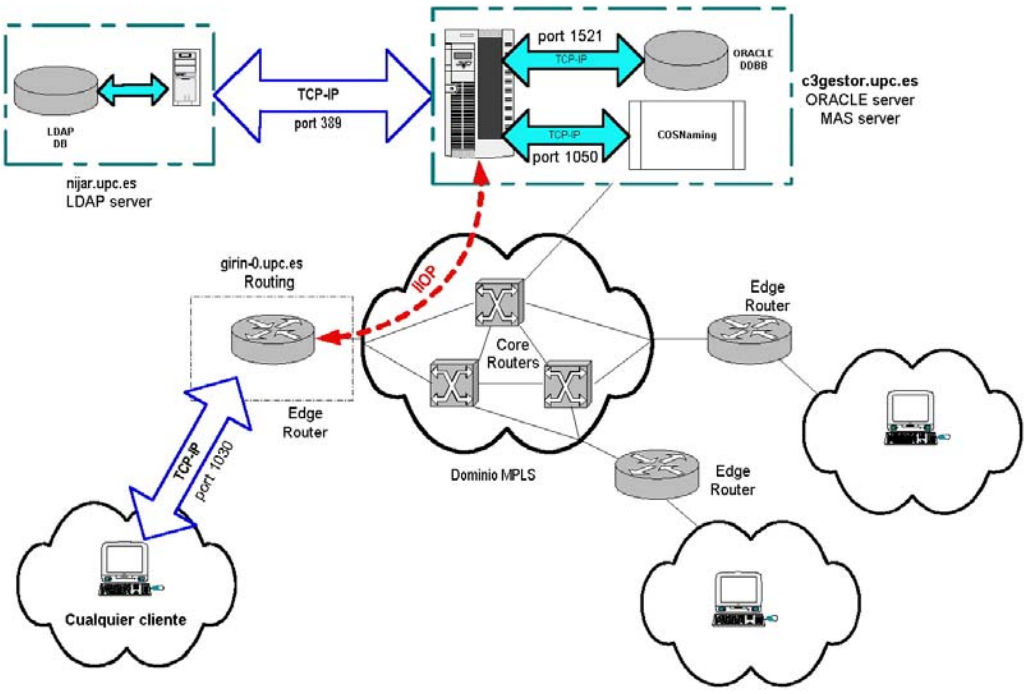

Figure 5. Implementation Scenario

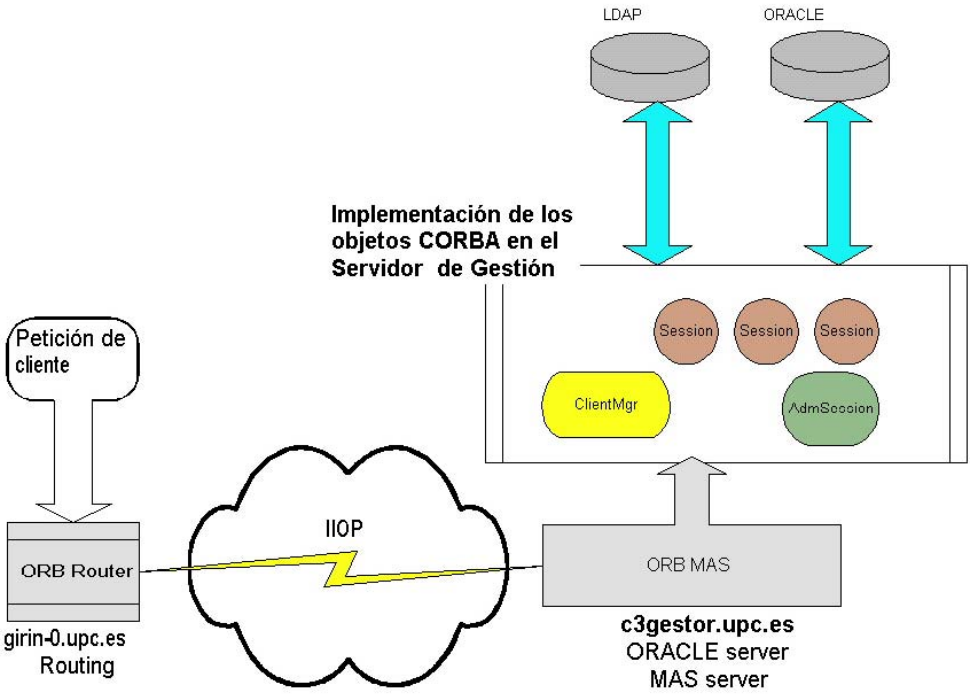

Figure 6. CORBA Application for the PBMS

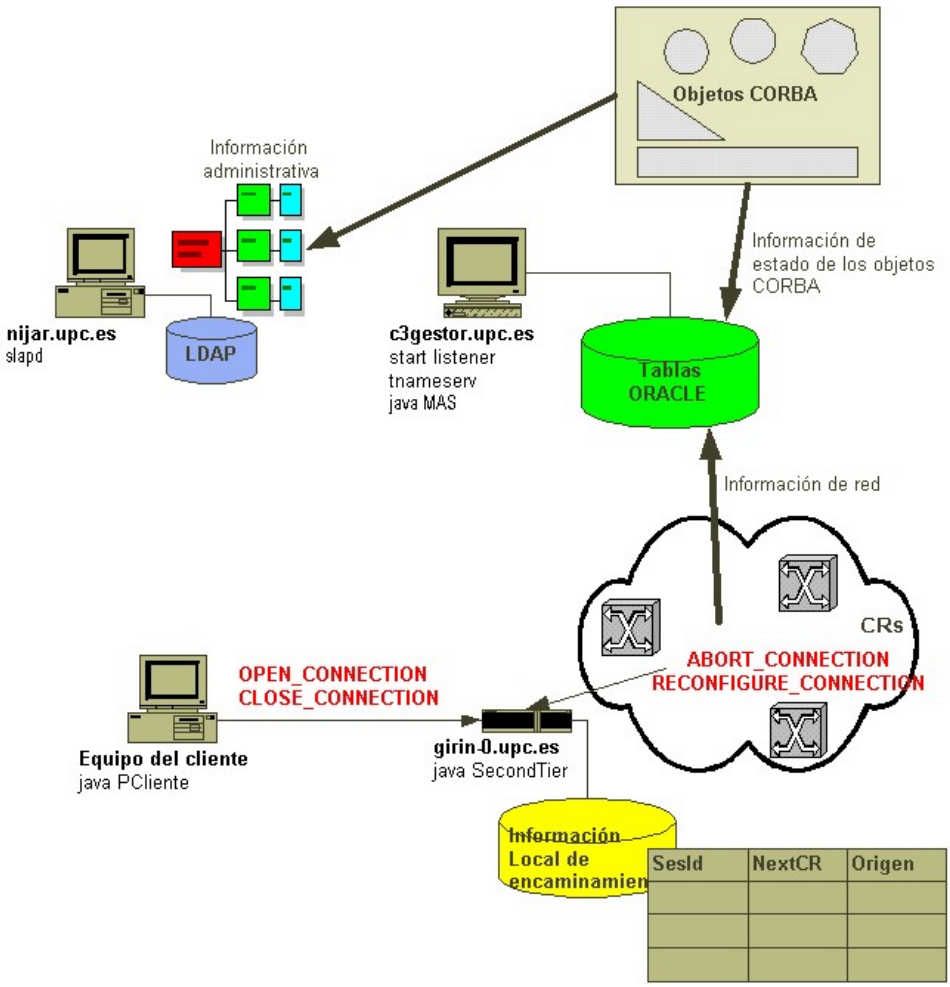

Figure 7. Databases and directories of the system

5.1.2 Policy-based Inter Domain Routing

The global interconnection of communications is presented every time as a bigger need due to the fact that the most offering servers require passing through different network domains. The great heterogeneity of the hardware elements that form part of every network, the software differences that they use and the divers business goals that every company establishes for the provision of its services cause that the interconnection of different network domains requires new management schemes to reduce the technological expenses

of the network operation, the maintenance complexity and the optimisation of the resources use.

