### Abstract:
Deliverable D2.1 describes the first phase of the development of the WINMAN solution. It defines the business model and the requirements. From these, the use cases and the management system architecture for integrated network management of IP connectivity services are derived. It is a comprehensive description of the work performed in WP2 and serves as the basis for the specification and implementation work, as well as integration and testing in WP3 and WP4.

### Keyword list:

* Type: PU-public, PP-limited, RE-restricted, CO-internal
** Nature: P-Prototype, R-Report, D-Demonstrator, O-Other
Executive Summary

This document is a deliverable from the project IST-1999-13305 WINMAN (WDM and IP Network MANagement). It describes the results from Work Package 2 “Management Requirements and Analysis” of the project.

First, Network Evolution Scenarios, IP-based services and the enabling technologies to support the networks and services have been analysed. In detail, a study about the networks, services and protocols that exist in telecommunications networks today, the ones that are now being installed and the ones that will probably dominate in the near future is described. WINMAN focuses on the two latter cases. The business drivers behind inter technology-domain management systems have been investigated.

The main objective of WINMAN is to develop an integrated management system for networks built on various technologies. Evolutionary scenarios were taken into account, such as IP over SDH over WDM, IP over ATM over WDM and IP over ATM over SDH over WDM. From the study of the state of the art and the trends in telecommunications networks, the focus of WINMAN on networks deploying IP directly over WDM has been justified. According to the business drivers identified, the WINMAN business model has been formulated. All the entities interacting with the WINMAN system relevant to the business case, the actors, are presented, and the requirements imposed on the WINMAN solution are elaborated. The functional requirements are categorised according to the TeleManagement Forum models that exist in the area. In detail, the system boundaries, the way that WINMAN should interact with other systems and the functional or non-functional requirements imposed by other systems and users have been specified.

After specifying the requirements, all the actions that WINMAN is expected to perform have been identified. This is expressed by the corresponding use cases, which analyse the behaviour of the system. The interdependencies of use cases are expressed by the use cases diagrams identifying the overall context. Subsequently, the use cases are elaborated per functional category, in the same way the requirements were presented. In addition, basic background principles and policies are introduced to establish a common understanding.

The use cases led to the high-level specification of the WINMAN system. The system domain model and the views of the internal and external interfaces of the system have been introduced, followed by the corresponding scenarios and the detailed architecture of the WINMAN subsystems.

Background material that is useful to understand the way the WINMAN consortium co-operated inside the project to capture and document the requirements is included as annexes to the document.
Table of Contents

Executive Summary

1. Introduction ........................................................................................................................................ 1

2. Analysis of Network Evolution Scenarios and IP Service Characteristics ............................................. 3
   2.1. Network Evolution Scenarios ........................................................................................................... 3
   2.1.1. Introduction .................................................................................................................................... 3
   2.1.2. Market Drivers ............................................................................................................................... 4
   2.1.3. Current Backbone Networks ......................................................................................................... 5
   2.1.4. Trends in Network Architectures .................................................................................................... 7
   2.1.5. Internet 2 ....................................................................................................................................... 10
   2.1.6. Next Generation Internet (NGI) ................................................................................................. 11
   2.2. Services Provided with IP based Networks ......................................................................................... 13
   2.2.1. Introduction .................................................................................................................................... 13
   2.2.2. A QoS framework for IP-based services ....................................................................................... 13
   2.2.3. Virtual Private Networks .............................................................................................................. 20
   2.2.4. Application-oriented services: VoIP and MoIP ............................................................................ 26
   2.3. Enabling Technologies ....................................................................................................................... 34
   2.3.1. Introduction .................................................................................................................................... 34
   2.3.2. End to end QoS in IP networks .................................................................................................... 34
   2.3.3. DiffServ .......................................................................................................................................... 36
   2.3.4. MPLS ............................................................................................................................................. 39
   2.3.5. Optical transport ............................................................................................................................ 44
   2.4. Integration of IP and WDM ................................................................................................................ 54
   2.4.1. IP over ATM over SDH for WDM transmission ............................................................................ 54
   2.4.2. IP over ATM directly on WDM ..................................................................................................... 55
   2.4.3. IP over SDH, Packet over SONET (POS) ...................................................................................... 56
   2.4.4. IP over SDL directly over WDM .................................................................................................. 56
   2.4.5. IP over Gigabit Ethernet for WDM ............................................................................................... 57
   2.5. Business Drivers for Inter Domain Management Systems ................................................................... 58
   2.5.1. Technical Analysis .......................................................................................................................... 58
   2.5.2. Financial Analysis ........................................................................................................................ 71

3. Business Model and System Requirements ............................................................................................... 75
   3.1. The WINMAN Methodology ............................................................................................................. 75
   3.2. WINMAN scope ................................................................................................................................. 76
   3.2.1. The Control Plane vs Management Plane approaches in the IP/WDM Integration ....................... 79
   3.2.2. Control & Management Plane Interaction ...................................................................................... 81
   3.3. Business Reference Model ................................................................................................................ 86
   3.3.1. Business Roles ............................................................................................................................... 86
   3.3.2. Business Model Relationships ...................................................................................................... 87
   3.3.3. Business Case ............................................................................................................................... 88
   3.3.4. Business Processes addressed by WINMAN ................................................................................. 90
   3.3.5. High Level Management Architecture .......................................................................................... 91
   3.4. Requirements Specification ............................................................................................................... 92
   3.4.1. Introduction .................................................................................................................................... 92
   3.4.2. Requirements Attributes .............................................................................................................. 93
   3.4.3. Requirements Identification .......................................................................................................... 95
   3.4.4. Non-functional Requirements .................................................................................................... 110
4. **WINMAN USE CASES (UCs)**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>112</td>
</tr>
<tr>
<td>4.1.1 UCs Actor’s Model</td>
<td>112</td>
</tr>
<tr>
<td>4.1.2 Transport entities terminology adopted</td>
<td>113</td>
</tr>
<tr>
<td>4.2 Identification &amp; Classification of UCs</td>
<td>113</td>
</tr>
<tr>
<td>4.3 UCs Overview</td>
<td>115</td>
</tr>
<tr>
<td>4.3.1 Policies Provision (P)</td>
<td>115</td>
</tr>
<tr>
<td>4.3.2 Scheduling (S)</td>
<td>120</td>
</tr>
<tr>
<td>4.3.3 Network provisioning (NP)</td>
<td>122</td>
</tr>
<tr>
<td>4.3.4 Network Inventory Management (NIM)</td>
<td>134</td>
</tr>
<tr>
<td>4.3.5 Network Maintenance &amp; Restoration (NM&amp;R)</td>
<td>137</td>
</tr>
<tr>
<td>4.3.6 Network Data Management (NDM)</td>
<td>139</td>
</tr>
</tbody>
</table>

5. **System Analysis and Architecture**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 Introduction</td>
<td>140</td>
</tr>
<tr>
<td>5.2 High level architecture</td>
<td>141</td>
</tr>
<tr>
<td>5.2.1 Systems Description</td>
<td>142</td>
</tr>
<tr>
<td>5.3 Domain Model and Network Views</td>
<td>147</td>
</tr>
<tr>
<td>5.3.1 Description</td>
<td>147</td>
</tr>
<tr>
<td>5.3.2 Class diagrams</td>
<td>150</td>
</tr>
<tr>
<td>5.3.3 Internal and External Network Views</td>
<td>153</td>
</tr>
<tr>
<td>5.4 System scenarios</td>
<td>159</td>
</tr>
<tr>
<td>5.4.1 Provisioning and configuration scenarios</td>
<td>159</td>
</tr>
<tr>
<td>5.4.2 Policies scenarios</td>
<td>174</td>
</tr>
<tr>
<td>5.4.3 Network inventory scenarios</td>
<td>178</td>
</tr>
<tr>
<td>5.4.4 Network maintenance and restoration scenarios</td>
<td>187</td>
</tr>
<tr>
<td>5.4.5 Network data management scenarios</td>
<td>207</td>
</tr>
<tr>
<td>5.5 Systems architecture</td>
<td>213</td>
</tr>
<tr>
<td>5.5.1 Reference architecture for NMS</td>
<td>213</td>
</tr>
<tr>
<td>5.5.2 INMS architecture</td>
<td>219</td>
</tr>
<tr>
<td>5.5.3 WDM NMS architecture</td>
<td>223</td>
</tr>
<tr>
<td>5.5.4 IP NMS architecture</td>
<td>226</td>
</tr>
<tr>
<td>5.6 GUI architecture</td>
<td>233</td>
</tr>
<tr>
<td>5.6.1 Overview</td>
<td>233</td>
</tr>
<tr>
<td>5.6.2 Functional Architecture</td>
<td>233</td>
</tr>
<tr>
<td>5.6.3 GUI Functional Components – Graphical view</td>
<td>236</td>
</tr>
</tbody>
</table>

