www.euro6ix.net

Title:	Document Version:
Deliverable D2.1 **Specification of the Internal Network Architecture of each IX point**	4.4

Project Number:	Project Acronym:	Project Title:
IST-2001-32161	Euro6IX	European IPv6 Internet Exchanges Backbone

Contractual Delivery Date:	Actual Delivery Date:	Deliverable Type* - Security**:
30/06/2002	30/07/2002	R – PU

* Type: P - Prototype, R - Report, D - Demonstrator, O - Other
** Security Class: PU- Public, PP – Restricted to other programme participants (including the Commission), RE – Restricted to a group defined by the consortium (including the Commission), CO – Confidential, only for members of the consortium (including the Commission)

Responsible:	Organization:	Contributing WP:
Ivano Guardini	TILAB	WP2

Authors (organizations):

Cesar Olvera Morales (Consulintel), Jordi Palet Martinez (Consulintel), Alvaro Vives (Consulintel), Alain Baudot (FTRD), Carlos Parada (PTIN), Raffaele D'Albenzio (TILAB), Mario Morelli (TILAB), David Fernandez (UPM), Tomás de Miguel (UPM).

Abstract:

This deliverable summarizes the work done in the first semester of Euro6IX project inside the WP2 in the context of A2.1 activity, regarding the definition of the architecture of each Internet Exchange. The main goals of this deliverable are the analysis of the layer 2 and layer 3 infrastructure, the characterization of the route server, the individuation of the high layer services (e.g. DNS, HTTP) and of monitoring facilities to place inside each Internet Exchange point.

Keywords:

6to4, AAA, Addressing, Application Services, BGP4, DNS, DSTM, FTP, HTTP, HTTPS, ICMP, IMAP, Internet Exchanges, IX, LAN, LDAP, Monitoring, NAT-PT, NTP, POP3, RADIUS, Route Server, Routing, Routing Policy Database, Switch, SMTP, SSH, STP, Tunnel Broker, VLAN.

Revision History

The following table describes the main changes done in the document since his creation.

Revision	Date	Description	Author (Organization)
v1.0	01/07/2002	Document creation	Mario Morelli (TILAB)
v2.0	09/07/2002	Adds-on of some contributions of the partners	Mario Morelli (TILAB)
v3.0	16/07/2002	Further adds-on of some contributions.	Mario Morelli (TILAB)
v4.0	25/07/2002	Last adds-on and final revision.	Mario Morelli (TILAB)
v4.1	26/07/2002	General Review	Jordi Palet (Consulintel)
v4.2	30/07/2002	Approved by PSC and send to the PO	Jordi Palet (Consulintel)
v4.3	30/07/2002	Submitted to the EC	Jordi Palet (Consulintel)
v4.4	29/08/2002	Updated Figure that doesn't appear in PDF	Jordi Palet (Consulintel)

Executive Summary

This deliverable represents the final result of the work done in the context of the A2.1 activity inside the WP2 in the first semester of the Euro6IX project.

The A2.1 activity has as final aim the analysis, from a technical and theoretical point of view, of all the various aspects needed to individuate the optimal IX configuration. In other words, Deliverable D2.1 aims to give the technical basis needed for the design and deployment of the IPv6 Internet Exchanges inside the Euro6IX backbone.

The document is subdivided in six main sections, each of these developing a particular technical topic related to IX infrastructure.

So, after an overall overview of the IX architecture, the following sections have been developed.

First section (§3:Layer 2 Infrastructure) regards the layer 2 infrastructure and contains a set of technical considerations and proposals about the possible interconnecting scenarios at layer 2.

Second section (§4:Layer 3 Infrastructure) gives an overview of the options that can be considered for the routing and the addressing inside the Internet Exchange.

Third section (§5:IPv6 Route Server Functionality) regards the route server functionality considering the route server architecture and the relative policies to be applied for filtering routes.

Fourth section (§6:Transition Facilities) takes into account the transition facilities that can be implemented inside the IX to permit the access to the Euro6IX backbone to those users that are still IPv4 capable and for the interconnection between Euro6IX network and traditional IPv4 networks (e.g. Internet).

Fifth section (§7:IPv6 Monitoring Facilities Placed Within the IX) is focused on the monitoring facilities and its main aim is to analyze what can be monitored inside the Internet Exchange and which are the variables that can be taken into account. Moreover, in this section a proposal regarding some monitoring tools has been made.

At the end, in the final section (Chapter §8:Applications Services Provided by each IX), an analysis about more suitable application services (like DNS, WWW, RADIUS and so on) to place inside the Internet Exchange has been made.

Table of Contents

Table of Figures

1. INTRODUCTION

Euro6IX project has, as primary scope, to accelerate the introduction of IPv6 protocol in Europe. To reach this purpose, an appropriate architecture will be researched in order to design, develop, deploy and validate the first Pan-European non-commercial IPv6 Internet Exchanges Network.

The network will connect regional and strategic neutral IPv6 Internet Exchange points (in the following often named IX, for short) across Europe so to achieve higher levels of robustness and service quality than currently offered by IPv4 Internet Exchange Networks.

This project has been promoted by Euro6IX consortium consisting of competent partners from large Telecom Companies, Internet and networking manufacturers, academic and independent research institutes working together and representing a broad range of European countries (Denmark, France, Germany, Italy, Portugal, Spain, Switzerland, United Kingdom).

The Euro6IX project will give the possibility to test advanced network services and IPv6 enabled application (such as IPv6 based On-Line Education tool using multicast over IPv6, IPv6 based instant messaging application, Voice over IPv6 based application, only to mention some of most interesting and innovative) emphasizing the new features of IPv6 by involving real users in order to guarantee the scalability of both the network and the applications to prepare for a secure and reliable wide scale commercial operation in the next future.

This document, produced inside the WP2 of Euro6IX Project in the context of A2.1 activity, aims to describe a proposed architecture of the Internet Exchanges and the related supported functionality.

2. GENERAL ARCHITECTURE OF THE INTERNET EXCHANGE

The general infrastructure of EURO6IX network will consist of three different levels:

- **IX–level:** Regional native IPv6 Internet Exchanges.
- **Backbone-level**: Pan-European core network interconnecting regional Exchanges.
- **Node-level:** ISPs and other providers accessing the core network in order to provide IPv6 services and end-user access. The users will be able to connect to IX by means of different access technologies, such as mobile, cable and xDSL and they will be able to be both native IPv6 and IPv4, if no IPv6 link is available.

The backbone will be formed by six main Internet Exchanges, extended with the Portuguese IX, interconnected in a mesh topology and linked among each other by point-to-point links. All the backbone links will be IPv6 native; in other words, it is excluded the possibility of tunneling IPv6 in IPv4.

Internet Exchanges will be located in Turin (TILAB), Madrid (TID and Consulintel), linked to Portuguese IX sited in Aveiro (PTIN), Paris (France Telecom), London (BT Exact), Berlin (T-Systems), Zurich (Telscom).

In the following picture, the logical IX architecture is shown.

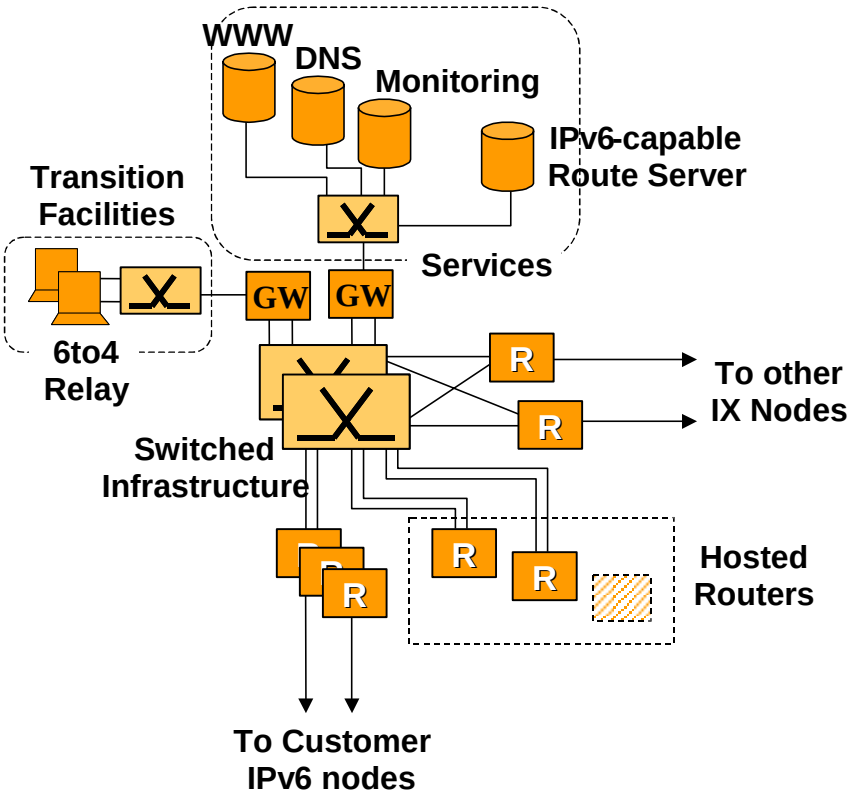

Figure 2-1: IX Architecture

As shown in Figure 2-1, five main logical blocks characterize the IX architecture:

- Layer 2 infrastructure (i.e., high performance switches).

- Layer 3 infrastructure to interconnect IX to other backbone IXs and IPv6 Customer nodes.
- Service-Transition infrastructure (including e.g. Tunnel Broker, 6to4 relay, NAT-PT).
- Service-Farm infrastructure (including e.g. WWW and SMTP servers).
- Hosted routers.

The core of IX is basically a layer 2, high performance switched infrastructure connecting the various blocks previously individuated. Inside the IX, some routers will interconnect the IXs among each other while other routers will interconnect the IX to the Customer IPv6 nodes. Moreover, the IX architecture proposed will be fully redundant to prevent failures and service unavailability.

It will also be possible to get connected to the IX through hosted routers, i.e. machines co-located inside the IX but managed by a third party (e.g. an IPv6 ISP or end-site).

Users linked to the IX will be able to access various kinds of services hosted in the IX Server Farm such as WWW or e-mail (by means of SMTP/POP based mail tools). In each IX a network monitoring service will be provided to control routing performance and routing stability inside the backbone, to monitor the status of peerings with the other IXs and to collect various flavors of reachability and traffic statistics. Inside IX there will be also a DNS server and an IPv6 capable Route Server.

As we said, both traditional IPv4 users and native IPv6 users will be able to get access to the Euro6IX backbone, since each IX will host a set of transition facilities, placed in a Transition Server Farm. This way, native IPv4 users will be able to access (e.g. through Tunnel Broker tool or 6to4 relay) innovative IPv6 services and at the same time native IPv6 users will be able to access well-known IPv4 services through e.g. NAT-PTs or Application Level Gateways.

3. LAYER 2 INFRASTRUCTURE

This chapter describes the Layer 2 (OSI Data Link Layer) architecture that should be implemented in each IX node. The Layer 2 provides basic connectivity between the equipments of an IX while allowing logical separation between distinguishable functional blocks.

3.1 General Layer 2 Architecture

Each IX will consist of a Layer 2 Switched Network providing connectivity between the main IX blocks: IX Backbone Routers, Traditional IX LAN, Transitions/Access Facilities, Services and Monitoring/Management network. Traditional IX Customers can connect to the IX using a logical separated segment in the Euro6IX IX node.

The following picture describes the general L2 main blocks.

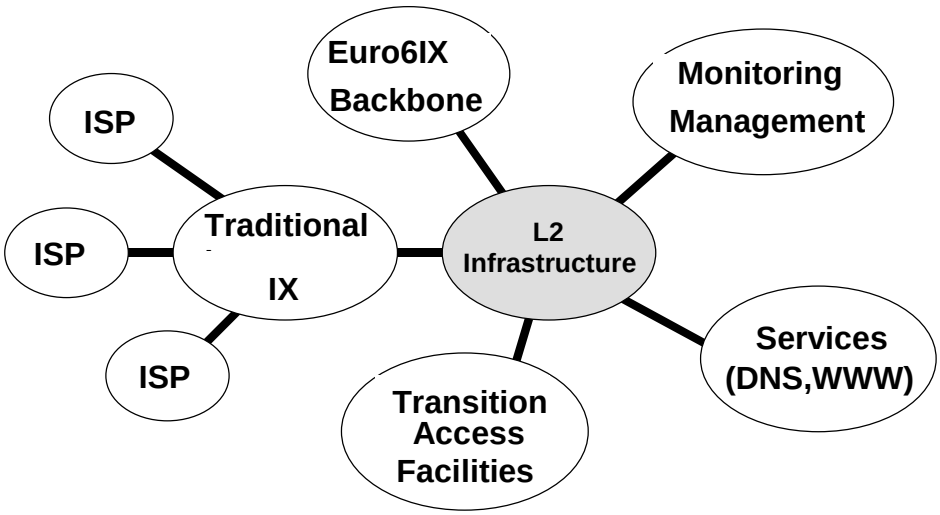

Figure 3-1: L2 Main Blocks

According to each IX node particularities, a Euro6IX node can have some or all the components defined. Anyway the basic and common components will be the Euro6IX Backbone Access and the Traditional IX or the Access Facilities Block depending on if the IX has traditional IX facilities or only allows Euro6IX backbone access.

This chapter will describe each of the IX components and the way they will connect to the L2 infrastructure.

The Switched Network can be built based on two main technologies: Ethernet or ATM. Ethernet is a broadcast, mature, cheap and well-understood technology. ATM is a point-to-point technology and holds the promise of scaling well beyond 100 Mbps to Gigabit rates, while still delivering QoS guarantees. The development of Layer 2 switching in hardware with technologies such as virtual LANs (VLANs) allows logical separation between segments and will provide increased scalability being the most recommended technology for L2 in an IX node.

3.2 Layer 2 Blocks Description

In the following, it will be provided a step-by-step description that allows the implementation of an IX node defining the technologies and the architecture for L2 connection of the main blocks previously described.

3.2.1 Basic Configuration- Layer 2 Infrastructure

Euro6IX IX is supposed to provide services collocated in the IX facilities: either Traditional IX Service or Euro6IX Access Services using either Native IPv6 Dial-Up, IPv6 over IPv4 Tunneling or using IPv6/IPv4 Transition mechanisms. Typically, dedicated equipments will provide these services so connectivity between them is needed. The easiest and cheaper way to establish local connectivity is using Ethernet technology.

Each Euro6IX IX node should consist of at least two Ethernet switches supporting local LAN segments. The IX Backbone Router will have local connectivity using two high-speed interfaces (Fast-Ethernet/Gigabit-Ethernet). For redundancy, each IX Backbone Router should have a link for each one of layer2 Ethernet switches.

Other equipments installed in the IX can also be connected directly to these switches. For equipments installed in other rooms/buildings, the local LAN is extended installing a hub/switch in each room and connecting them to the backbone switches.

The following picture shows the general connection model for the IX Local LAN.