The objective of the policy-based Inter Domain Routing is to construct and maintain routes, between source and destination management domains, that provide user traffic with the requested services within the constraints established for the domains transited.

The proposed policy-based inter-domain routing system considers as a main challenge to provide a QoS specific level through multiple heterogeneous network domains. First some definitions are provided.

- Peer-to-peer connection: It is considered as a whole of different domains related among them. A network management domain is understood as a group of different network items (network nodes, links, switches, routers, etc.) defined according to geographic and technologic criteria or other characteristics, and a combination of these criteria.
- Network domain: It consists of the network items that share the same management authority. Therefore, a domain manager compiles the information from the network items belonging to its domain and the necessary mechanisms for its configuration.

In a network scenario with multiple domains, the manager of each domain acts as collectors and distributors of routing information between domains. We propose to use policies to solve the distribution of information via CORBA protocols for the communication. The routing information mainly consists of connectivity data.

A connection from a source to a destination can require being distributed in more than one network domain. Links to interconnect domains are created for these connections that cross multiple domains, see figure 1 at the beginning of this chapter. In this way, the complete connection would be the addition of all domains through which that mentioned connection goes and the existing links among these domains and the user in a transparent way.

Policies participate in several processes, first of all an admission control process is made, due that each domain establishes their rules and normative to use their network elements. For example, some domains can prohibit the use of certain routers or even paths for security reasons or for business goals to extern users.

Once the all service provider accepts the connection, policies control the connection setup along the path from source to destination domain. Policies also participate in the management of routing information in databases. The policy server manages the database with routing information and the policy repository. One important function of the server is to resolve Internet addresses and names to management domains.

A dynamic database maintains information regarding to path-agents and route servers. It can happen that some network elements belong to several domains simultaneously and considering the fact that every domain defines the management operations that can be executed over its items, some conflicts or inconsistencies, which managers must solve, can emerge. Therefore, we define some mechanisms for the resolution of conflicts that can appear when different domain managers indicate to the same network item to take different actions or even opposite ones (see chapter 6). Figure 1 at the beginning of this chapter presents a network scenario scheme with multiple management domains.

Every domain can own several sub-networks at the same time, even with different technologies. The sub-networks within the same domain can appear due to several reasons; the most evident ones is the fact of dividing the management tasks and establishing different administrative policies in every sub-network. For example, an applicable policy to a company that creates the sub-networks according to the department to which they belong could be “The traffic proceeding from the X direction sub-network must be routed through priority paths” or “The traffic proceeding from the accounts department must no be routed through Y router”.

Within the same domain there can be several routing paths to connect a source with a destination and those paths can require going through several sub-networks. In a scheme

with several interconnected networks, three different kinds of links can be distinguished basically: inter-domains, intra-domains (among sub-networks) and among nodes. The information to manage each of the three kinds of links is different. The links used inter-domains require the knowledge of the network status, and of the adjacent domains that participate in the chosen path. For this reason, elements within a domain generate link state messages containing information about the originating domain, including the set of policies that apply and the connectivity to adjacent domains, and they distribute these messages to their neighbour domains. Based on the set of link state messages collected from other domains, on its domain's source and on the policies, a routing entity constructs and selects paths from its domain to other domains.

Policy-based Inter Domain Routing has several benefits, some of them are:

- Each domain has complete control over policy path generation from the perspective of itself as source.
- The cost of computing a route is completely contained within the source domain. Hence, network elements in other domains need not bear the cost of generating paths that their domains' local hosts may never use.
- Source policies may be kept private and hence need not be distributed. There is no memory, processing, or transmission bandwidth costs incurred for distributing and storing source policies.