6. **Conclusions**

- References | 240 |
- Abbreviations | 244 |

**Appendix A. Requirements Capture Methodology** | 246 |
**Appendix B. Use Cases Detailed Description** | 248 |
**Appendix C. Component based architectures** | 310 |
1. Introduction

WINMAN main objective is to provide IP connectivity to third party Service Management Systems. WINMAN proposes a novel approach for providing Internet services over the Optical Transport Network by means of management functions. The telecom-style network management approach is extended to the IP layer with the co-operation of MPLS Internet protocol. The appropriate synergy and integration of the two layers is performed with management means capable of performing integrated provisioning of Label Switched Paths over optical paths, as well as integrated multi-layer fault and performance management. This approach is adopted by WINMAN, while in the background the synergy between the control and management plane is being investigated. The proposed solution offers an integrated network management solution for the provisioning of end-to-end IP connectivity services with QoS constraints derived from Service Level Agreements (SLAs).

The overall WINMAN aim is to offer an integrated network management solution, the WINMAN solution, which is capable to provide end-to-end IP connectivity services derived from Service Level Agreements (SLAs). WINMAN will capture the requirements, define and specify an open, distributed, and scalable management architecture for IP connectivity services on hybrid transport networks (ATM, SDH and WDM). The requirements will include and the architecture will support multi-vendor, multi-technology environments and evolution scenarios for end-to-end IP transport from IP / ATM / SDH / WDM towards IP/WDM. WINMAN will consist of optimised architecture and systems for integrated network management of IP connectivity services over hybrid transport networks.

From implementation point of view, the project will address the separate management of IP and WDM networks. Per technology domain the integration of the management at Network Management level will be developed. An Inter-technology domain Network Management System (INMS) as a sub-layer of the Network Management Layer will be implemented to support IP-connectivity spanning different WDM sub-networks and to integrate the management of IP and WDM transport networks. The INMS and the IP and WDM Network Management Systems will implement Configuration, Fault and Performance (CFP) Management application functions. Development will be carried out in two phases in the project resulting in release R0 and R1. WINMAN will set up an infrastructure including several sites and experiment scenarios to demonstrate and validate the system. Results will be disseminated and exploited.

The telecom operators will provide the necessary input for the requirements. The areas to be examined include physical topology resource allocation (network planning), path provisioning functionality (automatically monitoring and optimally fitting the IP/MPLS traffic needs), IP/MPLS and WDM optimum routing interworking (involving wavelength translation and optical cross-connections), path tracing, and capacity management in both the physical and logical domains, fault management and restoration mechanisms including alarm correlation between optical and client layers, alarm localisation and multi-layer survivability, performance management including relation of optical layer performance and the client layer performance, traffic monitoring, and identification of the end-to-end QoS requirements.

Along the road from requirements capture down to the demonstration of the project results, the following milestones are distinguished: Requirements and high-level overall architecture, the system specifications and designs, implemented and tested systems integrated in the experimentation sites, and the solutions validated in a real-world environment. This sequence applies to both releases. Note that the term domain in this project, does not refer to administrative domains, but to technology domains, like IP and WDM.
WINMAN focuses on IP directly over WDM networks. Nevertheless, some evolution scenarios were taken into account, such as IP over SDH over WDM, IP over ATM over WDM and IP over ATM over SDH over WDM. The first step we took was to make a study of the actual state of the art in telecommunications networks. Next we took care of the definition of the business model and the requirements. The following step was the definition of the use cases. The use cases lead to the high-level specification of the WINMAN system, and finally we reached the detailed architecture of the WINMAN subsystems. This is the limit of the scope of this Work package.
2. Analysis of Network Evolution Scenarios and IP Service Characteristics

2.1. Network Evolution Scenarios.

2.1.1. Introduction

Today telecommunications market is highly driven by IP-oriented applications and technologies. This is a completely different situation in respect of what happened in the past when almost all the traffic of telecommunications networks was voice information from telephone calls. The fantastic growth of the Internet made the volume of data traffic close to the volume of voice traffic and the technology that was foreseen only for specific data communication is being pushed to be of widespread use not only for data but also for services like voice and video, traditionally supported in circuit switched networks. The move by telecom operators' corporate customers into the IP world, and the need for interoperability between private and public networks, drives the telcos to adopt IP in their core networks as a means of unifying traffic types that are compatible with their customers' networks. Nevertheless, the above mentioned and other final services require quality of service and the data networks conceived for IP data applications are essentially best effort transport networks. Very high efforts are taking place on standard organisations and equipment vendors to give data networks the means to support that kind of traffic with quality of service requirements. But quality of service is not the only problem to solve in IP networks. In fact, the volume of traffic has grown so much that it is necessary to look for more transport capacity. This leads to the use of WDM to exploit the bandwidth of the optical support media. Therefore it seems apparent that IP and WDM and advanced QoS techniques need to interact to fully exploit the network resources and at the same time providing services with the required levels of quality.

Although IP and the WDM technologies will be dominant, they will be introduced gradually in an evolutionary process starting from present days situation. Therefore is worthy to look at what the transport network is today to derive evolutionary scenarios to the target goal. The dominant transport technology for most wide area packet-switched networks for the last five years has been the Asynchronous Transfer Mode (ATM). Nevertheless, ATM was not accepted in the LAN environment, where 100BaseT Ethernet was significantly cheaper, easier and, more important, supporting thousand of IP applications running on top. In respect of long haul systems the SONET/SDH transmission protocol was the evolution of PDH networks based in TDM technology. The outcome was that, based on the different needs regarding the services offered to the users and the corresponding traffic type, there was a balance mainly between the three different protocols, IP, ATM and SDH, each one with its pros and cons according to the different service types. The coexistence of these transport mechanisms has brought the need of their integration not only in the transport plane but also at the control and management planes. The need of this integration has also forced the appearance of new technologies. Among them we can mention Packet over Sonet (POS), which is capable of transmitting IP packets through SDH frames in all SDH rates.

Today’s networks are following an evolutionary path towards Optical Transport Networking. The type of network can be seen as organised in four layers: the Backbone Network, the Metropolitan Core Network, the Access DWDM Network and the Distribution/Customer Premises Network. The first layer is the WAN Backbone Network, the so-called “long haul” network. This layer should provide the intercity Dense WDM or ultra-Dense WDM backbone network and will normally connect central offices located in different cities. The next layer is the Metropolitan DWDM Network, also called “Metro” or Regional network. The Metropolitan WDM equipment will be organised in OADM-based shared protection rings, depending on the service provider’s desired degree of bandwidth “overbuild” and survivability time-scale requirements. The Access WDM network can be seen as the Metropolitan WDM network, but it is adapted for smaller number of wavelengths. Its role is to distribute traffic to the Access and Customer Premises Network.