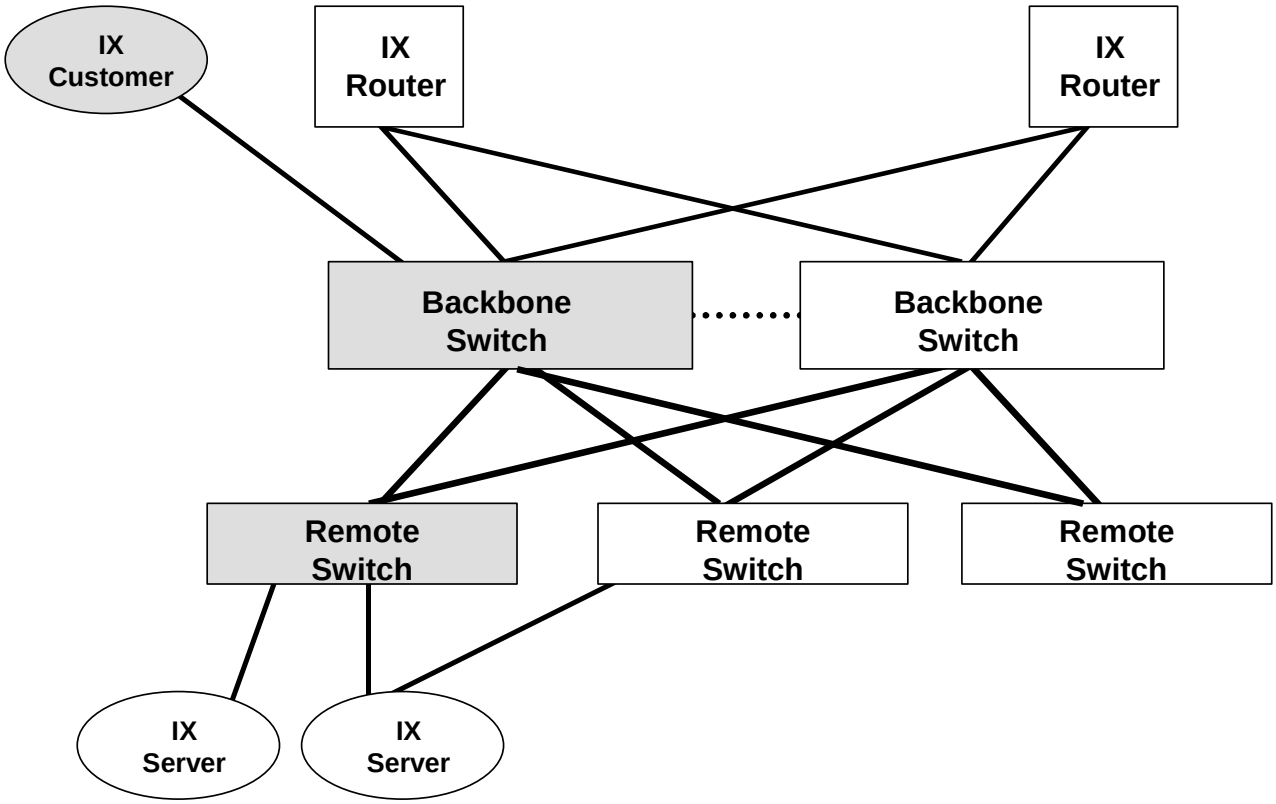

Figure 3-2: General Model for IX Local LAN

For higher availability Remote switches should have dual links to the backbone Switches. The main advantage of this design is that each remote switch maintains two equal-cost paths to every

destination network, so recovery from any link failure is fast. This design also provides double the trunking capacity into the IX switched backbone.

In the following, there are some technologies that support L2 redundancy.

3.2.1.1 Spanning Tree Protocol (STP)

The Spanning-Tree Protocol (STP) is a Layer 2 protocol designed to run on bridges and switches. The specification for STP is called 802.1d. The main purpose of STP is to ensure that we do not run into a loop situation when we have redundant paths in our network.

To provide a path redundancy and avoid a loop condition, STP defines a tree that spans all switches in an extended network. STP forces certain redundant data paths into a standby (blocked) state, while leaving others in a forwarding state. If a link in a forwarding state becomes unavailable, STP reconfigures the network and re-routes data paths by activating the appropriate standby path.

With STP, the switches in the network elect a root bridge that becomes the focal point in the network. All other decisions in the network, such as which port is blocked and which port is in the forwarding mode, are made from the perspective of this root bridge. A switched environment, which is different from that of a bridge, most likely deals with multiple VLANs. When implemented in a switching network, the root bridge is usually referred to as the root switch. Each VLAN (because it is a separate broadcast domain) must have its own root bridge. All the root bridges of the different VLANs can reside in a single switch, or they can reside in various switches.

All the switches exchange information to use in the selection of the root switch, as well as for subsequent configuration of the network. This information is carried in Bridge Protocol Data Units (BPDUs). The BPDU contains parameters that the switches use in the selection process. Each switch compares the parameters in the BPDU that they are sending to their neighbors with those one that they are receiving from their neighbors.

If the root ID (that a Switch A is advertising) is smaller than the *root ID* that its neighbor (Switch B) is advertising, Switch A's information is better. Consequently, Switch B stops advertising its *root ID*, and instead accepts that of Switch A.

According to IX L2 Topology, Backbone Switches should be root switches. These switches should have a **bridge priority** that is lower than all remote switches so that the remote switches will automatically select one of them as the root switch. Typically the bridge priority default value is 32768.

3.2.1.2 VLAN Trunking Protocol (VTP)

VTP is a Layer 2 messaging protocol that maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs on a network-wide basis. VTP minimizes miss-configurations and configuration inconsistencies that can result in a number of problems, such as duplicate VLAN names, incorrect VLAN-type specifications, and security violations.

VTP capable devices can be configurated to operate in the following three modes:

- **Server:** In VTP server mode, we can create, modify, and delete VLANs and specify other configuration parameters (such as VTP version and VTP pruning) for the entire VTP domain. VTP servers advertise their VLAN configuration to other switches in the same VTP domain and synchronize their VLAN configuration with other switches based on advertisements received over trunk links. VTP server is the default mode.
- **Client:** VTP clients behave the same way as VTP servers, but we cannot create, change, or delete VLANs on a VTP client.
- **Transparent:** VTP transparent switches do not participate in VTP. A VTP transparent switch does not advertise its VLAN configuration and does not synchronize its VLAN configuration based on received advertisements. However, in VTP version 2, transparent switches do forward VTP advertisements that they receive out their trunk ports.
  VTP pruning increases available network bandwidth by reducing unnecessary flooded traffic, such as broadcast, multicast, unknown, and flooded unicast packets. VTP pruning increases available bandwidth by restricting flooded traffic to those trunk links that the traffic must use to access the appropriate network devices. By default, VTP pruning is disabled.

3.2.1.3 VLAN Routing

Four different protocols are available for routing between VLANs. All these technologies are based on OSI Layer 2 bridge multiplexing mechanisms.

- **Inter-Switch Link Protocol:** The Inter-Switch Link (ISL) protocol is used to interconnect two VLAN-capable Ethernet, Fast Ethernet, or Gigabit Ethernet devices. The ISL protocol is a packet-tagging protocol that contains a standard Ethernet frame and the VLAN information associated with that frame. The packets on the ISL link contain a standard Ethernet, FDDI, or Token Ring frame and the VLAN information associated with that frame. ISL is currently supported only over Fast Ethernet links, but a single ISL link, or trunk, can carry different protocols from multiple VLANs.

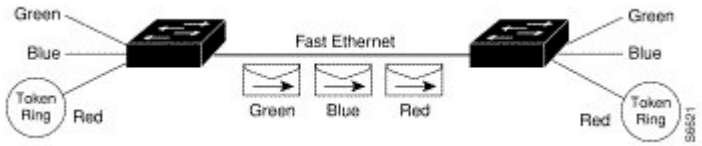

Figure 3-3: Inter-Switch Protocol

- **IEEE 802.10 Protocol:** The IEEE 802.10 protocol provides connectivity between VLANs. Originally developed to address the growing need for security within shared LAN/MAN environments, it incorporates authentication and encryption techniques to ensure data confidentiality and integrity throughout the network. Additionally, by functioning at Layer 2, it is well suited to high-throughput, low-latency switching environments. The IEEE 802.10 protocol can run over any LAN or HDLC serial interface.
- **IEEE 802.1Q Protocol:** The IEEE 802.1Q protocol is used to interconnect multiple switches and routers, and for defining VLAN topologies.
- **Layer 3 Routing:** If L2 have routing capacity they can route traffic across VLANs. In this case IP addressing in each VLAN cannot be overlapped

3.2.2 Basic Configuration-Traditional IPv6 IX Customers

A simple Euro6IX IX will work as traditional IXs do, allowing ISP customers to install a router in IX facilities, establishing a local interface to the IX LAN segment and allowing peering traffic with other ISPs.

Traditional customers are those who use the IX in the traditional way (i.e. just to set-up their own peering with the participating ISPs). Next generation customers are those who use the Layer 3 mediation function of the IPv6 IX.

Customers can have one or multiple links for redundancy. Each IX should establish rules for establishing connectivity defining user interfaces allowed: Half-duplex 10BaseT, Full-duplex 100BaseTX, Full-duplex 1000BaseSX, etc.

One possibility to establish connectivity between ISPs that could be tested during the Euro6IX project lifetime is the usage of VLANs to control the connectivity between customers.

3.2.3 Basic Configuration-Connection between Regional IXs

Some IX could want to interconnect existent Regional IXs to the Euro6IX IX. This can be done establishing an Ethernet extension of the local bridge has been described to the Remote switches.

3.2.4 IEEE 802.1P – QoS Prioritization

The 802.1P is applied both for the *Ethernet* (IEEE 802.3) and the *Fast Ethernet* (IEEE 802.3u) and the *Gigabit Ethernet* (IEEE 802.3ab). 802.1P traffic is simply classified and sent to the destination; no bandwidth reservations are established. 802.1P is a spin-off of the 802.1Q (VLANs) standard. The 802.1Q standard specifies a tag to be appended to the MAC frame. The VLAN tag carries VLAN information.

Priority	Traffic Type
1	Background
2	Spare
0 (default)	Best Effort
3	Excellent Effort
4	Controlled Load
5	Video
6	Voice
7	Network Control

Figure 3-4: IEEE 802.1P Priority Levels Table

The VLAN tag has two parts: The VLAN ID (12-bit) and Prioritization (3-bit). The Prioritization field was never defined in the VLAN standard. The 802.1P implementation defines this prioritization field. 802.1P establishes eight levels of priority similar to IP Precedence (see Figure 3-4). Network adapters and switches route traffic based on the priority level. Using Layer 3 switches allows you to map 802.1P Prioritization field to IP Precedence field before forwarding to the routers.

3.2.5 Basic Configuration-External Connectivity

The switched Infrastructure technology is dependent of the services that each IX is willing to provide in particularly if the Euro6IX IXs node will switch traffic as normally traditional IX do (using a common shared medium where ISP establish peering sessions to flow traffic between them).

A common building block to all the Euro6IX IXs is the international connectivity to other Euro6IX nodes. In the Euro6IX project, Internet exchanges will be interconnected with each other through dedicated trunks. These dedicated trunks terminate on an IX and can be: Dedicated lines (E3, STM1/OC3) or logical connections provided by a public ATM network.

Each IX will also provide long-haul connectivity to regional customers by creating logical connections between customer's equipments and the IX Router. For flexibility and scaling reasons this connections will normally use ATM or Frame-Relay technology.

So, normally IX external connectivity will be established using ATM or Frame-Relay connections passing though to a commercial ATM/FR network, exception being made for links to other IXs.

The following picture represents a basic model for interconnection with IX external entities.

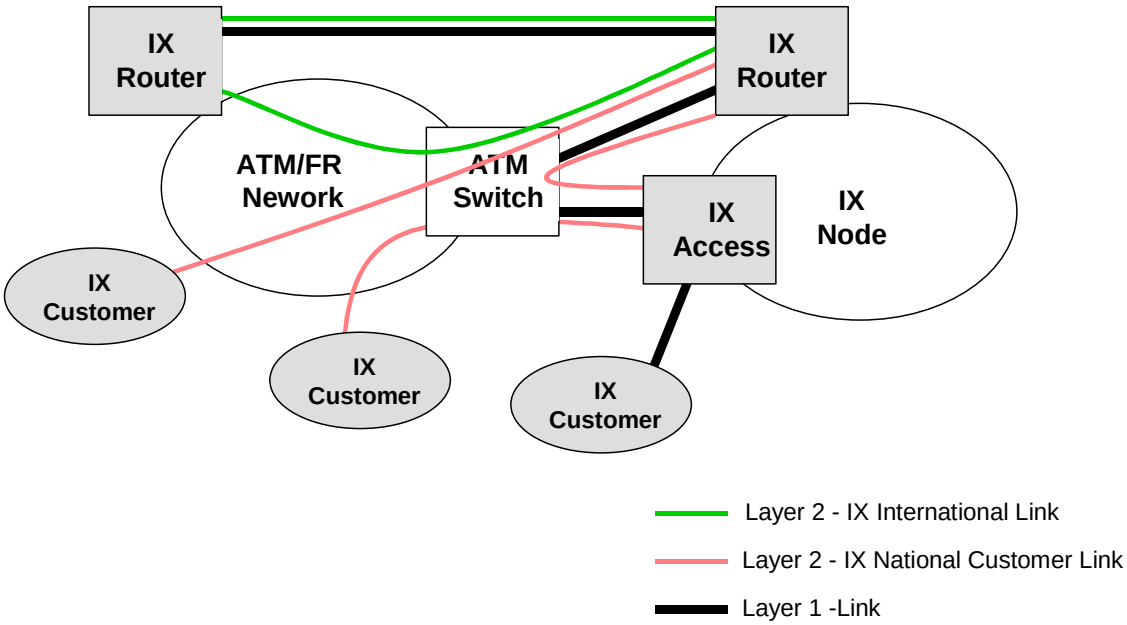

Figure 3-5: External Connections

Basically IX equipments installed in the IX can have connections using: Physical direct links either for the international link to other IXs or for leased lines access if the IX provides Access Services; logical links passing though a ATM Network. Typically, IX equipments will connect to an L2 switch port of the ATM Network Switches.

An IX node could also have a local ATM Switch (L2 Switch) allowing ATM local connectivity between equipments installed in the IX. ATM/FR public Networks will policy traffic sent in each logical connection to assure that the contracted traffic parameters are respected. So to establish

multiple logical connections inside a single logical connection created on the ATM Network (VP Tunneling) is not a possibility.

This architecture could be used for an IX providing long-haul connectivity.

3.2.6 Basic Configuration-International Connectivity

Euro6IX IX can also function as NAPs (Network Access Points) to an IPv6 international network. In this case Customers will peer with a long-haul IPv6 Operator that could also be connected to the IX node. In this case peering agreements are more complex and long-haul providers could: Either charge for international connectivity based on measurement of the total traffic transferred per Operator or to limit the total bandwidth provided per Operator.

The long-haul router should do bandwidth limitation by defining: Layer-3 traffic shaping on the egress of the long-haul provider router interface; ATM policing through an ATM switch.