5.2 Policy-based algorithm for path selection

In great scale networks the management could require a repository with hundreds or thousand of policies. However, only a subset of policies could be applied in the network, for this reason a main fact for the policy-based routing with QoS restrictions is the path selection process, which considers the required QoS by the connection and the available resources in the network.

The path selection is typically formulated as an optimisation problem for the different links that are involved in the connection of a source with a destination, for example, when a function that reduces the number of hops, expenses, delay or any other measure corresponding to a individual link parameter addition throughout the path is determined.

However, calculations become more and more complicated when several QoS requirements have to be satisfied, due to the divers restrictions that must be added to the optimisation problem. A restriction example on selecting a path could be the fact that the peer-to-peer delay does not overcome certain limit values. Therefore, when the number of restrictions is increased, then the problem becomes untreatable as it was checked in [Wang96] where the problem of finding an adequate path to multiple restrictions is NP-complete.

One of the existing heuristic algorithms to solve this problem is called Sequential Filtering under which a combination of parameters is ordered in some fashion, reflecting the importance of different metrics (e.g. cost followed by delay, etc.) Paths based on the primary parameter are computed first and a subset of them are eliminated based on the secondary parameter and so forth until a single path is found [Crawley98]. This is a trade-off between performance optimisation and computation simplicity.

Depending on the network domain size, the algorithms for path selection could produce great complexity and delay in the network. For this reason, it is important to design efficient and scalable algorithms. We propose to solve the selection problem via methods and heuristics that can choose a path able to cover the QoS required by the connection, while keeping an efficient utilization of the network resources.

The proposed algorithm for path selection uses previously defined policies by the network administrator in order to determine the adequate route that a peer-to-peer connection has to follow. Because of the flexibility inherent to the policy-based management systems, the proposed selection scheme can be used in heterogeneous network domains with different management platforms, protocols, topologies, software, etc.

The algorithm searches for a path satisfying a connection request guaranteeing the specific QoS level, between a source and a destination. There are two basic ways of knowing which policy or set of policies to apply in the network elements, first one via an event occurrence, it means, that an event activate a policy action or a set of policies actions, such as: congestion alarms, router failures, policies related to timetables, etc. The second process consists of inquiring the repository to know which policies could be applied in the network. This process can be applied whenever we want to obtain a specific behaviour in the network without having to wait for any alarm or event trigger. In both methods it is necessary to evaluate each policy condition and when it is true, then the policy action or the set of policy actions has to be applied. This method could be very slow for systems with thousands of policies, for this reason, we structure the LDAP directory in roles, so it is necessary to analyse only the policies that belong to a specific role. In the same way, it is possible to create sub-roles, where each role and sub-role has a default policy that is applied only when any other policy could be applied. Default policies should be generic enough to avoid network inconsistencies.

For each role, we establish a set of metrics to measure the QoS levels, each role considers different parameters to define its metrics. In relation to the particular case of routing path selection we use three metrics that represent the basic properties of the network: bandwidth, delays and losses. This algorithm can be used in any functional area of the system, for example account management, marketing, etc. for which only it is necessary to establish the adequate metrics for every management area.

Metrics define the QoS guarantees that the network can provide. The QoS requirements that are not mapped to a metric or to some combination of them will not be able to be guaranteed. There are interesting works related to the establishment of metrics to provide point-to-point QoS [CHEN99].

Every role metrics own different characteristics; in case of the routing management we consider three aspects: additive, multiplicative, and concave. They are defined as follows:

Let metric m(node 1, node 2) be a metric for link(node 1, node 2). For any path = (node 1, node 2, ..., node i, node j), the metric is additive, if m(path) = m(node 1, node 2) + m(node 2, node 3) + ... + m(node i, node j). For example, the delay of a path is the sum of the delay of every hop.