The above-mentioned scenario is under the point of view of formerly telecommunication carriers. But at the same time, the Internet community is also trying to bring solutions to actual customers needs. In this respect is worthy to mention two initiatives, namely Internet 2 and the so-called Next Generation Internet. Internet2 is a
consortium being led by over 180 universities working in partnership with industry and government to develop and deploy advanced network applications and technologies, accelerating the creation of tomorrow's Internet. Internet2 and its members are developing and testing new technologies, such as IPv6, multicasting and quality of service (QoS) that will enable revolutionary Internet applications The Next Generation Internet (NGI) Initiative is a U.S. program designed to fund and co-ordinate federal agencies and academia to design and build the next generation of Internet services. The goal of the NGI initiative is to conduct R&D in advanced networking technologies, to demonstrate those technologies in testbeds that are 100 to 1,000 times faster than today's Internet. The NGI will rely on Internet2 for advanced campus-based, local-area, and select regional network infrastructure

2.1.2. Market Drivers

During many years almost all the traffic of telecommunications networks was voice data from telephone calls. That fact leads to the deployment of technologies that fulfill the needs of this kind of traffic, like SONET/SDH and TDM, on service provider's backbones. These networks were based on circuit switching and tried to achieve constant bit rates and short delays.

On the other hand we had data communications. Appropriate technologies to this kind of traffic were developed and deployed. These technologies were designed to support best effort networks. They were based on statistical multiplexing and particularly good on conducting data bursts. The best examples of this kind are local area networks like Ethernet and Token Ring.

The volume of data traffic was much smaller than the volume of voice traffic. In many cases the majority of the data traffic generated was intended to be delivered within the same LAN or building. For this reason the service providers built their networks for voice traffic. For data traffic there was no other solution than to be adapted to fit in voice networks. This adaptation lead to bandwidth misuse.

Today the situation has changed. The fantastic growth of the Internet made the volume of data traffic close to the volume of voice traffic. If the growth persists (as everyone thinks it will) in the close future data communications will be dominant. This fact changes the requisites of communication networks. Everybody is starting to believe that the model used today (voice communications first) may no longer be correct.

On the other side, the evolution of the electronic terminal equipment (powerful PCs and multimedia applications) changed the requisites of data networks. Today's applications demand quality of service to support voice and video. In fact the traffic that some of the applications that we pretend to use on data networks generate is very similar (or equal) to the traffic on telephone networks. Very high efforts are taking place on standard organisations and equipment vendors to give data networks the means to support that kind of traffic. If they succeed we will soon have IP (data) networks capable of supporting telephone calls and TV broadcast.

If IP traffic is dominant in communication networks and if it is capable of transmitting the same kind of traffic that voice networks do, than we can foresee that we are heading to the implementation of IP in all communication networks.

One of the problems we are facing now, as a result of the tremendous increase of traffic in data networks, is the lack of bandwidth. The best effort to solve this problem is the development of WDM. WDM allows taking advantage of all the capacity of fibre optic links. By now IP over WDM equipment allows only point-to-point connections and usually with SDH between, but in a not very far future we expect much more.

Everything points to a future were communication networks will be optic, with WDM taking full advantage of all the fibre's capacity. Maybe we will get to fast optic routers and switches, eliminating the electronic bottleneck. Maybe we can get fibre closer to the home in a better interface for data communications than the actual ISDN and PSTN, granting at the same time a proper medium for voice communications.

In all of these new things there is a factor that can not be forgotten: the majority of today’s communication networks are made for voice communications, and we can not send them away and bring new IP over WDM networks as soon as we have the technology to do it. We will have to support old networks and provide means for the coexistence with the new ones. This means that service providers will have to manage both technologies
at the same time in their networks. This can mean more management software and more complexity to their system. The WDM solution has built in means to solve part of this problem. We can use a few lambdas to transport old network’s traffic in a completely transparent way. But the problem of management software persists.

It’s right here that the WINMAN solution fits. WINMAN will create a universal management solution for IP networks. It will support management of IP over ATM, IP over SDH and IP directly over WDM, hiding the details to the network manager. The network manager will be able to configure network equipment, create VPNs, drive traffic away from congested links and many other operations, always with the same software tool and without having to worry about what network he is operating.

The WINMAN is a solution designed to be able to follow the evolution of communication networks.

2.1.3. Current Backbone Networks

The dominant transport technology for most wide area packet-switched networks for the last five years was the Asynchronous Transfer Mode (ATM). ATM combining transmission, switching and multiplexing functions based on fixed-length cells appeared a promising solution not only for core networks, but also for access networks. The characteristic of being asynchronous as opposed to the Synchronous Transfer Mode (STM) of TDM multiplexing was considered ideal for Internet-type data networks, whose traffic comes most of the times in bursts. In addition, the characteristic of being connection-oriented matched also the concept of traditional circuit-switched based voice calls. Other features such as the sophisticated traffic control and management, the support of user-defined QoS classes and the use of optical fibres for the transportation of the cell-based traffic, presented ATM as the all-in-one network solution.

Nevertheless, ATM was not accepted in the LAN environment, where 100BaseT Ethernet was significantly cheaper, easier and more important supporting thousand of IP applications running on top. On the other hand, neither ATM PC cards nor native ATM applications did reach the real end-users and were only part of research projects. The inter-working of IP with ATM gave the opportunity to use the “beloved” user-friendly IP applications, but pushed ATM to the WAN networks, most of the times not even taking into account the ATM QoS classes, sticking to the usual “best-effort” IP service. The continuous growing demand for IP applications and the support for real-time and non-real time multimedia applications had as an immediate effect the demand for higher IP bandwidth. In addition the convergence of traditional voice-carrying circuit-switched networks with packet-switched data networks through the use of protocols such as VoIP foretell that packet-switched data traffic will soon surpass the traditional voice traffic in the upcoming years.

On the other hand, over the last 10 years, there has been an explosive growth in the number of Internet users, which have been expanded from the original focus on academic oriented use to the development of new services used by the general public and commercial organisations. The development of the World Wide Web was the major force for the Internet explosion. Internet is in its 11th year of doubling its size since 1988, as illustrated in the Figure 2.1. There are over 44 million hosts on the Internet and estimated 140 million users world-wide.
As far as the synchronous transfer mode, the SONET/SDH transmission protocol was the evolution of PDH networks based on TDM technology. All incumbent telecom operators currently operate SDH networks. SDH technology has significant advantages supporting high and variable transport rates, sophisticated management, automatic protection techniques and easy path provisioning schemes using add/drop capabilities. SDH succeeded in satisfying the needs and growing demand for better quality mass POTS service and ISDN telephony. In parallel as a result of the smooth inter-working with the ATM protocol, SDH captured part of the ATM or other type “data” traffic into its network.

PDH, beyond the lower bit rates, lacks the significant advantages of SDH and thus suffers from the lack of network management functions, easy end-to-end path provisioning and automatic recovery, which makes it expensive, non-flexible and inconvenient.

The outcome was that, based on the different needs regarding the services offered to the users and the corresponding traffic type, there was a balance mainly between the three different protocols, IP, ATM and SDH, each one with its pros and cons according to the different service types. IP dominated the LAN area, where all the applications feed the users. SDH was used in the WAN area and specifically the core network to multiplex mainly traditional telephony traffic, while ATM was again used in the WAN to switch “data-type” traffic. In most cases ATM worked on top of a physical layer, being SDH where traffic demands were higher than the least SDH traffic rate, STM-1 or 155Mbps, or being PDH for lower traffic rate demands. Concluding, according to what was mentioned above, the integration of multiple layers was soon a reality forming stacks such as IP over ATM over SDH (IP/ATM/SDH) or IP over ATM over PDH (IP/ATM/PDH). In other words, most wide area traffic, including IP, is converted to ATM cells and then transported over SDH paths. This usually occurs because IP traffic needs to be multiplexed with ATM traffic or other TDM traffic to make it cost-effective to send.

The rapid growth of the Internet market opened up many new market opportunities, and this was the main reason why ATM was squeezed between the IP and the SDH. IP is the driving force and the demand for Internet bandwidth is increasing rapidly, reaching traffic rates of Gigabit per second for WAN environments. ATM technology reaches its limits around 2 Gbps, because of the inner Segmentation and Re-assembly (SAR) process of the ATM Adaptation Layer. AAL is responsible of segmenting and bringing together higher layer frames. In addition ATM brings an extra overhead to IP being encapsulated using AAL5 — the infamous cell tax, which is around 17%. In other words the total real capacity of ATM virtual paths/channel available for real IP traffic is being taxed by a significant percentage, visible especially in high transmission rates.