When a long-haul provider offers international connectivity through the local Euro6IX LAN, the following Layer 2 configuration applies:

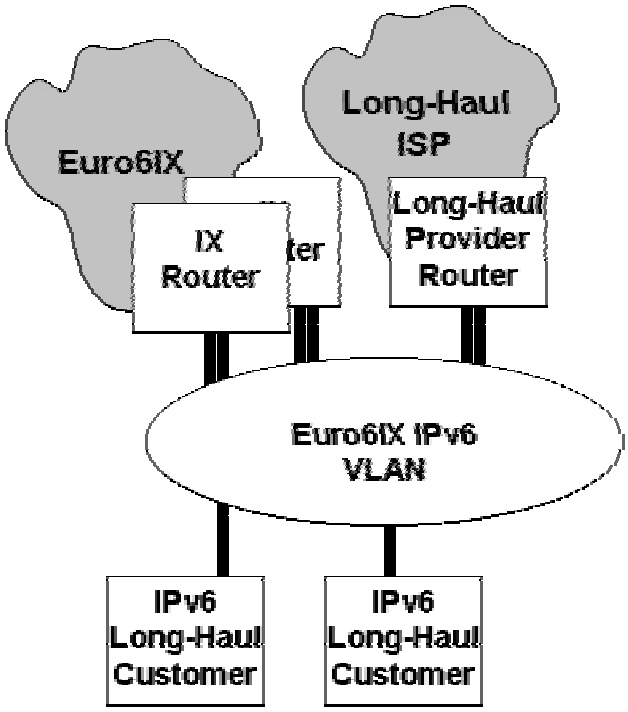

Figure 3-6: International Connectivity with VLAN

When a long-haul provider offers international connectivity though an ATM connection the following configuration applies:

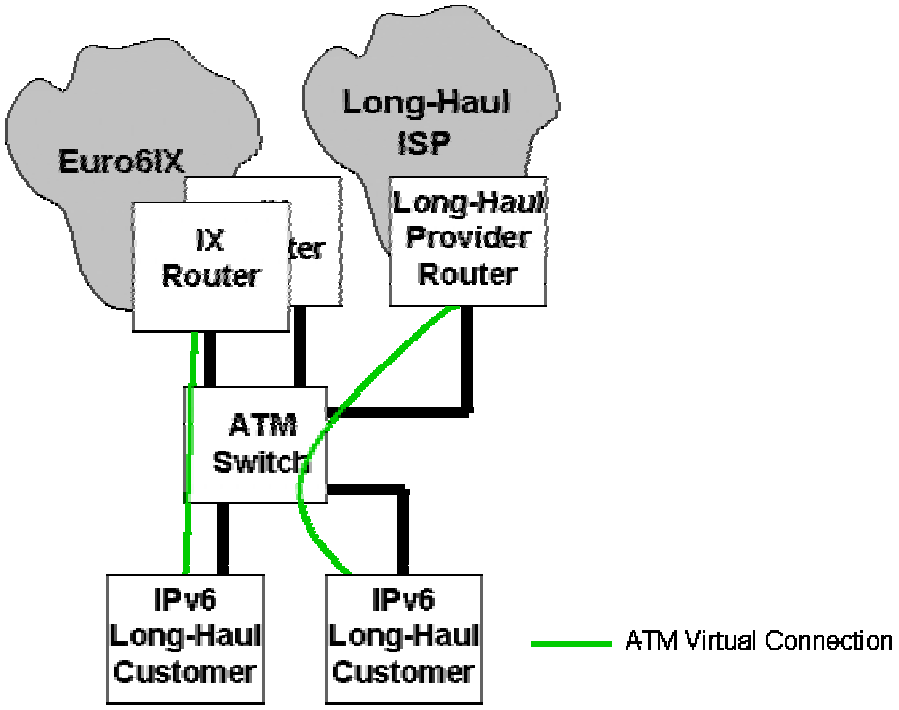

Figure 3-7: International Connectivity with ATM

The use of ATM technology for customer-connections provides guaranteed, adjustable, dedicated bandwidth for exchanges of traffic among interconnected networks.

Thanks to ATM PVC operating features this connections can provide:

- Inter-peer interconnection throughput guarantee (configurable speed).
- Impenetrability of inter-peer flows enabling control of throughputs allocated.
- Differentiated Qualities of Service (QoS), favoring purchase or sale of connectivity among providers with quality of service configurable per PVC.
- Granularity and flexibility for each ATM PVC bandwidth.
- Scalability by simple reconfiguration of PVCs corresponding to inter-peer throughputs.

ATM interconnection is implemented by setting up ATM PVCs between interconnection routers. IX managers would assign a PVC number to customers for their interconnections with long-haul provider router. IX customers that want to have ATM international connectivity must negotiate the ATM PVC with de the long-haul provider. This PVC will be dedicated to peering or transit traffic between the two ISPs.

The class of ATM service that is available on the switch should be: Variable Bit Rate Non Real Time (VBR-NRT). The VBR-nrt class corresponds best to support IP data flows from the Internet by offering a guarantee on an average cell rate. The switch should have a UPC (User Parameter Control) that can be activated by VC. This control operates PCR (Peak Cell Rate), SCR (Sustainable Cell Rate) and MBS (Maximum Burst Size) parameters and uses up to two levels of GCRA (leaky bucket) algorithms. For VBR-nrt traffic, a first level of the GCRA algorithm will police on the PCR(0) and a second level on the SCR(0) while marking outsize cells. Some rules should be defined like the PCRs (Peak Cell Rate) and SCRs (Sustainable Cell Rate) may have values of 1Mbps to 34Mbps or 155 Mbps by increments of 2Mbps. The SCR is ½ of the PCR selected.

3.2.7 Basic Configuration – Next Generation Customers

Euro6IX IX nodes are supposed to provide a more flexible architecture to connect customers to the IX. According to this, one IX node could have an L3-mediation router that receives customers' traffic and forwards it to the local ISPs and to long-haul providers. This router must be attached to the IX LAN. The following picture represents this router in the IX architecture.

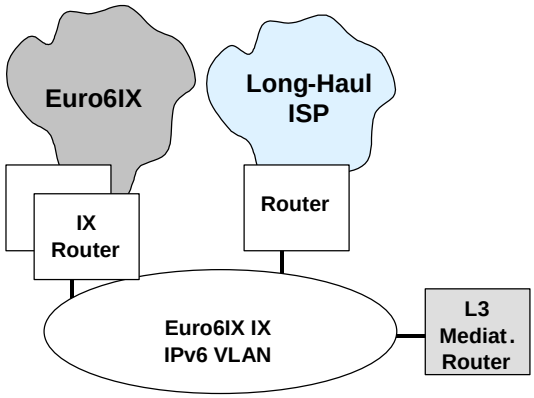

Figure 3-8: Layer 3 Mediation Router

The connection to the long-haul provider could use the local LAN or a Remote connection.

3.2.8 Basic Configuration-Management Network

The basic IX will also have a LAN segment where to connect Network Management Systems (Monitoring/Management). Access to these systems will be protected by a firewall. The rules applied on the firewall should be defined according to each IX implementation. Firewall could also control the access to Services Area (DNS/WWW). The following picture represents the basic model for implementation of the Management Network.

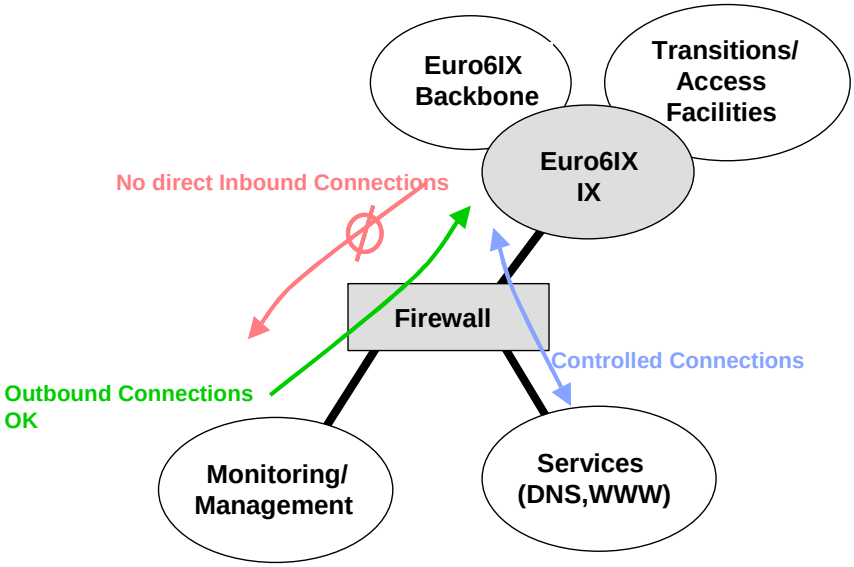

Figure 3-9: Management Network with Single Firewall

The firewall filtering should be implemented using dedicated equipments. In more critical IX firewalls, redundancy should be applied and in this case should be implemented based on two distinct boxes.

Typically each firewall has three Ethernet interfaces connected to Management LAN, IX Backbone LAN, and to the other firewall for synchronization. Traffic directed to the firewall uses virtual interfaces that represent the whole firewall system, regardless of which box forwards the packets. There should be two virtual interfaces, one for the Management LAN and the other for IX LAN.

In a dual-firewall environment the typical configuration should be that one described in the following scenario.

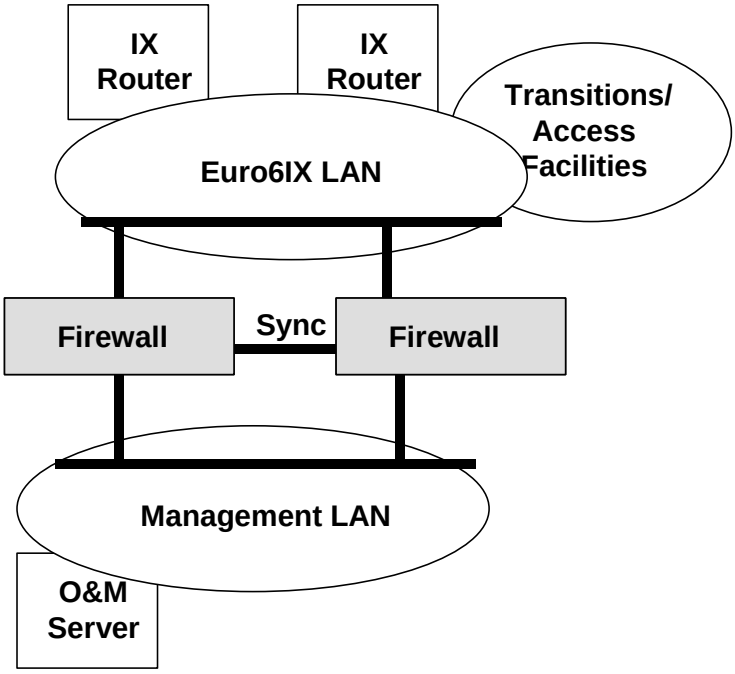

Figure 3-10: Management Network with Dual Firewall

Preferably Management LAN should be implemented in a separated VLAN secured by the firewall. In simple IX the firewall could be implemented using a router and defining simple ACL (Access Control Lists).

3.2.9 Advanced Configuration-Transition Facilities

A more complex IX will have mechanisms to translate packets from IPv4 to IPv6. These mechanisms will need fast access to the public IPv6 network and, as they route customer traffic no advanced security mechanisms are needed. The best way to connect these equipments is to direct connect them to the Euro6IX default VLAN allowing them to send/receive traffic from IPv6 customers. Layer 3 routing mechanisms should be applied in order to route traffic to these equipments.

Traditional IPv4 IX could evolve to IPv6 IX. To separate IPv6 and IPv4 traffic a special VLAN could be created in order to transport IPv4 traffic and a bridge between IPv6 and IPv4 world would route traffic among the VLANs using different layer-2 logic interfaces.

The following picture represents the architecture for IPv4 and IPv6 traffic separation in the IX.

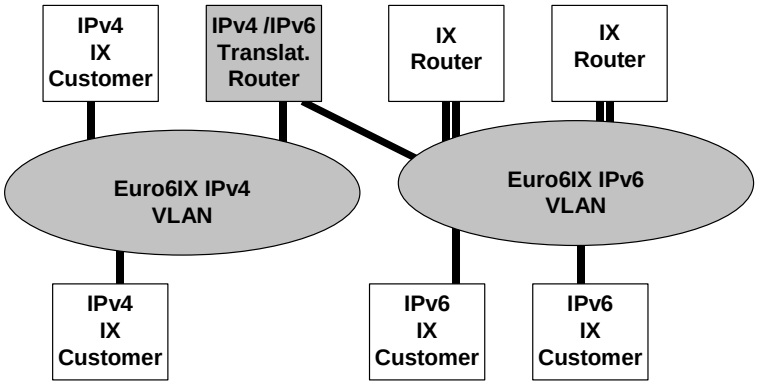

Figure 3-11: IPv4/IPv6 Transition

3.2.10 Advanced Configuration-Remote Access Customers

The IX will also provide remote access to customer either connected though leased lines or dial-up. The customers access will typically be done either by a direct connection to an IX backbone Router or though an Access router.

Access Routers will belong to IX manager and should have connectivity to the IX long-haul Backbone.

The following picture describes the standard architecture for connectivity between IX equipments.

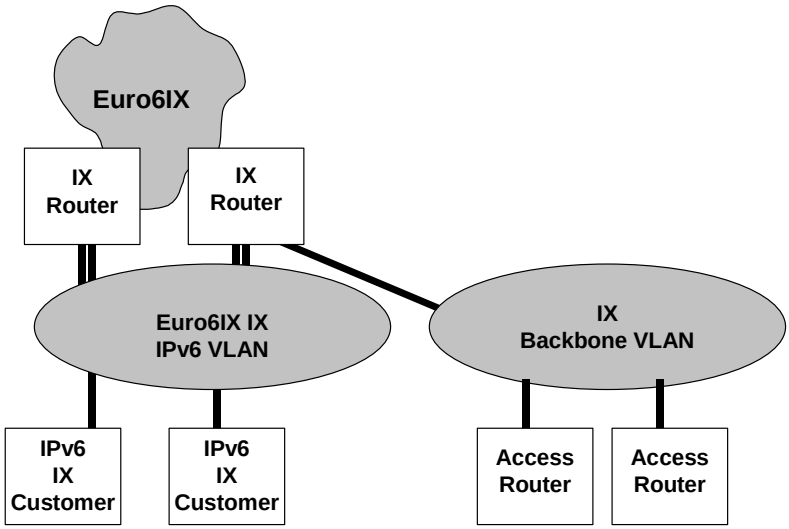

Figure 3-12: Inter-IX Connectivity

3.2.11 Advanced Configuration-Servers Farm

The IX will also support servers offering either Basic Internet Services like (DNS, IRC, NTP) or Content Delivery Services (WWW, FTP). These equipments will be connected to a specific network segment that should have high-speed links to the backbone segment and secured access from IPv6 customers.

The following picture represents the generic diagram for connecting a server farm to the IX backbone segment.

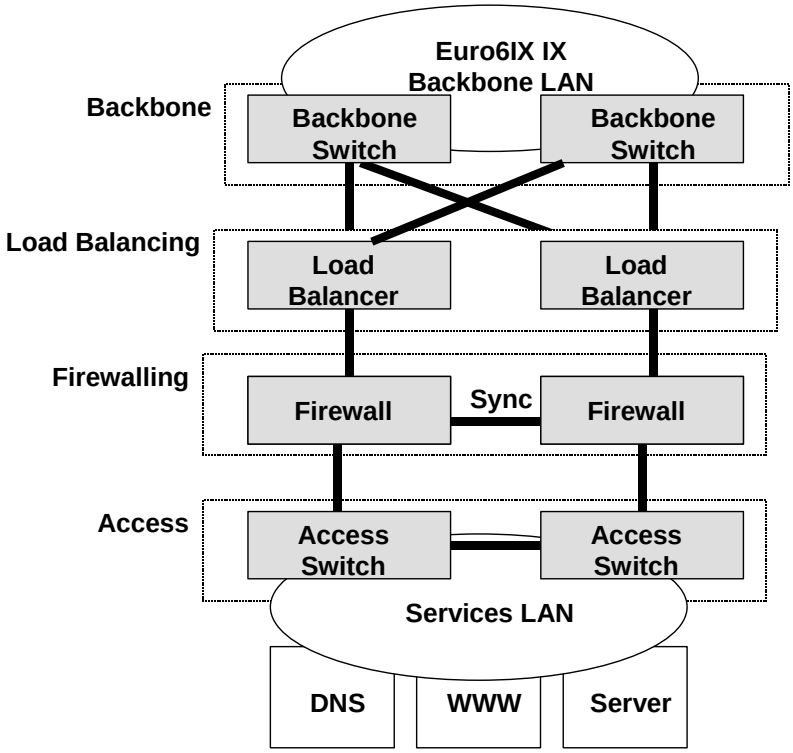

Figure 3-13: Server Farm

3.3 Redundancy, Replication and Scalability

In order to increase the bandwidth available between two switches of the Layer-2 infrastructure the Ethernet Channel mechanism should be applied.

With this mechanism, two physical links are established between the switches but they work as a single link (trunk). Moreover it supports the load balancing across the bundle.

The following picture represents the usage of Ethernet Channel for establish a trunk between the IX backbone switches.