Multiplicative, if m(path) = m(node 1, node 2) * m(node 2, node 3) * ... * m(node i, node j) Example is reliability, in which case 0 < m(node i, node j) < 1.

Concave, if m(path) = min{ m(node 1, node 2), m(node 2, node 3), ... , m(node i ,node j) } Example is bandwidth, which means that the bandwidth of a path is determined by the link with the minimum available bandwidth.

Policies belonging to the same role are represented in a policy space determined by the set of metrics applicable to this role. For routing management the space has three dimensions due that there are three metrics (bandwidth, delay and losses). In order to determine the corresponding space for each policy, it is necessary to consider the rank of values that each policy has in the three metrics and the features that has each metric (additive, multiplicative, concave).

As we see in figure 8, more than one policy from the same role could be applied to the same event producing an overlapping between policies, in other words a policy conflict.

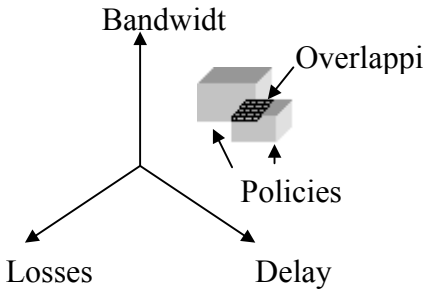

Figure 8. Metrics of the Routing Policy Role

The algorithm first analyse if there is a policy condition or a set of policies conditions that evaluates to true (using the event and inquire method), in case of finding any policy then the default policy is applied. For the specific case of routing, the default policy indicates information has to be transmitted via the routing protocols defined in each domain, for example MPLS, RSVP, ATM, etc. In our implementation, the network is based on a Differentiated Services scheme and the monitoring process uses intelligent agents to know the network characteristics. [Barba02], [Reyes02-1].

The following figure shows the management platform applied to routing functions.