New technologies have already emerged being capable of transmitting IP directly over fibre using SDH framing. Such a protocol is Packet over Sonet/SDH (POS), which is capable of transmitting IP packets through SDH frames in all SDH rates STM-1, STM-4 (622Mbps), STM-16 (2.5 Gbps) and the maximum of STM-64 (10Gbps). In other words, Gigabit IP routers could be equipped with SDH line interface cards and can be connected either directly with each other (POS over dark fibre), or through a usual SDH network (POS over SDH). The overhead of POS is significant lower for IP traffic (around 6%), but such interfaces are very expensive at the time being. In any case the ATM layer is missing resulting in the direct IP over SDH (IP/SDH) approach.
One step further is the scenario of POS over DWDM, where mainly high-speed interfaces (such as STM-16 or STM-64) could let routers and switches connect directly into a DWDM system without going through the SDH network. So there will be cases where data and voice traffic are multiplexed by SDH before going over DWDM, and cases in which IP and ATM data connect directly to DWDM without any interference from SDH.

The above are the main reasons why ATM is being thrown away from the core part of the WAN networks and pushed only to the access part of the WAN, where it seems that will stay for some years. ATM capable of producing Virtual Paths/Channels with a determined Peak Cell Rate can serve different “low” broadband rates, so as to aggregate the traffic to the SDH backbone.

The move by telecom operators’ corporate customers into the IP world, and the need for interoperability between private and public networks, drives the telcos to adopt IP in their core networks as a means of unifying traffic types that are compatible with their customers' networks. Gigabit Ethernet is the new dominant fast-switching IP enabling technology for the LAN and moreover for the MAN environment, putting aside the usual framing transmission techniques like SDH. At least for distances less than 100km, Gigabit Ethernet starts dominating the converged LAN and MAN data networks using only WDM or dense WDM equipment as transport technology without any framing, resulting in IP over GigE over WDM (IP/GigE/WDM). Gigabit Ethernet will reach traffic rates up to 10 Gbps, named for this reason 10-Gigabit Ethernet.

WDM technology started being used mainly between point-to-point SDH WAN links multiplying the capacity that a single fibre can transmit, by using more than one wavelength in a single fibre. Dense Wave Division Multiplexing steps in nicely here by lowering the cost of necessary equipment and by solving the bandwidth bottleneck reached by the TDM limits. Besides the above mentioned “long-haul” WDM, WDM or DWDM equipment are now being employed into “medium-haul” or metropolitan areas as well, called “metro WDM”, and will gradually throw aside the SDH equipment, mainly in the access part, feeding the WDM equipment with traffic.

In a couple of years WDM equipment will form the all-Optical Transport Network in the backbone, functioning as an optical pipe multiplexing different types of signals such as Gigabit Ethernet, ATM and SDH. This will be the case sooner for Competitive Telecom Operators, while Incumbent Telecom Operators have significantly invested in SDH equipment and there will be a longer period of time having both, sometimes overlay networks, so that their investments are paid-off.

As far as the integration of IP with WDM, there will be a period of time where there is no direct interaction between the two layers, named the overlay approach or model. Integration of IP layer with WDM layer is still under research with approaches like MP-S and DiffServ over DWDM and will be reported in detail hereafter.

### 2.1.4. Trends in Network Architectures

Optical Transport Networking (OTN) represents a natural next step in the evolution of transport networking. As an evolutionary result, optical transport networks will follow many of the same high-level architectures as followed by SONET/SDH, i.e. optical networks remain connection-oriented, multiplexed networks. The major differences derive from the form of multiplexing technology used: TDM for SONET/SDH vs. wavelength division for OTN. To satisfy the short-term need for capacity gain, large-scale deployment of WDM point-to-point line systems will continue. As the number of wavelengths grows, and as the distance between terminals grows, there will be an increasing need to add or drop wavelengths at intermediate sites. Hence, flexible, reconfigurable OADMs will become an integral part of WDM networks. As more wavelengths become deployed in carrier networks, there will be an increasing demand to manage capacity. In much the same way that digital cross-connects emerged to manage capacity into the electrical layer, Optical cross-connects (OXC)s will emerge to manage capacity at the optical layer.

Figure 2.2 depicts an OTN for core, metro, and high-capacity access applications. Initially the need for optical-layer bandwidth management will be most acute in the core environment. The logical mesh-based connectivity found in the core will be supported by way of physical topologies, including OADM-based shared protection-rings, and OXC-based mesh restoration architectures. As bandwidth requirements emerge for the metropolitan and access environments, the OADMs will be used there too.
2.1.4.1 Access

Today the copper-pair-based star-like structure is still dominant in most areas. Cable modem and xDSL technologies are widely used. However, fibre is penetrating into access areas. The point-to-point optical fibre star structure is preferred for business customers with critical security requirements. In major cities, fibre already connects most big business offices (FTTO) and some residential buildings with ring or star structures. Fibre is getting closer to small business customers and residential customers with double star or tree-branch structures. Fibre to the cabinet (FTTCab) and fibre to the curb (FTTC) are getting more common, also fibre to the town (FTTT) and fibre to the village (FTTV) are increasingly popular.

The dream of fibre to the home (FTTH) or desktop is yet to materialise, mainly because of the cost-sensitive nature of this part of the network. In the near future, residential access may remain copper-based, using technologies such as ADSL to boost the capacity of traditional copper lines. However, for business offices, optical technology will be used to bring bandwidth to the end-user. Currently, a lot of Fibre To The Business (FTTB) networks are being deployed involving ATM and SDH access equipment at customer premises. The next step is to use WDM technology for these applications. WDM will first be used in industrial and campus LAN environments. The DWDM network at the Microsoft headquarters in Redmond is a good example of a trial of these latest technologies, which use DWDM in the enterprise environment. We can choose to call this ‘Lightwave to the Business’ (LTTB) and ‘Lightwave to the Desktop’ (LTTD). The reason why this will become technically and economically feasible is the very large number of wavelengths that a single fibre can carry, thus spreading the cost to more subscribers. Introducing more wavelengths per fibre can also lead to new topologies for home access by using ring or bus like structures with an add/drop port per home so that each home has its own wavelength.

2.1.4.2 Metro

In the optical core network, transmission at rates of 10 Gbps and up per wavelength are providing access to terabits of bandwidth between the metropolitan areas. At the access side of the metropolitan network, Fast Ethernet is becoming commonplace. The metropolitan network should extend the transparency and the
scalability of the LAN through to the optical core network. Metropolitan networks can take on this challenge using a hybrid TDM/DWDM network architecture. Both ATM and IP have been brought to market, however neither has dominated the metropolitan network. ATM as well as IP could be the transport technology on top of the hybrid TDM layer. Other trends in the metropolitan network are the introduction of hybrid add/drop multiplexers, TDM MultiPort Multiplexers and cross-connects. These will enable transparent optical layer services and a flexible bandwidth expansion.

### 2.1.4.3 Core

Advances in optical fibres are making the available transmission window much wider and it is expected that fibres will be able to carry more than a thousand wavelengths in the future. Historically, the wavelength region between 1350 and 1450 nm has not been used because of the high fibre attenuation over much of this region caused by the presence of hydroxyl (OH-) ions. A residual impurity from the fibre manufacturing process, these ions cause an absorption peak near 1385 nm. Recently a breakthrough was announced in fibre for metropolitan applications, which eliminates the water absorption peak in the fibre’s attenuation curve and thus makes an additional ‘window’ available for transmission. In effect, this makes the entire spectrum from 1300 nm to 1600 nm available for transmission. This much broader available window leads to a new concept, which we can call ‘Broad Wavelength Division Multiplexing (BWDM)’. Particularly for short-distance applications, where the need is for diverse signals at multiple bit rates, the signals can be transmitted using wavelengths that are not so closely spaced as in the ultra-dense wavelength division multiplexing used in long-distance transmission. This will make it possible to cost-effectively support various services such as high-performance computing, videoconferencing, broadband access (including wireless), multimedia and Internet.