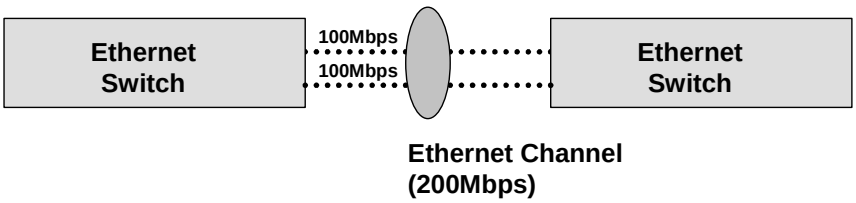

Figure 3-14: Ethernet Channel

3.3.1 Traffic Throughput within the Exchange

In the implementation of an IX node customers should make some considerations related to the expected traffic to define the necessary and best design for the IX.

4. LAYER 3 INFRASTRUCTURE

In this section some general guidelines about routing and addressing inside the IX are provided and two different scenarios will be shown. These scenarios refer to the different architectural models considered for two different phases in the evolution of the IX model. In any case, topic related to this section is currently under discussion among the partners.

4.1 Routing

4.1.1 First Phase Scenario

The model adoptable in the first scenario would be very similar to the classical conception of the IX where the providers' routers are directly connected to the second layer infrastructure. In this scenario, the router linking the IXs between each other belongs to the domain administratively managed by the Telco owning that IX. So an intra-domain routing protocol could be run on this router to exchange reachability information with those IX routers belonging to the same administrative domain.

4.1.2 Second Phase Scenario

Second phase scenario considers on a new conception of the IX based on the so-called "layer 3 mediation function". This new role is based on the possibility, from the IX to assign IPv6 prefixes independent of the provider. In this case, each customer accessing the IX chooses the provider (one or more) and the IX assignee them the IPv6 prefixes. If a customer decides to change the provider, it does not have to change IPv6 prefixes, because they are provider independent and are assigned by that particular IX. In this scenario, the router linking the IXs between each other can belong to a different administrative domain than that managed by Telco owning the IX. So an inter-domain routing protocol could run on this router to exchange routing information with the IX routers belonging to other administrative domains.

4.2 Addressing

4.2.1 First Phase Scenario

In this case, for our scopes, a /48 IPv6 prefix could be enough to number the IX devices. The Telco owning the IX will manage the pTLA or sTLA to number the IX devices.

4.2.2 Second Phase Scenario

Also in this case a /48 could be enough. In this scenario, the IX is the owner of the sTLA or pTLA used to assign prefixes to Next Generation Customers.

4.3 Layer 3 Redundancy Mechanisms

In this section a brief overview about two redundancy protocols is given. These mechanisms are implemented on the routers, but it is important that Layer infrastructure 2 is configured to support them.

4.3.1 Layer 3 VLAN Redundancy: HSRP

The Hot Standby Router Protocol (HSRP) is a Cisco innovation, which provides excellent fault tolerance and enhanced routing performance for IP networks. HSRP allows Cisco routers to monitor each other's operational status and very quickly assume packet-forwarding responsibility if the current forwarder in the HSRP group fails or goes down for maintenance. This mechanism remains transparent to the attached hosts and can be deployed on any LAN type. With Multi-Group Hot Standby, routers can simultaneously provide redundant backup and perform load sharing across different IP subnets.

4.3.2 Layer 3 VLAN Redundancy: VRRP

The Virtual Router Redundancy Protocol (VRRP) is a standard protocol **[1]** that provides a function very similar to that one provided by the Cisco proprietary protocol HSRP. A group of routers individuate the so-called VRRP group where a router is elected the “Master virtual router” and the others are the “Back-up virtual routers”. These routers are seen as a single virtual router that is configured as default gateway for the clients in a LAN. If the Master virtual router goes down, then the back-up virtual router becomes the Master virtual router (according to a VRRP priority number) forwarding the packets in place of the Master virtual router gone down.

5. IPv6 Route Server Functionality

This section describes the Route Server functionality (a modification of the peering model used in Internet Exchange points that improves scalability when the number of peering partners is high) and its application inside the Euro6IX project. Besides, it briefly describes Routing Policy Registries and other useful tools associated with them, which are often used in conjunction with Route Servers.

5.1 Introduction

The interchange of routing information between ISPs connected to an IPv4 or IPv6 IX is made by means of the BGP4 protocol **[1] [3]**. Typically, when two ISPs reach an agreement to exchange traffic through an IX, they configure and establish a BGP session between their border routers. Through this BGP session, each router will export its owns routes (i.e., prefixes that are accessible inside its Autonomous System) and other routes learned from other ASs (for example, routes learned from peering sessions in other IXs or routes from client networks).

Routes received from a peering neighbor are always filtered for several reasons. First of all, because it is natural in inter-domain routing to filter routes in order to comply with routing policies defined for the AS. Only the compliant routes are redistributed, first to internal BGP neighbors that later will redistribute them to other ASs through peering sessions in other IXs.

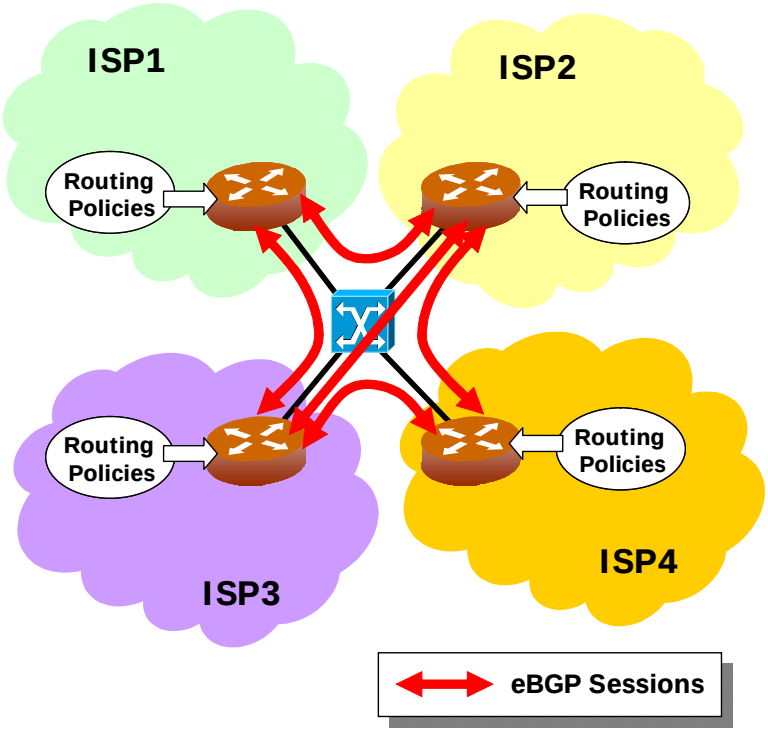

Figure 5-1: Peering Without a Route Server

But also filtering is made for security and stability reasons, to avoid, for example, instabilities due to configuration errors in border routers. It has been the case in the past that errors in the list of prefixes to export in a border router have taken down an important part of the routing system for some hours.

Figure 5-1shows an IX with four ISPs connected to it. In this simple example, all the ISPs participating in the IX establish peering sessions between them. Only ISP1 and ISP4 do not peer on the IX for any reason (commercial, technical, etc), as it happens in real IXs.

Maintaining an almost complete mesh of peering sessions in an IX could be difficult to manage if the number of participants is high (some present IXs host more than 50 participants). Each router has to maintain a high number of peering sessions and configure a different set of filters for each session. Besides, when a new participant comes to the IX, almost all routers would have to modify its configuration to peer with the new participant. For all of these reasons, the scalability of IXs based on full mesh peering sessions is seriously compromised.

The solution to these scalability problems came as a result of the Routing Arbiter Project **[4]**, started in 1993 as part of the initiatives to design the post-NSFNET Internet in USA around the Network Access Points (NAPs). This project gave rise to the Route Server concept, described in detail in the next section.

5.2 Route Server Architecture

In order to solve the scalability problems, the Router Arbiter Project proposed the creation of a new system, named the **Route Server (RS)**, which centralized the interchange of routes between participants in an IX. Instead of peering among each other, providers peer only with the RS, as shown in Figure 5-2. The RS maintains a complex database with all the information needed for providers to set their routing policies.

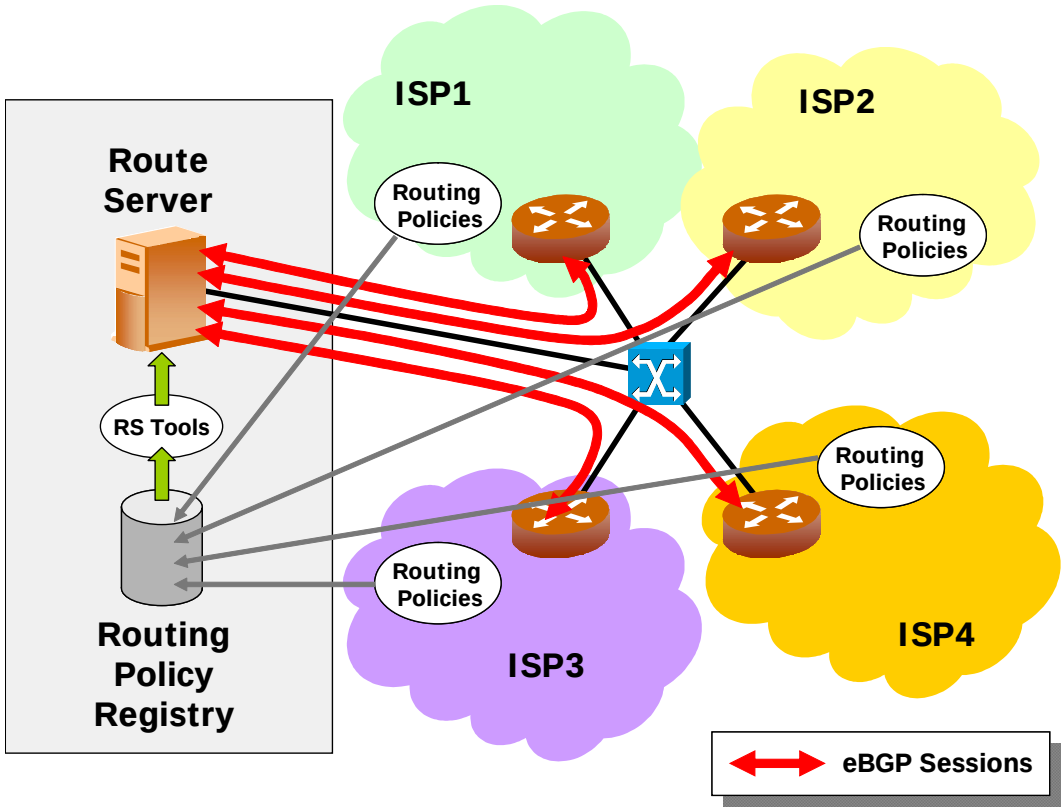

Figure 5-2: Peering Through a Route Server

As stated in **[5]**, "the Route Server facilitates routing exchange among IX-attached ISPs by gathering routing information from ISPs routers, processing the information based on the ISP's routing policy requirements, and passing the processed routing information to each ISP router.

The RS uses the BGP-4 inter-domain routing protocol to exchange routing information with each ISP router".

"The Route Server does not forward packets among the ISP routers attached to a connection point. Instead, it uses BGP's third-party routing information capabilities to pass routing information from one ISP to another, with the next hop pointing to the ISP router that advertises the route to the RS. Traffic is therefore exchanged directly among the ISP routers on the NAP, even though the routing information is provided by the Route Server".

Basically a Route Server is a modified BGP4 daemon capable of maintaining several copies of the Routing Information Base (RIB), commonly named *Views*, one for each participant ISP. Instead of directly applying its routing policies, ISPs store them in a common database named *Routing Policy Registry*, using a *Routing Policy Description Language*. The policies are processed using some tools (*RS Tools* in the figure) and stored in the RS where they are used to filter routing updates and create particular "views" for each ISP with exactly the same content as if the ISP would have peered directly with other providers. For instance, the RS could give a different path towards a given destination to different ISPs, in case those paths were available and the policy requirements of the different ISPs stated it.

The next subsections describe with more detail the main components of a Route Server.

5.2.1 Route Server

They are typically UNIX based workstations running modified versions of BGP daemons able to maintain multiple routing databases (views), one for each ISP participating in the IX. Contrary to normal BGP implementations, RSs do not install the learned routes into its own routing tables; they just act as intermediaries between participants. They usually require an important amount of memory, although they are not critic in other aspects, as they do not switch data traffic, which goes directly through L2 IX switches.

Route Servers typically have their own AS number and can be configured for each peering session to insert or not its own number on the AS_PATH attribute of routes exchanged. In the former case, the Route Server is said to work in transparent mode and, as the AS number is not propagated, identifiers from the AS numbers private range (64512 to 65535) can be used.

As the Route Server becomes a critic point of failure in the whole routing systems, it is typically possible group two or more Route Servers in a cluster to provide fault tolerance.

5.2.2 Routing Policy Specification Languages

Routing Policy Specification languages have been designed to express and store routing policies in routing policy databases, apart from storing general information about ASs in order to facilitate the diagnostic and solution of global routing problems.

The language widely used for this purpose was originally expressed in RIPE-181 specification **[6]**, released in 1994. Together with other RIPE supporting documents it provided details about the objects and attributes to be used within a Routing Policy Database.

Later, a new language, named Routing Policy Specification Language (RPSL), was developed by the IETF Routing Policy System Working Group. RPSL is able to express a wider range of routing policies that makes it more adequate for router policy programming.

RPSL is defined in **[7]** and specifies the object-oriented data that contain pieces of policy and administrative information in attribute-value pairs. RPSL allows a network operator to specify routing policies at various levels in the Internet routing system such as those policies affecting Autonomous System articulation and router configuration. RPSL is designed so that router configurations can be generated from the description of the policy for one AS combined with the description of a router.

RPSL, together with some additions and improvements proposed by RIPE, is the language used in most of the public routing policy registries in Internet, like RIPE, ARIN or APNIC.

At present, RPSL does not support IPv6. Some extensions have been proposed in recent years in order to store IPv6 information in routing databases to facilitate the coordination of the 6BONE network. That is the case, for example, of "inetnum6" or "ipv6site" object types, defined to store IPv6 addresses and information about IPv6 sites respectively, and implemented in 6BONE routing registry **[8]**.

However, deeper modifications to RPSL are needed in order to use it for describing IPv6 routing policies. A new version of RPSL, named RPSLng, is at present being discussed and defined in the context of IETF and RIPE Working Groups. Although some drafts have already been published **[9]**, there is not yet general agreement on this subject.

5.2.3 Routing Policy Databases

They are just the databases where the routing policies of ASs are stored. They can be queried using different tools, being the most commonly used the "whois" application. There are at present more than 50 around the Internet operated by organizations such as Verio, Cable and Wireless, and Merit. All together form what has been named "The Internet Routing Registry" (IRR) **[10]**, a public repository of announced routes and routing policy in a common format that ISPs use to configure their backbone routers, analyze routing policy, and build tools to help in the effort of coordinate inter-domain routing.

Merit Network Inc. operates since 1995 the main routing policy database in the Internet, named RADB **[11]**. RABD was the database set up for the Routing Arbiter Project. Nowadays it stores routing policies from an important number of providers, but its main mission is "to mirror all component databases so as to provide the most complete view of the entire IRR". Currently, querying the Merit whois server, you can retrieve information from more that 34 IRR databases.

5.2.4 Route Server Tools

The operation of a Route Server needs to be supported by a set of tools that help in tasks like the automatic generation of router configurations starting from databases contents, the analysis of routing policies or the diagnostic of global routing problems.