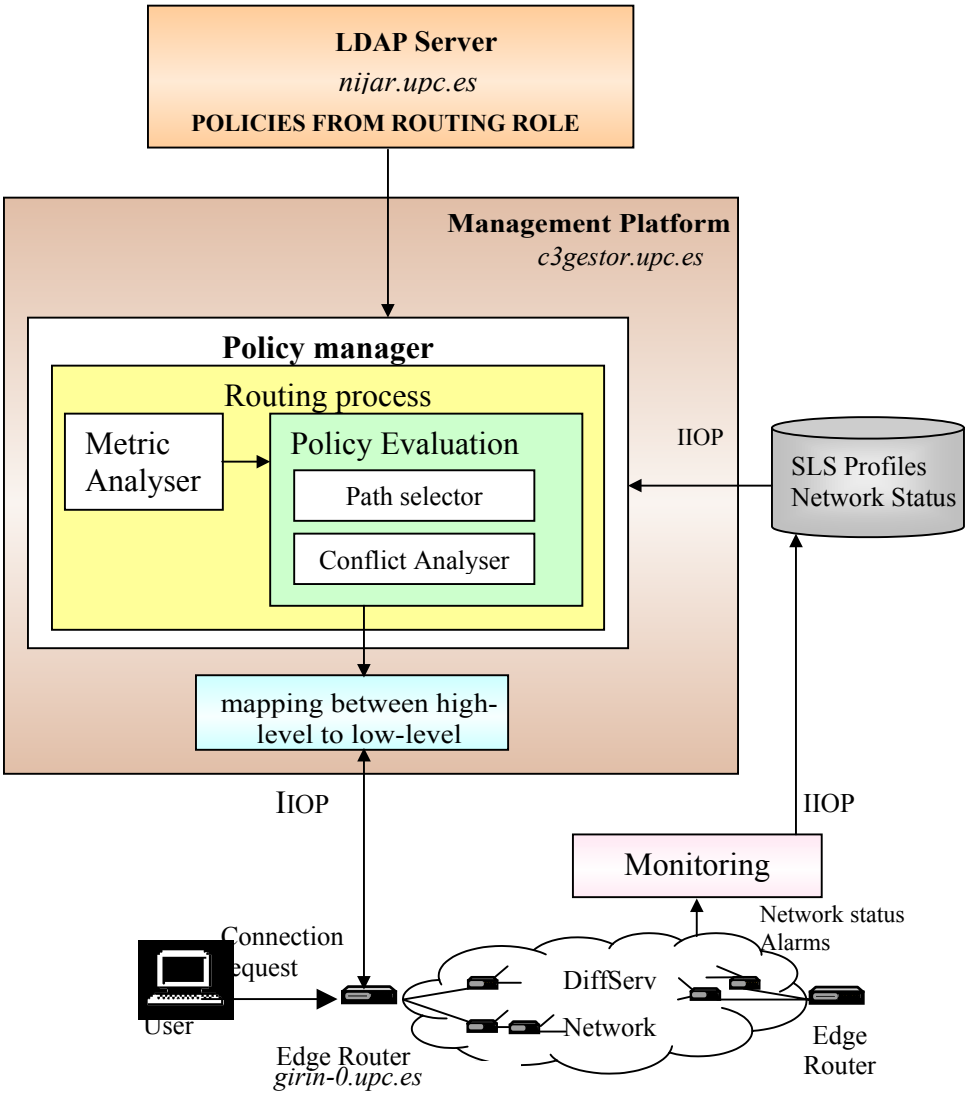

Figura 9. Management Platform to solve routing

The network administrator adds, deletes or changes policies via a policy editor, later a policy conflict resolution module solves any overlapping between the existing policies and the new ones, finally policies are stored in a LDAP directory. In great scale networks, the policy repository could have hundreds or thousands of policies.

On the other hand, the management platform sends to the path selector the parameters (source address, target address, QoS parameters). QoS parameters specify the minimum values that each link of the path has to guarantee in the three role metrics: bandwidth, delay and losses. The rank of values corresponds with the SLS profile assigned to the connection.

When the path selector receives the parameters then it looks for some policy or policies in the LDAP directory, it means, that the path selector looks for restrictions or special treatment for the connection request. Because of the default policy has the low priority then any other policy could overwrite the default instruction. Policies could be business goals, network topology restrictions, security requirements for a specific flow of traffic, etc.

Considering the fact that the role forms a hierarchic structure in which a role can consist of several sub-roles, in [Reyes02-S] we proposed that the policy selection process could carry out a first approach or screen within the corresponding role that consists of selecting those policies whose condition is true. In case there is more than a resulting policy (as it can be seen in the figure 8), then a second approach or screen could be applied to choose the policy associated to the biggest number of sub-roles.

-

Another element we use in the policy selection process is the use of a generic policy that indicates the selection criterion with higher priority for the path selection process. Some examples of selection criterion are priority for network resources, priority to the user QoS requirements, business policies, etc. All the functional areas of the system can use the generic policy. Once generic policy indicates the selection criterion, the algorithm calculates the minimum distance between the overlapping policies and the selection criterion point.

By other hand, in case any path satisfies the minimum QoS requested requirements, some policy actions can be taken: reject the connection, offer the user a minor-quality path, eliminate other connection to be able to place the new connection, readjust the bandwidth corresponding to every class of service, etc. The policy manager takes this decision using the previously defined policies by the network administrator.

Methods and heuristics used to get an adequate policy from the LDAP directory and to solve conflicts between policies should be scalable enough. The policy-based algorithm could be applied in all functional areas of the system. Only it is necessary to establish the adequate metrics for each area. For example, for the accounting management we create four metrics, so the policy space has four dimensions to represent the policies belonging to the accounting role. Next chapter shows some graphical representations.