Figure 2.3 is another approach for the future network architecture, which follows the evolutionary path towards Optical Transport Networking. The network is organised in four layers: the Backbone Network, the Metropolitan Core Network, the Access DWDM Network and the Distribution/Customer Premises Network.

The first layer is the WAN Backbone Network, the so-called “long haul” network. This layer provides the intercity Dense WDM or ultra-Dense WDM backbone network and will normally connect central offices located in different cities. It consists of point-to-point DWDM connections and/or STM-64/STM-16 rings. Optical networking products that can be placed in this layer are Optical Line Systems (OLS) (e.g. the WaveStar OLS 400G or the WaveStar OLS 80G by Lucent Technologies) and Terabit Routers/Switches (e.g. the NX64000/NX32000 Router or the GX550 Multiservice WAN Switch). Apart from DWDM links, the OLS may be connected with other cities utilising SDH links.

The next layer is the Metropolitan DWDM Network, also called “Metro” or Regional network. The Metropolitan WDM equipment will be organised in OADM-based shared protection rings, depending on the service provider’s desired degree of bandwidth “overbuild” and survivability time-scale requirements. It may be connected either directly to an OLS or to a terabit router performing traffic routing at optical (wavelength) or IP layer accordingly. The Metropolitan DWDM systems deliver a quite large number of wavelengths and distribute channels of traffic to more nodes located in the city region. For example the Lucent WaveStar AllMetro system is capable of delivering up to 40 wavelengths of 2.5Gbps per single fibre and distributing 2,4, 20 or 40 channels of the traffic to up to 10 nodes.
The Access WDM network is primarily the same as the Metropolitan WDM network, but it is adapted for smaller number of wavelengths. Its role is to distribute traffic to the Access and Customer Premises Network. Various systems can be attached to the Access Ring: xDSL DSLAMs, SDH fibre rings, SDH direct links, ONU etc. In all cases the Edge Device can be either connected directly to the Optical Terminal, or a Core Switch can be utilised to concentrate the traffic. For example a Lucent GX550 or CBX500 switch with IP Navigator can route IP traffic between the metropolitan DWDM network and the access network.

The enterprise network many alternatives can be considered. One example is to use an ATM Switch (e.g. CBX500 with IP Navigator) or a Gigabit Router (e.g. Cajun 880/550) as edge devices. The access interface can vary from 25Mbps to 155Mbps or even to 625Mbps. At the SOHO access, traffic will be normally distributed via an xDSL (e.g. SDSL, ADSL, VDSL) interface from a modem or a residential gateway.

2.1.5. Internet 2

Internet2 [I2], [I2-NEWS], [I2-EL], [I2-IK] is a consortium being led by over 180 universities working in partnership with industry and government to develop and deploy advanced network applications and technologies, accelerating the creation of tomorrow’s Internet. Internet2 is recreating the partnership among academia, industry and government that fostered today’s Internet in its infancy. Internet2 is a not-for-profit research and development consortium and does not have publicly traded stock. The primary goals of Internet2 are to:

- Create a leading edge network capability for the national research community
- Enable revolutionary Internet applications
- Ensure the rapid transfer of new network services and applications to the broader Internet community.

Through Internet 2 working groups and initiatives, Internet2 members are collaborating on:

- Advanced Applications
- Middleware
- New Networking Capabilities
- Advanced Network Infrastructures
- Partnerships and alliances

Internet2 is not a separate physical network and will not replace the Internet. Internet2 brings together institutions and resources from academia, industry and government to develop new technologies and capabilities that can then be deployed in the global Internet. Close collaboration with Internet2 corporate members will ensure that new applications and technologies are rapidly deployed throughout the Internet. Just as email and the World Wide Web are legacies of earlier investments in academic and federal research networks, the legacy of Internet2 will be to expand the possibilities of the broader Internet.

Internet2 and its members are developing and testing new technologies, such as IPv6, multicasting and quality of service (QoS) that will enable revolutionary Internet applications. However, these applications require
performance not possible on today's Internet. More than a faster Web or email, these new technologies will enable completely new applications such as digital libraries, virtual laboratories, distance-independent learning and tele-immersion. A primary goal of Internet2 is to ensure the transfer of new network technology and applications to the broader education and networking communities.

It is expected that the capabilities needed to use new technologies and applications being tested and developed by Internet2 and its members to be built into upcoming generations of commercial products. Internet2 corporate partners are working closely with the Internet2 community to expand the capabilities of their products and services as well as the global Internet. For example, just as most personal computers sold today include the ability to use the Internet, tomorrow’s commercial products will include the ability to use advanced networking capabilities.

The university led Internet2 and the federally led NGI are parallel and complementary initiatives based in the United States. Internet2 and NGI are already working together in many areas. For example, through participation in a NSF NGI program, over 150 Internet2 universities have received competitively awarded grants to support connections to advanced backbone networks such as Abilene and the very high performance Backbone Network Service (vBNS). Internet2 is also forming partnerships with similar advanced networking initiatives around the world. Working together will help ensure a cohesive and interoperable advanced networking infrastructure for research and education, and the continued interoperability of the global Internet.

University research and education missions increasingly require the collaboration of personnel and hardware located at campuses throughout the country in ways not possible over today's Internet. Moreover, universities are a principal source of both the demand for advanced networking technologies and the talent needed to implement them. Researchers, instructors and students at Internet2 universities are able to explore capabilities beyond today's Internet as they teach and learn and conduct science in disciplines ranging from the fine arts to physics.

Participation in Internet2 is open to any university that commits to providing on-campus facilities that will allow advanced applications development. The investment this requires may be more than many institutions can manage right now. However, Internet2 also supports collaboration by Internet2 universities with non-member institutions. Fifteen years ago, connecting to the Internet could be as expensive as participating in Internet2 today. As the technology dropped in price, the entire academic community benefited from the efforts of the initial research participants. Deployment of Internet2 technology will follow a similar pattern.

A key goal of Internet2 is to accelerate the diffusion of advanced Internet technology, in particular into the commercial sector. Internet2 will benefit non-university members of the educational community as well, especially K-12 and public libraries. Internet2 and its members aim to share their expertise with as wide a range of computer users as possible. This approach characterised the first Internet and it can work again today.

### 2.1.6. Next Generation Internet (NGI)

The Next Generation Internet (NGI) Initiative is a U.S. program designed to fund and co-ordinate federal agencies and academia to design and build the next generation of Internet services. The program was first proposed by President Clinton in 1996 and has not yet been formally specified or funded by Congress. It is still unclear how the NGI Initiative will complement other initiatives, such as the NSF's very high-speed Backbone Network Service (vNBS) and Internet2 (I2). The R&D needed to address this is beyond the scope of any single institution, company, or industry. The Next Generation Internet (NGI) initiative, with its broad agenda and ability to involve government, research institutions, and the business sector, is a timely program that will address these challenges. The goal of the NGI initiative is to conduct R&D in advanced networking technologies, to demonstrate those technologies in testbeds that are 100 to 1,000 times faster than today's Internet, and to develop and demonstration on those testbeds revolutionary applications that meet important national needs and that cannot be achieved with today's Internet.

Through its NGI initiative, the US Government will help create an environment in which advanced networking R&D breakthroughs are possible. The NGI initiative, together with other investment sectors shown in Figure 2.4, are expected to create the foundation for the networks of the 21st century, setting the stage for networks that are much more powerful and versatile than the current Internet. The NGI will foster partnerships among
academia, industry, and governments (Federal, state, local, and foreign) that will keep the U.S. at the cutting-edge of information and communications technologies. The NGI will also stimulate the introduction of new multimedia services in our homes, schools, and businesses as the technologies and architectures designed and developed as part of the NGI are incorporated into products and services that are subsequently made available to the general public. The NGI program is essential to sustain U.S. technological leadership in computing and communications and enhance U.S. economic competitiveness.