An initial set of tools were designed and developed inside the Router Arbiter Project named the RAToolSet **[12]**. They are basically the standard set of tools used worldwide. Later, as USC Information Sciences Institute (the initial developer of RAToolSet) discontinued its effort on them, a new project was created by RIPE, the Internet Routing Registry Toolset Project, in order to continue its development and support. The set of tools has also been renamed to IRRToolSet **[13]**.

5.2.5 Other Route Server Related Tools

There are several tools available on Internet to help in the management of global routing. We just mention here some of them that have relation with the Route Server functionality:

- **Looking Glass.** Basically it is a router located in an IX or inside the network of a provider that is publicly accessible (normally via telnet or a web page) to provide a view of inter-domain routing from that point of view. Looking Glasses are very useful tools to know how other providers around the network see the prefixes exported from an AS. Besides, they typically allow the use of tools like ping or traceroute from them for testing, diagnostic and problem solving purposes. See **[14]** for a good list of Looking Glasses available in Internet for IPv4 or **[15] [16]** for IPv6.
- **Route Viewer[17].** It is a tool developed inside the University of Oregon Route Views Project to obtain real-time information about the global routing system from the perspectives of several different backbones and locations around the Internet. It is similar to a "looking glass" tool, in the sense that provides information about the status of inter-domain routing. But the main difference is that a "Route Viewer" maintains connections (BGP neighborhoods) to several different and distant points in Internet. Route Viewer uses an AS number to establish peering sessions with multiple neighbors around the Internet to receive and store routes from them, but the routes received from neighbors are never passed on nor used to forward traffic. Besides, the route viewer itself does not announce any prefixes. It is a passive tool.
- **Route Server[1].** Some Route Servers are publicly accessible through telnet in order to consult information related with BGP global routing like neighbors, routes, damped routes, etc. A good list of Route Servers available can be found in **[17]**.

5.3 Route Server Implementations

This section lists some implementations of route server functionalities and associated tools.

- Route Servers:
  - **The Route Server Daemon (RSd) [18].** Developed by ISI for the Routing Arbitrer Project, RSd is a routing daemon that implements route server functionality. It runs on several different UNIX variants and is derived from GateD. It does not support IPv6; only IPv4.
  - **GNU ZEBRA [19].** It is free routing software distributed under GNU General Public License that includes implementations of the most used routing protocols in the Internet (RIP, OSPF, BGP, etc). It supports IPv6 and Route Server functionality and its being regularly updated and maintained.
  - **MRT [20].** It is free multi-protocol IPv4/IPv6 routing daemon developed by Merit that includes routing analysis and simulation tools. It supports Route Server functionality for IPv6, but it is not being neither updated nor maintained.
  - **GATED [21].** Next Hop Technologies provides a commercial Route Server implementation for IPv4 and IPv6 derived from the well-known "gated" routing daemon. Free Licenses are provided for academic and research purposes.

[1] *Sometimes the term route server is misused. Some of the systems named route servers in Internet are basically looking glass located in IXs that do peers with all the participants and collects route advertisements from them, but do not reflect those routes to the other peers. They just collect them for debugging, diagnostic or research purpose.*

- Policy Databases:
    - **IRRd [22].** It is a freely available, stand-alone Internet Routing Registry database server. IRRd supports the RPSL routing registry syntax. The IRRd package includes all required IRR support services, including: automated near real-time mirroring of other IRR databases, update syntax checking, authentication/security, and notification (Merit).
    - **RIPE Database[23].** It is the implementation of the database used to support RIPE whois server. It is written in perl and supports RPSL and 6BONE IPv6 extensions.
- Route Server Tools:
    - **IRRToolSet [13].** It is a suite of policy analysis tools to operate with routing policies written in RPSL format and registered in the Internet Routing Registry (IRR). The main goal of the tool suite is to make routing information more convenient and useful for network engineers, by providing tools for automated router configuration, routing policies analysis, and maintenance. For example:
        - RtConfig, that analyzes the routing policies registered in the Internet Routing Registry (IRR) and produces router configuration files.
        - Prtraceroute, that prints the route and policy information packets take to a network host.
        - Prpath, that enumerates a list of paths between Autonomous System and specified destination.

5.4 Recommendations for Euro6IX

Having into account the importance of the route server functionality and being Euro6IX a project centered on IPv6 IXs, it is important to study, test and deploy route servers, at least on some of the IXs, in order to gain experience in the management of global routing IPv6 scenarios. However, the lack of maturity of Route Server standards and implementations does not permit at this moment to make a detailed architecture proposal for the implementation of this functionality at Euro6IX interconnection points.

Instead of that, we propose the following actions (some of them already started):

- Study and experiment with route server implementations available in private scenarios.
    - Evaluate IPv6 route server implementations and tools.
    - Evaluate routing policy database implementations.
    - Study how to emulate or simulate complex IPv6 routing scenarios (IXs with a high number of providers or a high number of routes).
    - Follow the RPSLng standardization process providing feedback from Euro6IX experiences.
    - Propose modifications and/or new developments of functionalities missing (RPSLng support, for example).
- Set up an IPv6 routing policy registry.
    - Set up a private routing registry for Euro6IX project participants.
    - Study the feasibility of extending the routing registry to other IPv6 networks/IXs.
- Set up route servers in some of Euro6IX IXs.

- Install selected route server implementations and experiment with typical procedures: Addition/deletion of ISPs, interaction with routing policy registry, etc.

- Set up route monitoring tools in all Euro6IX IXs (publicly accessed or maybe restricted to consortium members).
  - Set up Looking Glasses, a “Route Views” server and other tools like ASPath-Tree.

6. TRANSITION FACILITIES

This section lists the different transition facilities that an IPv6 IX may offer. These facilities are based on the usage of transition tools currently defined today. The goal is to provide temporary solution enabling access to the IPv6 IX and/or enabling some means of heterogeneous communication between IPv4 and IPv6 nodes. Since it is assumed that legacy IPv4 world will coexist with the new IPv6 world for a long period, an IPv6 IX may implement such facilities.

6.1 Transition Scenarios Involving IPv6 IXs

The main scenario we are currently focusing on is the scenario of an emerging IPv6 "new world" that is being deployed over a dedicated IPv6-only infrastructure. Then three different kinds of facilities may need to be provided:

- Facilities enabling the transport of IPv6 packets over remaining IPv4 piece of network, in order to access anyway the IPv6 IX.
- Facilities enabling heterogeneous communication between an IPv4-only node and an IPv6-only node, in order to provide some means of gateway between IPv4 and IPv6 network.
- Facilities enabling the transport of IPv4 packets over IPv6 network, in order to provide IPv4 connectivity over IPv6-only network.

Another possible topic to be analyzed could be the introduction of IPv6 protocol within an existing IPv4 IX. This scenario is out of the scope of this study, since Euro6IX project is based on the deployment of a dedicated IPv6 infrastructure.

At the end, another further point of attention could be the possible union of a legacy IPv4 IX with an IPv6 IX, in order to share the same infrastructure. This scenario is left for further study, and is also out of the scope of the Euro6IX project.

The sections below will detail facilities of the former scenario, based on currently defined transition tools (tunneling or translation tools). The different facilities described in the section below, enable to meet short-term or long-term transition needs.

6.2 Tunnel Broker

The Tunnel Broker tool **[24]** enables an isolated dual-stack host or an isolated IPv6 domain behind a dual-stack router, within the legacy IPv4 ocean, to get IPv6 connectivity through an IPv6-in-IPv4 tunnel. After a negotiation phase between the IPv6 host/router and the tunnel broker (TB), the tunnel server (TS) will set up the tunnel part from the IPv6 world to the dual-stack host/domain, while the TB returns some means (e.g. a script) to set up the reverse part of the tunnel. The Tunnel Broker will also allocate a permanent global IPv6 address or prefix, and permanent DNS names as well.

The Tunnel Broker tool is an efficient way to automate tunnel set up, and then to provide IPv6 connectivity over a legacy IPv4 network, IPv6 global address and DNS names as well.

Since the control of such a service relies on the Tunnel Broker that can be easily managed, an IPv6 IX may offer services based on this access technique. Basically, two levels of service can be defined:

- Simple Access Service: In this case, an IPv6-in-IPv4 tunnel replaces the classical L2 link. Tunnel Broker tool provides to the customer a basic IPv6 connectivity, typically allocating to the client remote node a /127 address, and setting up a tunnel between the former node and the IX. Then, a classical BGP session can be set up.
- Complete Access Service: In this case, where no BGP session needs to be set up, the IX provides IPv6 connectivity and prefix allocation (e.g. /64 or /48 prefix). The service may include also permanent DNS names management.

The main constraint relies on the user itself, which must have a fix global IPv4 address that is used for the tunnel configuration with the Tunnel Server. Such a constraint may not be met by users (e.g. dial up users), which get access to the Internet from an ISP that shares a pool of IPv4 addresses.

Such a facility may be needed during the beginning of the transition phase, when only little native IPv6 connectivity is available.

6.3 6to4 Relay

6to4 tool [25] defines a way to assign an interim unique IPv6 prefix, and a mechanism to encapsulate IPv6 prefix over IPv4 without explicit tunnel configuration. The main characteristics of 6to4 tool are reminded below:

- A 6to4 router is placed at the border of the unique 6to4 domain it serves, and the legacy IPv4 Internet.
- The interim unique IPv6 prefix is formed as follow: 2002:V4ADDR::/48, where V4ADDR is a global IPv4 address.
- The global IPv4 address V4ADDR is used by the encapsulating mechanism, since it ends every non-explicit tunnel to concerned 6to4 router.

Thus, 6to4 domains get an easy and transparent IPv6 connectivity with each other.

RFC3056 identifies mixed scenarios, which enables a so called 6to4 relay router to provide, either IPv6 connectivity to 6to4 domains, or IPv6 connectivity to isolated regular IPv6 domains using normal 2001:: TLA.

An IPv6 IX may deploy such a 6to4 relay router as alternative access facility, in order to provide IPv6 connectivity, either to isolated 6to4 domains, or to regular IPv6 domain. The IX's 6to4 relay router, located at the border of the IPv4 legacy Internet and the IPv6 new world, has two configuration options, with or without BGP4+:

- 6to4 relay without BGP4+: The 6to4 relay router may apply source address filtering, in order to accept traffic only from chosen domains. This is the only routing policy that can be deployed in this case.
- 6to4 relay router with BGP4+: This option enables to deploy a real, more complex, routing policy, choosing the routes to advertise and the traffic to accept.

Such a facility may be needed during the earlier transition phase, while only little ISPs offer IPv6 access services. This solution aims to meet short-term transition needs.

6.4 DSTM

DSTM (Dual Stack Transition Mechanism) tool [draft-ietf-ngtrans-dstm-08.txt] enables a dual-stack host connected to an IPv6-only network to get some temporary IPv4 connectivity. In some way, DSTM provides the facility complementary to that one provided by the Tunnel Broker tool.

When a DSTM client wants to communicate with an IPv4-only node, it gets an IPv4 temporary address from the DSTM server (the IPv4 address is temporary allocated using a protocol that is implementation dependant). The IPv4 traffic is tunneled to the tunnel end point (TEP) through the dynamic tunnel interface (DTI), and then forwarded within the legacy IPv4 network to its destination.

This transition mechanism, which relies both on the DSTM server and on the TEP, may be easily managed and provided by an IPv6 IX, in order to offer native IPv4 connectivity and it may be needed in the medium or later phase of the transition, when IPv4 global addresses are part of scarce resource. This facility assumes in fact that most of the traffic is based upon IPv6 and that only some remaining communications need IPv4 connectivity.

6.5 NAT-PT

NAT-PT tool **[26]** intends to provide transparent routing between IPv6 and IPv4 networks, so that a given NAT-PT box may act as a gateway from the IPv6 new world to the IPv4 legacy Internet with the aim of enabling its IPv6 clients' domain to communicate transparently with IPv4-only domains, and possibly vice versa.

However, RFC2766 makes the fundamental assumption: *"NAT-PT is only to be use when no other native IPv6 or IPv6 over IPv4 tunneled means of communication is possible. In other words the aim is to only use translation between IPv6 only nodes and IPv4 only nodes, while translation between IPv6 only nodes and the IPv4 part of a dual stack node should be avoided over other alternatives."* So that NAT-PT facility should be used as a last chance service.

Two types of NAT-PT are identified: basic NAT-PT that enables only communication initiated by IPv6 nodes, and bi-directional NAT-PT that enables communication initiated from both sides. The former case assumes that the IPv6 domains do not need to deploy servers that need to be visible from the legacy IPv4 Internet.

In both cases, and in order to avoid any specific configuration on the IPv6 domains side, the NAT-PT facility needs to support the defined DNS-ALG. This is the major constraint, among the various restrictions of the NAT-PT tool, since it assumes that the all the traffic has to go through the NAT-PT and its DNS-ALG, creating at least a single point of failure. Some other issues have been identified in [draft-durand-natpt-dns-alg-issues-00.txt], and a distributed solution (NAT-PT is a centralized solution) has been proposed in [draft-durand-ngtrans-nat64-nat46-00.txt].

Anyway, an IPv6 IX may choose to deploy a NAT-PT facility based on RFC2766, with a great care, according to the various limitations of this tool.

6.5.1 Implementation Specificities

According to [draft-ietf-ngtrans-interaction-01.txt], the deployment over an IPv6 domain of the NAT-PT tool supporting a DNS-ALG, avoids deploying DSTM tool that requires DNS answer to be unchanged.

7. IPV6 MONITORING FACILITIES PLACED WITHIN THE IX

In Euro6IX backbone all the machines inside the network (i.e. routers, switches, servers) have to be monitored. The approach followed in this section is to take into account the various architectural layers, as defined in Figure 2-1, and for each layer to individuate the suitable parameters that better describe the status of each machine inside that layer.

For example, this section has been divided in "layer 2" monitoring part, when the switch infrastructure inside the Internet exchange will be monitored, in "layer 3" monitoring part, when the router infrastructure will be monitored, "Server-Farm" monitoring part, if the servers (e.g. WWW server or DNS) will be monitored.

Generally, there are two available monitoring solutions: the first one is the so-called "passive" or "non-intrusive" method, because the testing operation does not affect the network status. This method is not suitable to individuate the exact fault location, and, since it requires the capture of all the packets, security and privacy issues can arise in accessing the data to be collected. Examples of such techniques include Remote Monitoring (RMON) **[27]**, Simple Network Monitoring Protocol (SNMP) **[28]** and Netflow **[29]** capable devices. The passive monitoring devices are polled periodically and information is collected (in the case of SNMP devices the data is extract from Management Information Base, MIB) to assess network performance and status [2].

The other approach is known as "active" or "intrusive" approach. In this case, the monitoring is realized injecting test traffic into the network, sending packets to servers and application, following them and observing the network behavior. The advantages of this solution are in the possibility to generate the traffic pattern according to the needs, e.g. modifying the type of traffic sent (UDP, TCP) or the packet size, and in the possibility to use different sampling techniques. Though small test traffic volumes are needed to obtain meaningful measurements, the main drawback is probably in the generating some traffic adding to that already flowing in the network and this could cause, depending on the network status, some variations in the results.

The adopted solution will depend on various factors and the decision should be taken on the basis of a compromise between economical feasibility of the solution and level of accuracy to obtain in the measurements.

7.1 Layer 2 Infrastructure Monitoring

A typical IX architecture consists of one or more switches (typically more than one are requested for redundancy) where different routers are linked. The switches could be linked among each other by high-speed links (e.g. Gigabit Ethernet connections) and, depending on the IX network architecture and largeness, they could be organized in a multilayer hierarchical structure. Consequently, the switch infrastructure is really the core of the IX and consequently its monitoring is a basic need.

However this infrastructure should not be a critical point regarding the performances in terms of throughput or packet loss or RTT. In fact, the switching module has generally a switching packet

[2] *As alternative method to classical SNMP-based solution, suitable test probes can be used. This solution links high-level measurement accuracy to the advantage of not modifying the network status, since no added traffic is added. This solution could be, on the other side, more expensive of the SNMB-based one.*

capability larger[3] than layer 3 infrastructure's forwarding packet capability, so it is unlikely that the layer 2 infrastructure can be a bottleneck[4].