Generalising the use of this methodology, it is possible to represent any policy from any role and sub-roles in a hyper-dimensional space.

5.3 Contribution in this chapter

This section is a contribution to the policy-based management systems specifically to the routing with QoS restrictions for intra and inter-domains.

Our architecture is based on a policy manager server, which analyses different elements such as the QoS requirements of a user connection, network status information, service priorities, business goals etc. in order to take an adequate peer-to-peer path decision for a specific traffic. The proposed methods guarantee the QoS requirements at the same time that maintain an efficient utilization level of the resources in the different network domains.

The policy-based inter-domain routing mechanism provides an efficient solution for managing increasing traffic in the Internet and for managing the business goals that each

Internet Service Provider specifies. The efficiency of this method depends on that several providers adopt it. Otherwise it will provide extra overhead to manage traffic between systems that accept policies and systems that do not.

The policy-based routing management that we propose is based on a policy architecture compatible with TMN. Its design and implementation works on a CORBA environment over a Diffserv-MPLS network

Policies are represented via an open scheme that makes easier the system scalability. The PCIM scheme [Moore01] of the IETF is used for the policy design and a repository based on directories by means of the LDAP protocol is used to store policies grouped into specific roles.

We create the concepts of generic policy, priority policy and default policy, which are used as auxiliary elements to facilitate the network management and to solve possible conflicts between policies.

Our PBMS can grow as much as necessary without any complex or expensive modifications, only adding or modifying groups of policies in the LDAP directory. In the same way, the mechanism that we follow to choose the policy or the set of policies that has to be applied in the network is scalable enough to operate with other functional areas of the system, for example failure management, accounting, etc., only it is necessary to establish the adequate metrics to define the corresponding policy role.

This research can be extended via using policies created dynamically by the network. In the work presented here static policies were only used, that is to say, policies were defined by the network administrator and later, they were stored in the LDAP directory. However, considering the creation of policies in a dynamic way, the human action could be less and less needed and the possible network instabilities could be quickly solved. Dynamic policies can be generated at real time with a prior knowledge base that indicates how to

generate rules from the network status information and from statically or dynamically previously defined policies.

References

Books

[Armitage00] G.Armitage. *Quality of Service in IP Networks. Foundations for a Multi-Service Internet*. MacMillan Technology Series. ISBN: 1-57870-189-9. USA. April 2000

[Huitema98]Christian Huitema. *IPv6: The New Internet Protocol*. Prentice Hall, 1998.

[Kilkki99] K. Kilkki. *Differentiated Services for the Internet*. MacMillan Technology Series. ISBN: 1-57870-732-5. USA. 1999

[Verna] D.Verna. *Policy-nased Networking : Architecture and Algorithms*. MacMillan Technology Series. ISBN: 1-57870-226-7. 2001

Papers

[Barba02] Barba, A. Sánchez, E. *An Architecture for Active Network Performance Management Based on Intelligent Agents*. MATA 2002. Mobile Agents for Telecommunication Applications, 4th International Workshop, MATA 2002 Barcelona, Spain. October 2002

[Chen98] Chen S., Nahrstedt K. *An overview of Quality of Service routing for next-generation high-speed networks: problems and solutions*, IEEE Network, pp. 64-79. November/December 1998

[Damianou01] Damianou N, et.al. *The Ponder Policy Specification Language.* Computer Science vol.1995 IEEE Policy 2001. Springer Bristol, U.K. 2001

[Escribano02] J. Escribano, C. García, C. Seldas, J. Moreno. *Diffserv como solución a la provisión de QoS en Internet.* II Congreso Iberoamericano de Telemática,CITA'2002

[Johari02] Johari, R., Tsitsiklis, J. *Routing and peering in a competitive Internet*. IPAM Workshop on Large Scale Communication Networks. Los Angeles. U.S.A. March 2002

[Lee95] Lee, W. Hluchyj, C., Humblet, P.A. *Routing Subject to Quality of service Constraints in Integrated Communication Networs*, IEEE Network, July/August 1995.