The NGI initiative is part of a highly successful ongoing multi-agency R&D program. It is a key component of the activities of the Large Scale Networking (LSN) Working Group of the Subcommittee on Computing, Information, and Communications (CIC) R&D. This Subcommittee reports to the Committee on Technology of the White House National Science and Technology Council.

The fiscal year (FY) 1998 LSN budget crosscut was $288.3 million, which included $100 million for the NGI initiative. Congressional FY 1998 NGI appropriations are $85 million, with the Defence Advanced Research Projects Agency (DARPA), National Science Foundation (NSF), National Aeronautics and Space Administration (NASA), National Institute of Standards and Technology (NIST), and National Library of Medicine (NLM) as the participating agencies. The Administration plans to propose $109 million for the NGI in each of FY 1999 and FY 2000 with DARPA, NSF, NASA, NIST, NLM, and the Department of Energy (DoE) as the participating agencies.

The NGI initiative has three goals:

- To advance research, development, and experimentation in the next generation of networking technologies to add functionality and improve performance.
- To develop a Next Generation Internet testbed, emphasising end-to-end performance, to support networking research and demonstrate new networking technologies. This testbed will connect at least 100 NGI sites -- universities, Federal research institutions, and other research partners -- at speeds 100 times faster than today's Internet, and will connect on the order of 10 sites at speeds 1,000 times faster than the current Internet.
- To develop and demonstrate revolutionary applications that meet important national goals and missions and that rely on the advances made in goals 1 and 2. These applications are not possible on today's Internet.

The NGI initiative is closely related to Internet2, a collaborative effort by more than 100 U.S. research universities to create and sustain a leading edge network for developing network engineering and management tools and broadband applications for advanced research and education. The NGI will rely on Internet2 for...
advanced campus-based, local-area, and select regional network infrastructure. The initiative also will rely on substantial matching funds from its private sector partners.

2.2. Services Provided with IP based Networks.

2.2.1. Introduction

This chapter is intended to provide an overview on the most relevant services provided by today and near future IP networks. Understanding the services to be provided is fundamental in order to identify appropriate requirements in any of the design planes we were faced. We have selected for this purpose Virtual Private Networks services (VPNs), Voice over IP (VoIP) and Multimedia over IP services (MoIP). These are different services that impose different requirements on the network. Most of these requirements being expressed in terms of quality of service. Therefore, we start giving an overview of the QoS specification in IP networks to continue with a description of the relevant characteristics of the above-mentioned services.

Talking on QoS means to deal with how it is specified provisioned and assured. The specification of QoS entails to give values to system intrinsic parameters like delay, jitter, losses, etc as well as to operational parameters like time to repair, time to restore service, etc. Different services occupy different positions in planes specified by pairs of the above-mentioned parameters. The values adopted by all these parameters are derived from the conditions specified in the contract subscribed between the customer and the service provider. This contract is called Service Level Agreement. All these topics are dealt in the first section of this chapter.

Section 2.2.3 is devoted to VPNs, a service with a great impact on IP over WDM networks. Once the concept is established we review the most important requirements where to base the design of VPNs. The specific values adopted by the design parameters are indeed addressed to fulfil the SLA. Therefore we take again the issue of SLAs but now under the point of view of its specification in VPN contexts. The last section of the chapter is devoted to present a few but representative examples of today existing commercial products. Two of these products are network based and the other two are customer premises based. All of them have as a common goal to simplify the provisioning and maintenance of VPNs.

The last section of this chapter is devoted to present two categories of services, which are called to be also driving factors in the evolution of the network. For their inherent similarities we treat both VoIP and MoIP together. Once again we start talking about the SLA as the driving factor to specify the service. An example of SLA for VoIP is also provided. The section continues with a presentation of network architecture to provide such type of services. The emphasis is now on the specific subsystems needed for that purpose. Finally, we conclude with a presentation of the most relevant standards. ITU-T, IETF, and ETSI standards are summarised.

2.2.2. A QoS framework for IP-based services

ITU-T E.800 [CAVEN] provides 'Terms and Definitions framework' for QoS concept, relating quality of service and network performance and a set of performance measures. Amongst many hundreds of terminology definitions, QoS is defined as: "The collective effect of service performance which determines the degree of satisfaction of a user of the service".

QoS means that network that transports traffic from different application must be able to provide better service to selected applications. To be able to support the management of QoS, IP networks must incorporate capabilities to carry out the correct treatment of IP packets, together with appropriate monitoring of network performance and service metrics. In a multi-layered network, as WINMAN deals with, providing end-to-end QoS requires the collaboration of all layers from top to bottom.

In order to properly manage QoS, networks must provide capabilities for:

- **Specifying QoS.** This mechanics must allow the users or the applications to request the level of quality that they need.
• **Provisioning QoS.** The network must provide the connections with the requested QoS.

• **Assuring QoS.** The network has to guarantee the QoS is maintained during the time the service is active.

QoS manages bandwidth according to application demands and network management configuration in order to distinguish traffic with strict timing requirements from that which can tolerate delay, jitter and packet-loss. Since bandwidth is a finite resource, QoS must guarantee that, although dedicating more resources to high-priority applications, low-priority ones will still function.

### 2.2.2.1 Specifying QoS

#### 2.2.2.1.1 QoS parameters

In order to provide IP connectivity services, applications must be able to request the network for the quality of the connections they need. QoS represents the level of service that a network offers to its users and can be measured as a number of quantifiable parameters [ISO].

Usually, QoS parameters can be divided in two blocks [TMF]:

• **Service intrinsic parameters.** These parameters characterise the service in terms of network performance and reflect the QoS provided by the network.

• **Operational parameters.** These parameters are related to the performance of an organisation and reflect the quality of the operational processes.

The most important service intrinsic parameters used to specify QoS in IP networks are:

• **Delay.** Delay is the time required for a signal to traverse the network. In a network context, end-to-end delay is the time required for a signal generated at one network end reaches the other end. There are several sources of delay that affect the transmission of information over IP networks:
  - **Accumulation delay:** It is caused by the need to collect a complete frame to be transmitted by the network. For example, for voice over IP (VoIP), accumulation delay is produced by the need to produce a complete frame of voice samples to be processed by the voice coders.
  - **Processing delay:** The process of encoding and collecting the encoded samples into a packet for transmission over the IP network causes it.
  - **Network delay:** It is caused by the physical medium and protocols used to transmit the data and by the buffers used to remove packet jitter on the receiver side.

The overall end-to-end delay is the sum of all these delays.

• **Jitter.** Jitter refers to a distortion of inter-packet arrival times compared to the inter-packet times of the original transmission. Removing jitter requires collecting packets and holding them long enough to allow the slowest packets to arrive in time to be placed in the correct sequence. This causes additional delay. The two conflicting goals of minimising delay and removing jitter have originated various schemes to adapt the jitter buffer size to match the time-varying requirements of network jitter removal. This adaptation has the explicit goal of minimising the size and delay of the jitter buffer, while at the same time preventing buffer underflow caused by jitter.

• **Packet loss.** In current IP networks, all information frames are treated like data. Under peak loads and congestion, frames will be dropped regardless of the supported service. The data frames, however, are not time sensitive, and dropped packets can be appropriately corrected through the process of retransmission,
whereas voice packets can not be dealt with in this manner, causing induced distortion in the original analog signal at the receiver's end.

- **Bandwidth.** Bandwidth is the maximum data transfer rate that can be sustained between two end points. It should be noted that this is limited not only by the physical infrastructure of the traffic path within the transport networks, which provides an upper bound to available bandwidth, but is also limited by the number of other flows which share common components of this selected end-to-end path.

- **Availability.** This parameter measures the time during which the service is operational. Usually, it is expressed as the percentage of the total time. The term "operational" must be defined in a service basis. For example, for a telephone service it means the availability to establish a call in a random time and for Internet services could be the time with packet loss less than 5%.

- **Throughput.** It is a measurement of the efficiency with which the bandwidth is used. Services as VPN-IP have to include some encryption mechanisms to guarantee the service security that can low the throughput for the same bandwidth.

Regarding timing parameters, IP services can be classified as **elastic**, **non real time** and **real time** services, as described below.