In any case, parameters that could be interesting to monitor, for each switch port, are the *transmitted* and *received packets* and *packets loss*.

Concerning the packet loss parameter, it is important to note that, although the architecture should consist of routers directly connected to the switch ports (i.e., no packet loss, theoretically, between router and switch), it is needed to monitor the traffic in both directions. In fact it could happen that some packets are lost by the router (direction from switch to router) or by the switch (direction from the router to the switch), e.g. due to losses inside the networks previously got through.

7.2 Layer 3 Infrastructure Monitoring

7.2.1 Routing Monitoring

Inside the Euro6IX backbone, BGP4+ is the routing protocol that should be used. The Border Gateway Protocol is an exterior gateway protocol used to exchange routing information among routers in different autonomous systems. It is a distance-vector routing protocol and runs over the Transmission Control Protocol (TCP) as its transport protocol, using port 179 for establishing connections. Running over a reliable transport protocol eliminates the need for BGP to implement update fragmentation, retransmission, acknowledgment, and sequencing used to exchange routing information among different ASs. BGP routing information also includes the complete route to each destination.

Moreover the routing information is used to maintain a database of network reachability information, which it exchanges with other BGP systems. In fact BGP uses the network reachability information to construct a graph of AS connectivity, thus allowing BGP to remove routing loops and enforce policy decisions at the AS level.

One of the main aims of the monitoring facilities placed in the IX will be to give a detailed view of the paths of routing protocol inside the backbone. To achieve this aim, every backbone router should be provided with a BGP4+ protocol-monitoring tool. In the following, the parameters retained the most important to monitor are indicated:

- *AS-Path:* This is a well-known BGP4+ attribute used to identify the path of routing information. This parameter can be used to create a graph of the network, from the AS point of view, i.e. a graphical overview identifying which AS are got through by the BGP4+ announcements. Each AS can be or an origin AS, being itself the AS generating the announcements, or a transit AS, since it does not generate any announcements and the routing information only pass through. The information can also be used to detect loops.
- *Peering BGP status:* It is possible to obtain information on how many and which routing update messages are exchanged in both directions, over the TCP connection, with the other BGP4+ speakers. So it could have an overview about the routing messages incoming and outgoing for each IX. Since routing updates carry also the network prefixes, this information could be used to keep track of the network prefixes advertised with BGP4+ towards a given IX. This information is very useful, since it permits to have

[3] *The reasons to explain the good performance of a switch are basically: switching functionality is hardware-based (faster than a software-based switching) and the switching module consists of a non-blocking matrix.*

[4] *Though the switch should not usually modify the performances, in terms of throughput or RTT or packet loss, if some hardware trouble is present in the machine, it is possible performances get worse.*

the control of the announcements present inside the network (e.g. to prevent by having prefixes outside of the IANA assigned address spaces) and to individuate very quickly the presence of un-aggregated network prefixes. This last information is very important since, in the core of the network, it should have only aggregated prefixes, just to avoid the routing table exploiting in the backbone routers.

- *Route instability:* It can happen that an entry, in the BGP4+ routing table, be not stable, i.e. there are frequent changes in some entry (flapping phenomenon).
- *Route unavailability*: It could happen temporary or permanent unavailability of a route to a destination.

Note that the information we could obtain regards only the routing inside the Euro6IX (i.e. between the border routers speaking BGP4 +) backbone not considering the customers linked to IX (i.e. the ISPs).

7.2.2 Routers Monitoring

All the routers placed in the IX should be monitored, both on the backbone and on the access side. Main performance parameters to have into account are: the CPU load, the packet loss and the throughput corresponding to the different network interfaces[5].

Depending on their use, many other parameters could be considered, e.g. the queue status, the applied traffic policies and so on.

7.3 Route Server Monitoring

The Route Server makes easier routing information exchange among the IX-attached users by collecting routing information from ISP routers, processing the information based on the user's routing policy requirements, and passing the processed routing information to each IX router.

It is a really critical point inside the IX and main parameters that could be monitored are:

- CPU.
- Memory Usage.
- Exchanged entries.
- BGP4+ sessions.

Referring to the memory usage, a Route Server will have to manage (i.e. to collect) many BGP4+ entries (depending on the users attached to the IX) and, consequently, more the BGP4+ entries more is the memory occupancy.

Concerning the CPU, it will be more solicited above all in transient periods, like starting up phase or when a session BGP goes down and update message are required to inform the BGP4+ speakers of the changes in the routing. In these situations, the CPU usage is at very high values, while in “normal” conditions (stable routing) the CPU has acceptable values. CPU load and memory usage are the most important parameters and they have to be monitored to guarantee a good working of this machine.

Additionally, there are the last two parameters (closely linked to the first ones) whose monitoring is a need. In fact it is possible that both the number of sessions and the announced entries can

[5] *For a more detailed discussion about performance measures, see Reachability and Performance Monitoring section.*

somehow affect the CPU load and the memory usage. In fact, with a simple consideration, if more entries are announced more room will be necessary in the route server memory and if more BGP4+ sessions are to set-up higher will be the route server CPU load.

7.4 Server Farm Monitoring

7.4.1 DNS Monitoring

DNS Server placed in IX infrastructure will have mainly function of cache DNS sever that permits routers and other servers to resolve direct (AAAA) Reverse Records and inverse (PTR) queries made by DNS clients. So, main parameters that should be monitored are:

- Response time of cached queries.
- Response time of non-cached queries.
- Number of queries (classified by type).
- Percentage of success of cached queries
- CPU Load.
- Memory.
- Disk usage.

7.4.2 HTTP Server Monitoring

The monitoring of the HTTP server can be realized under many points of view, depending on what to do with the collected information (accounting, billing or simple monitoring for statistics) and the level of detail that this information is required with (e.g. monthly, daily or hourly).

In a starting phase, the main aspects to be monitored in the web server could be:

- Hardware layer: CPU, Disk and memory.
- Service layer: URLs accessed, data transfer time, total response time[6], cache hit ratios, number of successful requests, number of failed and redirected requests and total data transferred.

Other information can be obtained about the top origins of visitors to the sites (by the suffix of their domain name) or about the top computers or ISPs of visitors to the sites. Moreover another useful information could be obtained by analyzing a file-type report that identifies information that is requested from the web sites (e.g. txt files or GIF or HTML files).

It could be also interesting to generate a status code Report listing the HTML headers returned to the client from the server. For example a status '200 OK' means that the requested page or image was found and the server will now send it. A '404 Document Not Found' means that the requested page or image cannot be found on this server at the specified location. This can occur if the client mistyped a URL or clicks on a broken link.

[6] *Total response time is approximately the sum of DNS_TIME+TCP_CONNECTION_TIME +DATA_TRANSFER_TIME. These times can be quite different depending on how many requests are processed in the same time by the server.*

7.4.3 Transition Facility Monitoring

The Server Farm also includes the Transition facility, because the IPv4 users must be supported, and consequently transition from IPv4 to IPv6 is needed. Parameters to be monitored depend on which transition mechanism is taken into account.

Concerning the Tunnel Broker mechanism, monitoring parameters could be:

- Active tunnels.
- Configured tunnels.
- CPU load.
- Memory usage.
- Disk.
- Traffic.

Regarding the NAT-PT, following parameters could be monitored:

- IPv4 address pool[7].
- Connections towards NAT-PT router.
- CPU load.
- Memory usage.

Concerning the 6to4 relay mechanism, main parameters to take into account could be:

- IPv4 packet entering the 6to4 relay routers.
- Active tunnels.

7.5 Reachability Monitoring

The IX reachability monitoring aims to assess if an IX is working or not, i.e., if there are any troubles in IP-level (or lower) connectivity[8]. The simplest way to do these measurements is probably to use the ICMPv6[9].

ICMPv6, like its predecessor, ICMPv4, is a protocol with diagnostic functionality that permits to establish if a particular machine (switch, router or server) is active or not, or in other words, if it is reachable.

The basic idea to test reachability is to send a set of echo requests towards a specific set of destinations in the Internet (manually or, better, via an automatic software tool) and to wait for the responses (echo reply). Generally, one can say if there is the reply to the request, the queried machine is reachable; otherwise the machine is not reachable.

In making these reachability measurements, note that, being ping-based, they are strongly dependent on topology and status[10] of the network got through by the echo requests. Concerning the status of the network, the measurements in fact are conditioned by “queuing” delay, due to

[7] *NAT-PT translates the IPv6 address inside the packets coming from the IPv6 host into IPv4 address chosen by an address pool configurated on the machine.*

[8] *If an echo reply does not come back, there can be many different reasons, from physical level trouble (e.g. interface down) to IP-level trouble.*

[9] *However the measurements can be realized also with non-ping method. In this case FTP and traceroute application can be used instead of ping.*

[10] *“Status” refers to the link occupancy.*

different status of the queues of the machines in the network. Hence a router in a congested network could discard an ICMP packet and the sender could wrongly retain, during the monitoring, this packet loss as a reachability issue (as if the queried machine was not active): In this case the conclusion would be wrong.

Moreover, all the measures ping-based are also conditioned by other three factors:

- First one regards the point from where the ICMP packets were sent. Indeed the only element to have into account is the delay introduced by the networks element; in other words, the links don't introduce any transmission delay and the only limiting factor is the delay due to network elements. So, referring to the value of the measure, it can be different to ping from a router inside the IX (where more networks elements have to be passed) or from a router directly linked to the link towards the other IX.
- Second factor to have into account is that the test traffic profile in the network is different by that one really flowing on the network. ICMP packets are usually shorter than the real packets and all of them have the same length. In the real traffic the packets have different length and normally the traffic is UDP or TCP or a mix of both. Consequently test realized give some results that are valid in a certain context, i.e. with a particular kind of traffic, but the results could be deeply different if the real traffic was used to do the tests.
- Third factor regards how the network load can affect the monitoring. In fact, it's possible to modify the rate of the ICMP packets sent in a second. This way, by increasing the amount of packets generated, the measure of RTT parameter will be more exact, but depending on how many packets are sent, network conditions[11] could be modified (mainly, in terms of congestion) and the measurements could be wrong.

The reachability information is not the only information that can be obtained with the ping-based method. In fact, starting from the ping measurements, information about the *network performances* and about *the network trend* can be obtained.

Concerning the network performances, main parameters to assess the network status are: *Throughput, packet loss and the response time (i.e. RTT)*. Also in this case, some simple explanations about these terms are provided.

The packet loss can be considered the fraction of sent packets that does not receive the acknowledgement from the destination testing point. These measures also include the packets that do not arrive to the test point and the acknowledgements lost before returning back.

Round Trip Time can be thought as the time range between, a time measurement agent sends a packet to a destination test point, and the time the test point sends an acknowledgement back to the agent. RTT includes all the delays in the intermediary nodes and in end machine, but does not include the DNS lookup time.

Throughput is the maximum speed to send the packets without being discarded.

A first method to evaluate them is based just on the ping application, as previously said, and their estimation is made on the basis of some statistical calculation, starting from the data collected with ping-based method.

[11] *Particularly in low traffic situations, where a high ICMP traffic could be comparable to the real traffic and modify the bandwidth occupancy of the links.*

As further specified in the next section, these parameters can also be evaluated with monitoring traffic tools[12] that permit the analysis of the traffic flowing through the network, estimating some parameters as packet structure or protocol distribution, but also the throughput or the packet loss.

Moreover by using ping-based tools much other information, if needed, can be obtained as, e.g. the inter-packet delay, minimum packet loss or TCP throughput to have a more detailed overview of the backbone. On the other hand, the "medium-term analysis" permits to understand if the network performances are improving or getting worse, making a forecast of the performance expected in the medium term in terms of capacity to transfer bytes inside the network. Another interesting factor to take into account is the so-called "ping predictability"[13] or "service predictability" factor that permits to have the measure of service variability (e.g. the variation in a WEB page download time), from the user's perspective.

This method is based on the realization of a graph where two dimensionless variables[14] are plotted. If in this graph, values close to 1 are obtained, it indicates that average performance is close to the best performances.

7.6 Traffic Monitoring

Monitoring facilities placed inside IX will also encompass a tool to monitor the data traffic going into and coming out of the IX. The main goal of the traffic monitoring is to analyze the traffic that flows into the network and show to the human operator what is into the network.

A first classification can be made based on the kind of tool to use. There are in fact both packet analyzer and traffic analyzer. The choice depends on what one wants to aim. A packet analyzer captures the packets inside the network, dissects them and provides an internal view of the packet. For this reason, most of the traffic monitoring tools gives information on the protocol distribution, packet size distribution, information on service quality (if enabled over the machines), bandwidth distribution. Moreover throughput measures are realized with these tools, e.g. estimating the current and actual bandwidth[15]. The traffic analyzers are, on the other hand, tools used to look at the traffic at a higher level, i.e. without analyzing inside the single packet, but providing both an aggregate view and some advanced measurements for other aims as billing and accounting.

---

[12] *From this point of view, a classification of the tools is not simple, because some tools (considered as traffic monitors) give also information normally provided with performance analyzers tools. Normally the traffic monitoring tools have, as main aim, to look at the traffic flowing through the network, looking into the packets, dissecting them and showing their structure. No performance evaluation, by definition, should be done. However there are tools that, added to these "simple" monitoring functionalities, can give the estimation of parameters more related to the performance of the network, as RTT or throughput. In this document, we choose to distinguish the traffic monitoring from the performance evaluation, but practically they can be strictly related and, for short, it is possible to find tools working both as traffic monitor and as performance analyzers. Two functionality are generally mixed among each other and no clear division can be realized.*

[13] The service predictability parameter has been defined at SLAC (Stanford Linear Accelerator Center) by Les Cottrell, Warren Matthews and Connie Logg.

[14] *The graph is (average ping data rate / maximum ping data rate) versus the (average ping success / maximum ping success), where:*

ping success = (total packets - packets lost) / total packets

*ping data rate = (2 * bytes in ping packet) / response time*

The factor "2"in the last expression is because the packets have to go and come back.

[15] *As explained in note 12, most packet analyzing tools not only analyses the traffic but also calculates performing parameters like packet loss or RTT or throughput.*

7.7 Implementation Options

This section aims to provide a TILAB proposal about the tools that could be used for realizing the monitoring functionalities previously discussed. These tools not only satisfy the requisites needed to match the key points already illustrated (see "Reachability monitoring" and "Traffic monitoring" paragraphs), but also implement some other added functionality to provide a better network monitoring.

Moreover, these tools are all developed to work in a dual-stack network, in fact, as better detailed in the following, the queries to the suitable servers require both IPv4 and IPv6 connectivity. If these tools will be chosen and used, it will be necessary to upgrade their software so to be able to use them in a full IPv6-based environment.

7.7.1 Routing Monitoring Tool: ASpath-Tree

The AS-Path tree tool [30] for routing monitoring inside the Euro6IX IX, has been designed to be used by an IPv6 site involved in the experimentation of the BGP4+ protocol inside the 6Bone network. It has been developed in TILAB internetworking laboratories and it is made up of a collection of UNIX scripts written in Perl 5.0. It automatically generates a set of HTML pages providing a graphical view of the routing paths towards the other 6Bone sites inside the BGP4+ cloud.

The root node of the tree is the node (i.e. IX) where the monitoring facility is placed. The leaf sites are reported in the tree and each of them can be or a transit site (if it does not advertise any routing information) or an origin site (if it advertise routing information towards the other sites).

Additionally, it provides pages for the detection of BGP4+ routing anomalies, i.e. invalid[16] and un-aggregated[17] prefixes and BGP4+ routing stability analysis. Current version of ASpath-tree (version 3.3) also includes the following features:

- Other IANA assigned prefixes (200x) routing tree.
- Other IANA assigned prefixes (200x) stability analysis.