[Mehra00] A. Mehra, D. Verma, R. Tewari. *Policy-Based Diffserv on Internet Servers: The AIX Approach.* IEEE Internet Computing. October 2000

[Reyes02] Reyes, A., Brunner, M., Barba, A. *Controlling IP Network Management Systems via Policies.* CIIT 2002 IASTED- IEEE St Thomas, U.S.A.2002

[Reyes02-1] Angélica Reyes, Antoni Barba, Ernersto Sánchez. *Gestión Inteligente de Tráfico en redes activas*. II Congreso Iberoamericano CITA'2002. Mérida, Venezuela. Septiembre 2002.

[Reyes02-S] Angélica Reyes, Antoni Barba. *A Policy Selector for Policy-based Management Systems* SoftCOM 2002, IEEE Croatia-Italy 2002

[Sloman99] Sloman, M., Lupu E., *Conflict Analysis for Management Policies.* Integrated Network Management VI IEEE U.S.A. May 1999.

[Trimintzios01] Trimintzios P, Griffin, D. et al. *An Architectural Framework for providing QoS in IP Differentiated Services Networks* 7th IFIP/IEEE International Symposium on Integrated Network Management. Seattle, USA. May 2001

[Wang96] Wang, Z., Crowcroft, J. *Quality-of-service routing for supporting multimedia applications*, IEEE JSAC, vol14, no 7, September1996

Standards

[Bates00] T. Bates, Y. Rekhter, R. Chandra, D. Katz. Multiprotocol Extensions for BGP-4. IETF Request For Comments (RFC) 2858 June 2000.

[Blake98] S.Blake, et. al., *An Architecture for Differentiated Services*, IETF Request For Comments (RFC) 2475. December. 1998

[Cahpin92] L. Chapin Internet Architecture Board, Applicability Statement for OSPF. IETF Request For Comments (RFC) 1370. October 1992.

[Crawley98] E. Crawley, R. Nair, B. Rajagopalan, H. Sandick, *A Framework for QoS-based Routing in the Internet*, IETF Request For Comments (RFC) 2386. 1998

[Faucheur02] F. Le Faucheur, L. Wu, B. Davie, et. al. Heinanen *Multi-Protocol Label Switching (MPLS) Support of Differentiated Services.* IETF Request For Comments (RFC) 3270 May 2002.

[Marques99] P. Marques, F. Dupont. *Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing*. IETF Request For Comments (RFC) 2545. March 1999

[Moore01] Moore, E. et. al. *Policy Core Information Model. Version 1 Specification*. IETF Request For Comments (RFC) 3060. February 2001.

[Moore03] Moore, B. *Policy Core Information Model (PCIM) Extensions* IETF Request For Comments (RFC) 3460. January 2003

[Rekhter95] Y. Rekhter, T. Li. March 1771 *A Border Gateway Protocol 4 (BGP-4).* . IETF Request For Comments (RFC) 1771 1995.

[Reyes03] Reyes, A., Barba A., Moron, D., Brunner, M., Pana M. *Policy Core Extension LDAP Schema (PCELS)*, Internet Draft. February 2003.

[Rosen01] E. Rosen, A. Viswanathan, R. Callon. *Multiprotocol Label Switching Architecture.* IETF Request For Comments (RFC) 3031. January 2001.

[Strassner02] Strassner, J. *Policy Core LDAP Schema* IETF Internet-Draft. October 2002.

[Yavatkar00] Yavatkar, R. Pendarakis D. Guerin R. *A Framework for Policy-based Admission Control.* IETF Request For Comments (RFC) 2753. January 2000.

[Zinin03] Zinin, A. Lindem, D. Yeung. *Alternative Implementations of OSPF Area Border Routers*. A. IETF Request For Comments (RFC) 3509. April 2003.