Elastic IP services are very tolerant to packet delay and jitter. Data are used by the applications as soon as they are received. Examples of these services are web browsing, file transfer, e-mail, etc. These are the original applications IP was designed for.

Non real time services are IP services with jitter constraints, such as unidirectional audio and video transmission. Timing constrains for these services depend on the capacity of the reception buffer to store a sufficient number of packets so it can continually serve data to the application.

Real time services are interactive applications, such as IP telephony or IP video telephony, and they have strong timing requirements.

Timing requirements of some IP services, along with other parameters such as bandwidth and packet loss, are shown in Table 2.1 and Figure 2.5:

**Table 2.1: IP services requirements**

<table>
<thead>
<tr>
<th>IP Service</th>
<th>Delay</th>
<th>Jitter</th>
<th>Bandwidth</th>
<th>Loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data</td>
<td>Tolerant</td>
<td>Tolerant</td>
<td>Network availability</td>
<td>Not tolerant</td>
</tr>
<tr>
<td>VoIP</td>
<td>Strict</td>
<td>Strict</td>
<td>Low</td>
<td>Tolerant</td>
</tr>
<tr>
<td>Video (rt)</td>
<td>Strict</td>
<td>Strict</td>
<td>High</td>
<td>Tolerant</td>
</tr>
<tr>
<td>Video (n-rt)</td>
<td>Bounded</td>
<td>Strict</td>
<td>High</td>
<td>Tolerant</td>
</tr>
<tr>
<td>VPN</td>
<td>Bounded</td>
<td>Bounded</td>
<td>Medium/High</td>
<td>Not tolerant</td>
</tr>
</tbody>
</table>

But, timing related parameters are not the only aspect to be considered, bandwidth and loss packets are also important parameters. Some services, as VoIP, have small bandwidth requirements, but have strong timing constrains. Although bandwidth and delay can be related, more network bandwidth implies that nodes can forward more packets and, hence, delays are reduced, but, this is not always true as delay and jitter depends on other factors. From the loss of packets point of view, applications span from tolerant applications, as VoIP, to non-tolerant applications, as data banking. Figure 2.5 shows the location of some services in a bandwidth timing constrain axis picture.
Operational parameters reflect additional aspects of services as for example (mean) time to repair, (mean) time to restore service, (mean) provisioning time, etc. Most of these parameters are related to the service provider operational processes, and not to the network itself.

As one of the main objectives of WINMAN is to provide IP connectivity services with QoS, the project will deal with both types of parameters. Regarding service intrinsic parameters, WINMAN will be in charge of providing and maintaining the proper level of QoS and, in the case of operational parameters, WINMAN will help service providers to improve their operational processes and, hence, to improve the quality of the services they provide.

### 2.2.2.1.2 Service Level Agreement

A consequence of market liberalisation there is a strong emphasis that network and service providers are putting on customer care processes. These companies need to differentiate their products from the competition ones as a way to increase the number and fidelity of customers and the revenues.

At the same time, customers require a guarantee that the service provider will fulfil the level of quality it has committed to or, in other case, that the bill will include some discount. Service Level Agreements (SLA) manages these aspects of the relationship between service providers and customers.

A SLA is a contract between two parties (the service provider and the customer), to specify the conditions a service must render. The SLA must reflect all the different aspects of the service that both parties consider of interest, and provide a common understanding about services.

SLAs should contain a number of objectives related to measurable QoS parameters which the service provider guarantees to its customer, such as service performance, customer care, billing (including the discount rates to apply if commitments are not fulfilled), provisioning time, etc. together with other issues as, for example, administrative aspects. The following is a list of items often appearing in a SLA:

- The type and nature of the service to be provided.
• The expected performance level of the service, which includes two major aspects: reliability and responsiveness.
• The process for reporting problems with the service, which form a big part of a typical SLA.
• The time frame for response and problem resolution, which specifies a time limit by which someone, would start investigating a problem that was reported.
• The process for monitoring and reporting the service level, which outlines how performance levels are monitored and reported.
• The credits, charges, or other consequences for service provider if not meeting its obligation (that is, failing to provide the agreed-upon service level).
• Escape clauses and constrains, including the consequences if the customer does not meet his her obligation, which qualifies access to the service level.

As Figure 2.6 shows, the service provider must translate the terms of the SLA to network QoS terms, which are related to the network performance parameters.

![Figure 2.6: Translation of SLAs into QoS](image)

Since SLA parameters must be measured and WINMAN deals with the performance, maintenance and configuration of the IP connectivity services over an optical network, WINMAN will provide tools for service and network providers to effectively manage SLAs.

### 2.2.2.1.2.1 Generic SLA enforcement architecture

The architecture consist of three types of devices, namely Edge Devices, Policy Server Complex and Performance Monitor

**Edge Device**

The edge device is responsible for the functions of identifying the packets and determining the level of performance that is to be provided to them. Thus, the edge device would be responsible for identifying packets that belong to specific customers or specific applications. To support network connectivity SLAs, for example, the edge device identifies the logical tunnel to which the packet belongs and places the packet on the appropriate logical tunnel. The logical tunnel may be provided by different mechanisms in the network - for example, the use of RSVP to establish reserved bandwidth pipes, the use of Differentiated Services (DiffServ) to support specific rates in the network, or the use of a frame relay or ATM tunnel to the appropriate egress edge device.

**Policy Server Complex**
To meet the SLA objectives, the edge device needs to be configured to treat packets in an appropriate fashion. Each of the edge devices can be configured individually. However, separate configuration of different edge devices can lead to possible inconsistencies in their configuration. It is easier and more scalable to validate the consistency of the different edge configurations in a centralised location and to distribute them to the different edge devices in an automated fashion.

The policy server complex is the software component (or group of software components) that contains the configuration information about all the edge devices in the system. In addition to the configuration details, the policy server can also contain a management system that includes the details of the specific SLAs expected of the logical links between the different access routers, as well as the administrative interface for creating and modifying SLAs.

If the configuration of each of the access routers is not co-ordinated, it is likely that packets belonging to the same customer may be marked differently at different access routers and fail to get the desired priority treatment in the network. On the other hand, if the configuration is co-ordinated through a policy server or another centralised tool, the chances of incorrect configuration are less likely.

The policy server is not restricted to keep configuration information on the edge devices only. It can also be used to configure other devices in the network like core routers.

Several protocols can be used to transmit the configuration information from the policy server to the edge devices. One common approach is to store the network policies in a directory server. A directory server is a repository that contains information about the different users in an enterprise. The same repository can be used to contain policies regarding the network configuration.

**Performance Monitor**

The network monitor is the component that monitors the performance of the network between pairs of edge devices and verifies that these are within the bounds specified by the SLA requirements. The network monitor gathers the statistics collected by the different edge devices and examines them to ensure that the desired SLAs are being satisfied.

The network monitor can keep track of the utilisation of the different devices in the network and determine which of the components have failed. These functions are usually provided as part of standard network-management packages. Although the functions to monitor the compliance with SLAs are not commonly found within network management tools, they can easily be provided as additional functions.

If SLA specifications are not being satisfied, the network monitor needs to take some action. Some of the possible actions that can be taken are the following:

- Alert a network administrator that the SLA are being violated. The alert may take the form of an email to the administrator or other alternative means, such as sending a message to the administrator's pager.

- Reconfigure the different edge devices and routers in the middle of the network so that the appropriate network responses are being satisfied.

### 2.2.2.2 Provisioning QoS

When a customer hires a service with a specific QoS, the service provider must manage the network to allocate the necessary resources so the required level of quality can be guaranteed. Several mechanisms are available to provide QoS, most of which can be classified in two basic types [QOS], namely Resource reservation and Prioritisation.

Resource reservation entails that network resources are reserved and allocated according to an application's QoS request, and subject to bandwidth management policy. These resources are exclusively used by the application that requests them and must be negotiated during set up phase. This type fits with the IETF
Integrated Services Model [IntServ] and a signalling protocol is needed in order to allocated specific resources to an application.