The outputs are obtained elaborating the AS path information of all the BGP4+ routing entries available on an IPv6 border router terminating BGP4+ capable tunnels[18].

It was tested on a Solaris 2.5.1 workstation but it should work without problems on any UNIX workstation (FreeBSD, Linux, etc.) equipped with a Perl 5.0 interpreter. BGP4+ routes are collected using RSH commands. In particular, the **current release works only with Cisco routers** equipped with an IPv6 capable IOS.

In Telecom Italia Lab ASpath-tree is currently used to monitor TILAB's BGP4+ routing configuration. The tool shows the BGP4+ routing tree from TILAB to the 6Bone backbone and from TILAB to the whole 6Bone and the update is automatically run every 5 min.

Other information can be obtained by the Aspath-tree as the prefix assignation, distinguishing among Origin AS, pTLA prefix owner and company that the prefix is assigned to.

---

[16] *Prefixes outside the IANA assigned address spaces.*

[17] *Prefixes belonging to 6BONE range but longer than a given pTLA delegation (currently /24 and /28). According to RFC 2456, these prefixes MUST NOT be advertised unless special peering agreement is implemented.*

[18] *Inside 6BONE backbone, the links among the PoPs are mostly realized using tunnel IPv6 in IPv4 mechanism.*

7.7.2 Traffic Monitoring Tool: Cricket

Cricket is a system to monitor the traffic, written entirely in Perl and distributed under the GNU General Public License. It consists of two modules: a collector and a grapher.

The collector is a software module running over a manager machine querying periodically (according to a pre-selectable sampling rate), via SNMP or other protocols, such as SSH, RSH, the involved machines (routers, switches and servers). Data collected are then stored in a data structure managed by RDD Tool. RRD Tool is a C program, which takes care of storing and normalizing the data after the collector fetches it, and draws the data into graphs when the grapher asks it to. Like Cricket, RRD Tool is distributed under the GNU General Public License.

The other module is the grapher that provides a graphical view of collected data, by means of a web-based interface.

Cricket tool is able to provide information about any parameter which has a MIB associated to, for example number of packet sent on a given interface, and it can monitor both the traffic ingoing and outgoing the IX flowing through the backbone router or the packet incoming or outcoming from the switches. Moreover, the collected information can be graphed in different time ranges (daily, weekly, monthly, yearly).

Other parameters can be monitored, like the CPU load, the temperature and (in the case of servers) the disk usage.

Cricket is developed on Solaris machines running under Apache. It is known to work on Linux, HP-UX, variants of BSD, and other operating systems. Some users are successfully using Cricket under Windows NT and/or Windows 2000, but at this time, no one has documented this.

Two main limits envisaged in using this monitoring tool are the effective presence of MIB for IPv6 (in fact not all the MIB available in IPv4 are also available in IPv6) and the IPv6 support by the routers or servers to be monitored. In fact, at the present time, the MIB from queried machines is carried, into the manager server (where the collector is placed), over IPv4 protocol[19].

7.7.3 Reachability Monitoring Tool: Ping View

The ping-view tool has been entirely realized in TILAB networking laboratories and provides some information about the connectivity towards a specific set of destinations in the Internet (host or routers).

The reachability information for every monitored site includes the packet loss and the Round Trip Time. Data are collected by sending sequences of ICMP echo requests (one per second) of fixed length towards all the selected destinations using the ping program. The answer to the first echo request of every sequence is always rejected because it is usually slower than the others (e.g. due to caches priming). If the network experiments high values of packet loss and Response Time, that indicates low connection performances. The acquisition of the reachability information towards all the monitored sites is repeated periodically (generally once an hour).

Corresponding to every data acquisition, the measured values (packet loss and RTT) are recorded in an archive in order to be available for further analysis and elaborations. In particular this information is used to create various kinds of graphic representations of the collected data

[19] *A software tool that can be used to provide a description of the traffic, in terms of application distribution or bandwidth distribution, is the Analyzer, developed by the Politecnico di Torino.*

(including loss and RTT versus time, loss and RTT frequency distributions, the Quality Factor[20] and the Service Predictability) and some history-based parameters ("$RTT_{70\%}$ - last 7 days"[21] and trend).

The latest ping results are used to produce a summary table displaying the latest values of packet loss and latency towards every monitored site is automatically created. Additionally, an estimation of the overall connection behavior during a selected timeframe ("$RTT_{70\%}$ - last 7 days") together with an indication of the medium term trend (time-scale of days) is provided.

Concerning the performances variations, looking at the summary table displaying the results of the latest measures, occasional degradations in the performance of the communications can be discovered by.

In order to provide some information about the medium term evolution of the network performance (i.e. if they are improving or if they are getting worse), it was decided to compare an effective[22] bit rate value calculated from loss and response time values measured over the last 7 days (medium term network performance) with the one calculated over the last 4 hours (short term network performance)[23].

An improved bit rate in the last 4 hours is shown as a positive trend, while a slower bit rate in the last 4 hours is shown as a negative trend. If, on the other hand, the two values are very close together an indication of substantial stability in the network performance is returned. In other words, this parameter provides practical information on the amount of bits that networks is able to carry.

The ping-view provides also statistics both for single sites and for sites groups. In fact different destinations may be grouped to show an estimation of the "network" performance. Groups of destinations, defined to put together destinations belonging to given "parts" of the network, may be used to show aggregated reachability statistics towards these "parts" of the network. Ping-view tool provides an aggregated view for custom defined groups, showing a summary table containing an estimation of the overall "group of connections" behavior ("$RTT_{70\%}$ - last 7 days") together with an indication of the medium term trend (time-scale of days).

---

[20] The "Quality Factor" quality parameter is an estimate of the probability of getting an answer to an ICMP Echo Request within a given RTT^* and is calculated as the number of Echo replies obtained back within RTT^* versus the total number of Echo requests sent to the destination(s). This parameter is calculated from the loss and RTT values collected during a given time frame. The Quality Factor (QF) is a function of RTT^* and gets values in the range [0,1]. A low value of $QF(RTT^*)$ indicates low connection performances while a value of $QF(RTT^*)$ very close to 1 stands for good performances.

*[21] "$RTT_{70\%}$ - last 7 days" is a performance parameter defined as the minimum response time within which the 70% of replies to the transmitted echo requests are received back during the last 7 days of data acquisition, in order to provide a "single value" indication of the network performance. If the overall ping success towards a specific site during the period of time chosen for the calculation of "$RTT_{70\%}$ - last 7 days" is less than 70% this parameter can not be calculated and a not rated (*n.r.*) indication is shown. Higher values of "$RTT_{70\%}$ - last 7 days" or the n.r. indication show lower communication performance.*

[22] The effective bit rate is calculated as follows:

*(2 * bytes in ping packet *8)*(total packets - packets lost) / total packets / average response time*

[23] The final output is affected by a tolerance due to sensibility of average response time parameter at the denominator of the formula.

8. APPLICATIONS SERVICES PROVIDED BY EACH IX

Users linked to the IX will be able to access various kinds of application services hosted in the IX Server Farm. The aim of this section is mainly to describe why and how these services could be used in an IX environment.

The usage of the network resources inside a hierarchical network can be improved if the distance between customer and service access point (IX, in our case) is reduced. This way, putting inside the IX all most accessed services, we would reduce not only the cost in using the network resources (since less hierarchical steps have to be done) but also the probability of having errors and delay. Consequently, it could be affirmed that having application services that generate a lot of traffic and a lot of requests, within an IX, is an optimization.

Consider the situation depicted in Figure 8-1 a), where n is the 'cost' of a message between a client and the IX, and m is the 'cost' of a message between the server and the IX. If R represents the requests in a single day, and u represents the updates of the service (by a client located anywhere in the network), always in a single day, this day's total cost will be:

$$C(a) = 2R(n+m) + 2u(n+m) \qquad (1)$$

Following the model in Figure 8-1 b), we will have a cost of:

$$C(b) = 2Rn + 2un \qquad (2)$$

Since, usually, the number of the requests is much higher than the number of the updates, then $R >> u$ and consequently $C(a) = 2R(n+m)$ and $C(b) = 2Rn$.

It demonstrates that, with the hypothesis $R>>u$, C(a) is higher than C(b).

This very simplified model just gives us the 'feeling' that by installing a very accessed application service within an IX it could be possible:

- To reduce the response time of the service.
- To reduce the probability to have service unavailability.
- To reduce the traffic generated in the network.

Finally, remind once again that if the assumption of a very accessed service ($R >> u$) is valid, then it is possible to conclude that these kind of services are the only that could be better to install in the IXs.

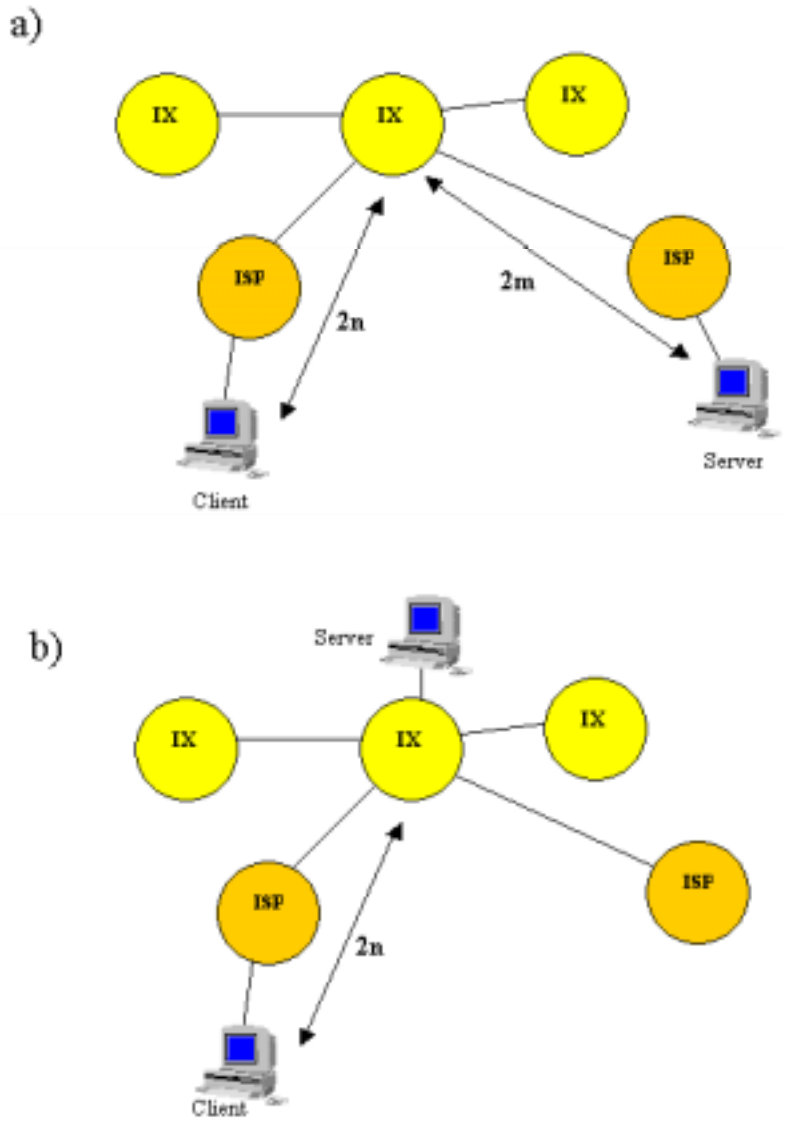

Figure 8-1: Why an Application Should be in IXs?

The following paragraphs contain a proposal about the application services that could fit to the characteristics stated before and that could be provided by each IX. Among these applications we consider Basic Internet services (DNS, NTP), Content Delivery Services (HTTP, S-HTTP, FTP, TFTP) and Network Access Services (AAA, RADIUS, DIAMETER). Other services as E-mail (SMTP/POP3/IMAP), SSH, LDAP and RPC are included too.

8.1 Basic Internet Services

8.1.1 DNS

The Internet Domain Name System (DNS) is a distributed hierarchical database that permits to establish a correspondence between names and IP addresses. Clients look up information in the DNS by calling a resolver library, which sends queries to one or more name servers and interprets the responses.

The Berkeley Internet Name Domain (BIND) implements a domain name server for a number of operating systems:

- IBM AIX 4.3
- Compaq Digital/Tru64 UNIX 4.0D
- Compaq Digital/Tru64 UNIX 5 (with IPv6 EAK)
- HP HP-UX 11
- IRIX64 6.5
- Sun Solaris 2.6, 7, 8
- NetBSD 1.5 (with unproven-pthreads 0.17)
- FreeBSD 3.4-STABLE, 3.5, 4.0, 4.1
- Red Hat Linux 6.0, 6.1, 6.2, 7.0

BIND 9 fully supports all currently defined forms of IPv6 name-to-address and address-to-name lookups. It will also use IPv6 addresses to make queries when running on an IPv6 capable system.

For forward lookups, BIND 9 supports both AAAA and A6 records. A6 was moved to experimental in IETF environment, but it is still useful for hosts to have both A6 and AAAA records to maintain backward compatibility with installations where A6 records are used. In fact, the stub resolvers currently shipped with most operating system support only AAAA lookups, because following A6 chains is much harder than doing A or AAAA lookups.

For IPv6 reverse lookups, BIND 9 supports the new "bitstring" format used in the ip6.arpa domain, as well as the older, deprecated "nibble" format used in the ip6.int domain.

DNS Server placed in the several Euro6IX IXs will have mainly the function of cache DNS server that permits routers and other servers to resolve direct (AAAA) Reverse Records and inverse (PTR) queries made by DNS clients.

Nowadays, there is an open discussion about the possibility that the IX assigns its own addresses: In this case the IX should manage the reverse delegation of its addresses.

8.1.2 NTP

The Network Time Protocol (NTP) allows the time synchronization for computer clocks through interconnected networks. It uses a returnable time design in which a distributed sub network of time servers, operating in a self-organizing, hierarchical master-slave configuration, synchronize logical clocks within the sub network and to time standards mechanisms via wire or radio.

The notion of time is problematic in distributed systems, as the clocks at computers are subject to clock drift, which means that they count time at different rates, and so diverge. Synchronizing clocks is subject to the constraints imposed by message passing. The problem arises because of the unpredictable amount of time needed by the messages. The most used reference of time is UTC (Universal Coordinated Time), which is a standard based on atomic time, but a so-called leap second is occasionally inserted or deleted to keep in step with astronomical time.

NTP is intended to provide:

- A service enabling clients across the Internet to be synchronized accurately to UTC.
- A reliable service that can survive lengthy losses of connectivity.
- Sufficiently frequent resynchronization to clients.
- Protection against interference with the time service.

A NTP IPv6-ready service could be used within Euro6IX, for example, by some security application that use timestamps and a "time window" to accept the timestamps. If this window is 5 minutes long and the difference between the clocks is over 5 minutes, two entities could never reach an agreement, i.e., get the authentication task. There are several algorithms that depend upon clock synchronization, for example, checking the authenticity of a request to a server using kerberos.

A network of servers located across the IXs, could provide the IPv6 NTP service (see the Figure 8-2). Primary servers are directly connected to a time source such as a radio clock receiving UTC. Secondary servers are synchronized, ultimately, to primary servers, and so on. The latest level servers execute in users' workstations. Errors are added at each level of synchronization.

Following this model we could have several configurations, which could coexist. See the Figure 8-3. The IX's NTP server would synchronize with the time source and with the others IXs. In case of a failure in the UTC signal, it could become a secondary server. There could be more levels of servers than those one drawn in the figure. Note that there is an IPv6 NTP Primary server.

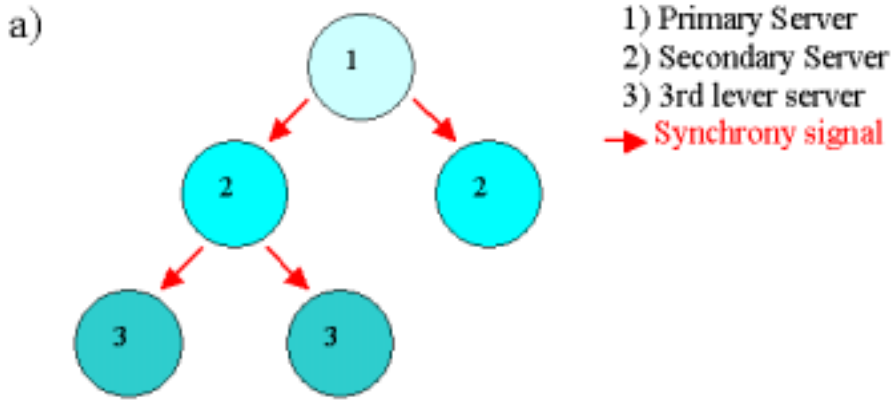

Figure 8-2: Hierarchical Structure for Synchronization

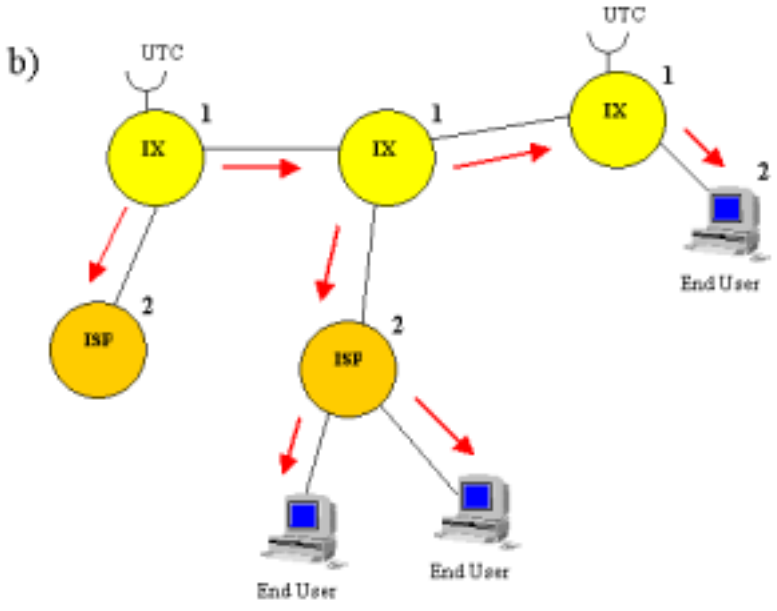

Figure 8-3: NTP Application Service

For more information about NTP, see **[31].**

8.2 Content Deliver Services

8.2.1 HTTP

The Hypertext Transfer Protocol (HTTP), widely used since 1990, is an application-level protocol with lightness and speed characteristics for distributed, collaborative information systems. It is a generic object-oriented protocol, which can be used for many tasks, such as name servers and distributed object management systems, through extension of its request methods.

Secure HTTP (S-HTTP) provides secure communication mechanisms between an HTTP client-server pair in order to enable information transactions for a wide range of applications. S-HTTP provides a flexible protocol that supports multiple operation modes, key management mechanisms, trust models, cryptographic algorithms and encapsulation formats through option negotiation between parties for each transaction. Syntactically, Secure HTTP messages are the same as HTTP, consisting of a request or status line followed by headers and a body. However, the range of headers is different and the bodies are typically cryptographically enhanced.

HTTP is a good example of a service-enabler that if is it installed in an IX the customer can access to the Web sites at high bandwidth rates.

There are several IPv6 HTTP server and clients implementations and the use within Euro6IX IXs could include hypermedia access to several databases related to IX networks. Using the concept of Web-Based Enterprise Management (WBEM), a set of standards designed to deliver an integrated set of management tools for the enterprise, and making use of XML could be enabled to manage network devices, desktop systems, telecom systems and application systems, all from a Web browser. More information is available in **[32]**.

8.2.2 FTP

The File Transfer Protocol (FTP) provides the basic elements of file sharing between users. FTP relies on TCP to create a virtual connection for control information and then creates a separate TCP connection for data transfers.

FTP is already working over IPv6, and currently is not very difficult to find some implementations. During the transition from IPv4 to IPv6 is necessary to consider the use of FTP applications working in both IP versions.

The Trivial File Transfer Protocol (TFTP) uses UDP. TFTP supports file writing and reading; it does not support directory service of user authorization. TFTP servers could be used for image or OS updates in IXs routers and switches. Another interesting application could be applied a TFTP multicast application permitting multiple clients to get files simultaneously from the server using the multicast packets. TFTP multicast option is described in RFC 2090.

FTP and TFTP are another service-enablers that could be installed in an IX in order to enhance the application service performance to customers.

8.3 Network Access Services

8.3.1 AAA

Authentication, Authorization, and Accounting (AAA) concept is used in IP based network management and policy administration. There are several AAA protocols, some of them in standardization process, some of them need an IPv6 implementation. Within the scope of Euro6IX IXs the most interesting AAA protocols are:

8.3.2 RADIUS

The Remote Authentication Dial In User Service (RADIUS) is a protocol for carrying authentication, authorization and configuration information between a Network Access Server, which desires to authenticate its links and a shared Authentication Server.

RADIUS can run over IPv6 and RADIUS attributes can be used to support IPv6 network access. With this tool the IXs could offer dial-up and broadband network access including managed VPN and wireless.

RADIUS Authentication Servers can be set up in a variety of ways, depending upon the security scheme of the network they are serving, but the basic process for authenticating a user is essentially the same.

Using a modem, a remote dial-in user connects to a network access server (NAS) with a built-in analog or digital modem. The NAS then prompts the user for a name and password. For protection against eavesdropping by hackers, the NAS, acting as the RADIUS client, encrypts the password before it sends it to the authentication server. If the primary security server cannot be reached, the security client or NAS device can route the request to an alternate server. When an authentication request is received, the authentication server validates the request, and if the user name and password are correct it enables the necessary procedure to allow the user the access rights to network services and resources. If at any point in this login process all necessary authentication conditions are not met, the security database server sends an authentication reject instruction to the NAS device and the user cannot access the network.

8.3.3 DIAMETER

The DIAMETER base protocol is intended to provide an AAA framework for applications such as network access or IP mobility. DIAMETER is also intended to work both with local AAA and with roaming situations. The IETF AAA working group is specifying the DIAMETER protocol for communication between servers where RADIUS is currently being used. The basic concept behind DIAMETER is to provide a base protocol that can be extended in order to provide AAA services to new access technologies.

A DIAMETER implementation could be used to support IPv6 network access to Euro6IX. For sure both RADIUS and DIAMETER test will be carry out.

8.4 Other Services

8.4.1 SMTP

IETF RFC821 defines the Simple Mail Transfer Protocol (SMTP) that is a mail service modeled on the FTP file transfer service. SMTP transfer mail messages between systems and provides notification regarding incoming mail.

8.4.2 POP3

Post Office Protocol version 3 (POP3) is a protocol used to retrieve e-mail from a mail server. Most e-mail applications use the POP protocol, although some can use the newer IMAP.

8.4.3 IMAP

Internet Message Access Protocol (IMAP) is a protocol for retrieving e-mail messages. The latest version, IMAP4, is similar to POP3 but supports some additional features, as the search through e-mail messages for keywords while the messages are still on mail server. There are SMTP, POP3 and IMAP implementations supporting IPv6.

The e-mail is a powerful tool and Euro6IX IXs will support e-mail via IPv6.

8.4.4 SSH

Secure Shell (SSH) client is a program for logging into a remote machine and for executing commands on a remote machine. SSH provide secure encrypted communications between two un-trusted hosts over an insecure network. X11 connections and arbitrary TCP/IP ports can also be forwarded over the secure channel.

SSH connects and logs into the specified hostname. The user must prove his identity to the remote machine using one of several methods depending on the protocol version used SSH v1 or v2.

Once the server has accepted the user's identity, the server either executes the given command, or logs into the machine and gives the user a normal shell on the remote machine. All communication with the remote command or shell will be automatically encrypted.

The session terminates when the command or shell on the remote machine exits and all X11 and TCP/IP connections have been closed. The exit status of the remote program is returned as the exit status of SSH.

SSH is already running for IPv6 and could be used to access to remote host in order to realize management tasks.

8.4.5 LDAP

LDAP defines a standard method for accessing and updating information in a directory. LDAP runs directly over TCP, and can be used to access a standalone LDAP directory service or to access a directory service that is back-ended by X.500.

A directory is like a database, but tends to contain more descriptive, attribute-based information. Different methods to provide a directory service allow different kinds of information to be stored

in the directory. An LDAP directory is entirely made up of a number of individual records, or entries. Each entry has a number of attributes, which are just name-value pairs, and each attribute may normally be multi-valued. The nitty-gritty details of LDAPv3 are defined in RFC 2251.

With the rise of different networks and applications, the number of specialized directories of information grows too, and it results in isolate centers of information that cannot be shared and are hard to maintain.

If all this information was maintained and accessed in a consistent and controlled manner, it would provide a focal point for integrating a distributed environment into a seamless system. An IPv6 LDAP service could be used to integrate, search and retrieve information relate to both research and operational Euro6IX activities. The IXs centered positions are a natural site for centered information point.

In Figure 8-4, the external client (out of IX) sends messages that specify the operations requested by the client (search, modify, delete, and so on), and then the responses from the centered server (in the IX) retrieve the asked task.

- The client establishes a session with an LDAP server. This is known as “binding to the server”. The client specifies the host name or IP address and TCP/IP port number where the LDAP server is listening. The client can provide a user name and a password to properly authenticate with the server, or the client can establish an anonymous session with default access rights. The client and server can also establish a session that uses stronger security methods, such as encryption of data
- The client then performs operations on directory data. LDAP offers both read and update capabilities. This allows directory information to be managed as well as queried. LDAP supports searching the directory for data meeting arbitrary user-specified criteria. Searching is the most common operation in LDAP. A user can specify what part of the directory to search and what information to return. A search filter that uses Boolean conditions, specifies which directory data matches the search.
- When the client has finished making requests, it closes the session with the server. This is also known as unbinding.

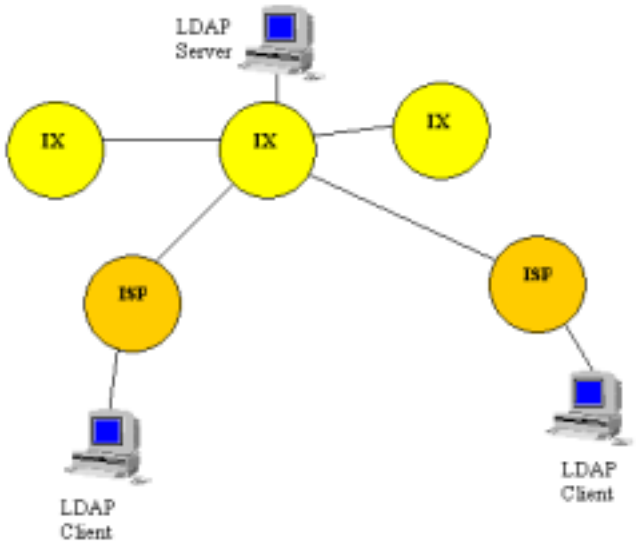

Figure 8-4: LDAP Application Service

8.4.6 RPC

Remote Procedure Call (RPC) is a protocol that a program can use to request a service from a program located in another computer in a network without having to understand network details.

RPC uses the client/server model and can be used for developing distributed applications.

As a regular or local procedure call, an RPC is a synchronous operation requiring the requesting program to be suspended until the results of the remote procedure are returned.

However, other models are possible; for example, the caller may continue processing while waiting for a reply, or the server may dispatch a separate task for each incoming call so that it remains free to receive other messages.

The remote procedure calls differ from local procedure calls in the following ways:

- Use of global variables as the server has no access to the caller program's address space.
- Performance may be affected by the transmission times.
- User authentication may be necessary.
- Location of server must be known.

Consequently, an IX point is a right place for a remote program that is used by an IX customer.

RPC supports IPv6 in a seamless manner since it is supposed to be transport independent. Most RPC applications will run over IPv6 without any change.

Some advanced RPC applications with transport knowledge might require updates.

9. SUMMARY AND CONCLUSIONS

The technical analysis realized inside the A2.1 activity in the first semester of Euro6IX project and summarized in this D2.1 deliverable, has been made in order to individuate the theoretical concepts upon which the Euro6IX backbone should be based.

The analysis has taken into account various issues needed to set-up the network and the further problems that can arise when a so complex structure has to be set-up, individuating at the end the technical options that better meet the requirements.

Moreover, this study, following the technical discussion among the partners, has taken into account the possibility to have two different scenarios regarding the IX model that can be used. So, particularly, in the Layer 3 section, where the impact of a different IX model has bigger, two routing and addressing models have been individuated and proposed.

From this study, it emerged that, not only a well defined set of technical key point to take into account in setting up the network, but also the real possibility to use the network infrastructure to test a large set of application services that can help to put in evidence the innovative services that IPv6 makes available.

10. REFERENCES

[1] Virtual Router Redundancy Protocol. RFC 2338. April 1998.

[2] A Border Gateway Protocol 4 (BGP-4). RFC 1771. March 1995

[3] Multiprotocol Extensions for BGP-4. RFC 2858. June 2000.

[4] History of the RSng Project. See: http://www.rsng.net/history.html

[5] Route Server Technical Overview. See: http://www.rsng.net/overview.html

[6] Representation of IP Routing Policies in a Routing Registry. T. Bates et al. RFC 1786. March 1995.

[7] Routing Policy Specification Language (RPSL). C. Alaettinoglu el al. RFC 2622. June 1999.

[8] 6BONE Registry Database. See: http://www.6bone.net/RIPE-registry.html

[9] F. Parent. RPSL extensions for IPv6 and Multicast Routing Policies. Work in progress. January, 2002. See: http://www.viagenie.qc.ca/en/ipv6/ietf/draft-rpsl-00.txt

[10] Internet Routing Registry. See: http://www.irr.net

[11] RABD Routing Registry. See: http://www.radb.net

[12] Router Arbiter Project Toolset. See: http://www.isi.edu/ra/RAToolSet/

[13] Internet Routing Registry Toolset Project. See:http://www.ripe.net/ripencc/pub-services/db/irrtoolset/index.html

[14] Traceroute.org. See: http://www.traceroute.org

[15] 6BONE Tools Sites. See: http://www.6bone.net/6bone_tools.html

[16] TILAB IPv6 Looking Glass. See: http://net-stats.ipv6.tilab.com/bgp/services.html

[17] University of Oregon Route Views Project. See http://www.routeviews.org

[18] The Route Server Daemon (RSd). See: http://www.isi.edu/ra

[19] GNU Zebra. See: http://www.zebra.org

[20] Multi-threaded Routing Toolkit. See: http://www.merit.edu/mrt

[21] NextHop Technologies GateD releases. See: http://www.gated.org

[22] Internet Routing Registry Daemon. See: http://www.irrd.net

[23] RIPE Database. See: ftp://ftp.ripe.net/ripe/dbase/software

[24] IPv6 Tunnel Broker. RFC 3053.January 2001

[25] Connection of IPv6 Domains via IPv4 clouds. RFC 3056. February 2001

[26] Network Address Translation – Protocol Translation. RFC 2766. February 2000.

[27] Remote Network Monitoring Management Information Base. RFC 1757. February 1995.

[28] A Simple Network Management Protocol. RFC 1157. May 1990

[29] NetFlow technology. See: http://www.cisco.com/warp/public/732/tech/netflow

[30] ASpath-tree. See: http://carmen.ipv6.tilab.com/ipv6/tools/ASpath-tree

[31] NTP. See: http://www.viagenie.qc.ca/en/ipv6/ntpv6

[32] HTTP. See: http://www.dmtf.org/spec