Prioritisation means that network resources are shared by all applications, but traffic is classified, giving preferential treatment to traffic generated by applications identified as having more demanding requirements. The IETF Differentiated Services model fits well this type.

Several protocols, like SBM, RSVP and MPLS, have been designed to implement these different types of QoS, some of which will be studied in further sections. These protocols are not exclusive, and can work together to provide end-to-end QoS across multiple service providers.

### 2.2.2.3 Assuring QoS

Once a service has been provisioned with the required QoS work is not yet finished. During the time the service is operational the network has to guarantee that quality is held to the appropriate level. For IP connectivity services, the packets must arrive in such way that the users feel that they are connected to the other end. In order to reach this objective, two main actions must be carried out: QoS monitoring and QoS maintenance.

QoS monitoring is the process to observe if service levels are being met consistently with the SLA. QoS monitoring implies checking that packets arrive to the end application within the committed parameters. When a QoS violation occurs, the actions to be taken depend on the SLA corresponding to the affected service, and span from doing nothing to re-configuring some of the network resources.

QoS maintenance is the process to take the appropriate actions to keep service levels within agreed targets for each service class.

As current network architectures use different technologies at different layers, guaranteeing the level of service that applications need, implies that co-operation between technologies must exist. In this context, co-operation means co-ordination between the different mechanisms that each technology has.

---

**Figure 2.7:** Sample layout of a service supported on IP over DWDM

Figure 2.7 shows a service supported by an IP/DWDM network. This service could be, for instance, the transmission of a video clip from one site to another.
As transmission delays or packet loss increase, the received audio and video signals begin to degrade and, above a specified level, the service is no longer operative. This problem can be produced by congestion on a link between two routers, and so could be solved increasing the bandwidth between the routers or partially re-routing the traffic. Both solutions are carried out at different levels and, probably, for different systems and, hence, co-ordination between the different management system must exist to avoid interference between the actions of different management systems.

As current management solutions are very vertical from network management level to service management level, there is a lack of co-ordination or "horizontal integration". One of the main objectives of WINMAN is to study how this integration must be carried out for providing end-to-end IP connectivity services over an optical (WDM) network.

### 2.2.3. Virtual Private Networks

A Virtual Private Network is a communication environment where access privileges are restricted to permit peer communications only within defined community of interest and is constructed through a means of a common communication medium with the associated network services. Virtual Private Networks can be implemented by using the MPLS enabling technology, as is refer in chapter 2.3.4.5.

A number of technologies must be integrated to provide “true” Virtual Private Network” solutions. They range from simple dial or leased line access for encryption, policy management, directory integration and system management. The successful network service provider will provide a unique set of technologies, integrated to meet the corporation and enterprises to offer VPN services. The technical components from the remote site/user, system access, core network and value-added services are illustrated in Figure 2.8.

![Figure 2.8: Technologies involved to offer VPN service](image)

VPNs products and services provide the opportunity for network service providers to secure and expand their customers’ base in an increasingly competitive telecommunication. Through the VPN, the corporate can provide but not limited to the following services:

- Packet telephony services
- Multimedia applications (e.g. video-telephony, video-conference and interactive learning)
- Database access
- Internet and Web access
- Electronic messaging
- Inventory management
- Customer service
- Collaboration
- Publishing
- Electronic commerce
2.2.3.1 Design issues

The key requirements of a well-designed VPN must include the following attributes:

- **Any-to-Any Connectivity:**
  It must have the capabilities to provide access and communication among two or more sites.

VPNs are built upon the notion of efficiency and security data tunnelling from one point to another. By using the tunnelling concept, the remote access server of the network provider wraps the user data inside packets of format depending on the employed technology by the network provider. Since Internet will be dominated as the technology where VPNs will be built. In this case, the user data are conveyed within an IP packet, which are routed through the carrier’s network or even across multiple networks up to the tunnel endpoint where the packet is unwrapped and sent towards its destination. Tunnelling employs point-to-point session protocols to replace switched connections, linking data addresses over a routed network.

There are two major tunnelling categories namely layer 3 and layer 2 tunnelling. Their major difference regards the location where the tunnels are initiated and terminated. Layer 3 tunnelling terminates the layer 2 connections at the RAS. It carries only the Layer 3 payload through the tunnel to the tunnel endpoint either in the enterprise network or at a router residing in the service provider’s network. On the other hand, layer 2 tunnelling carries the entire PPP frame over the service provider’s backbone to a predefined endpoint. Table 2.2 briefly outlines the two tunnelling types.

<table>
<thead>
<tr>
<th>Tunnelling Type</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Layer 2 tunnelling</strong></td>
<td>Simplicity, End-to-end encryption, Bi-directional tunnel set-up</td>
<td>Standards not finalised yet, Scalability issues, Reliability issues, Limited to PPP payload types, Questions on security</td>
</tr>
<tr>
<td><strong>Layer 3 tunnelling</strong></td>
<td>Scalability, Security, Reliability</td>
<td>Limited vendor participation, Complex to develop</td>
</tr>
</tbody>
</table>

There is a number of tunnelling standards:

- **L2F (Layer 2 Forwarding),** which has been developed by Cisco, Nortel and Shiva.

- **PPTP (Point to Point Tunnelling Protocol):** PPTP operates at layer 2 of the OSI model. It has been designed to work on the client-server basis and that is the reason why it can be employed for a single point-to-point connection.

- **L2TP (Layer 2 Tunnelling Protocol).** It is a hybrid of L2F and PPTP. Like PPTP, it operates at layer 2 of the OSI model thin the tunnel and has been designed for a single point-to-point client-server connection. Multiple protocols can be encapsulated within the tunnel.

- **Security**
  Given a tunnel that defines a method for carrying private information across the shared network, encryption can be used to protect the contents of the tunnel. VPNs support standard forms of cryptography, including public key cryptography and DES (Data Encryption Standard) cryptography.

  Compared to public key cryptography, DES is a symmetric crypto-system. When used for communication, both sender and receiver must know the same secret key, which is used both to encrypt and decrypt the message.

  While DES is CPU intensive, public key cryptography is 1000 times more expensive in terms of CPU cycles. Hence, public key technology is best used as a secure alternative to password-based authentication and for key distribution. DES should be used for bulk data encryption.
If another learns a valid user’s private key, then the user can be spoofed. Due to the requirements to control and distribute keys, a significant management infrastructure is required to control the technology:

- Certification Authorities are employed to vouch for the validity of keys
- X.509 Certificates provide a standardised way of representing names and associated public keys
- Certification Revocation Lists contain certificates and their associated keys
- An X.500 server provides a publicly accessible database for storing certificates and CRLs.

IPsec is a standard-based technology that governs security management in IP networks. Additionally, IPsec provides a standard way to exchange public cryptography keys, specify an encryption method (e.g. data encryption standards (DES) or RC4) and specify which parts of packet headers are encrypted. This is particularly important for emergent application such as the electronic commerce.

- **Scalability:**
  Scalability must include ways to expand the capacity of the existing devices in the network. As an example, in the case of a remote site requiring more connections, another hub can be inserted within the existing architecture. Scalability allows business expansion and eliminates forklift upgrades and usually provides load balancing and redundancy. Scalability greatly depends on the networking technology used to implement VPN services.

- **Network Resiliency:**
  One of the challenges that the service providers face is the network availability. It is therefore important for the VPNs to provide mechanisms for error diagnosis and recovery mechanisms in an efficient way.

- **Reliability and Flexibility:**
  Reliability and Flexibility in a VPN, means availability of services at all times, similar to the telephone network. This means that redundancy features (e.g. RAS) must be added to allow automatic recovery of failed devices. Additionally, VPN should offer flexibility to the corporate sites by taking advantage of the offered services.

- **Usability:**
  The VPN must be very ease to use and understand. For a VPN solution to be successful, the end users of the VPN must use their services without realising it. The VPN must be transparent to the end-user when tunnels are established and torn down.

- **Management Capabilities:**
  This is essential for cost-effective provisioning, management and billing with advanced monitoring and automated flow-through systems to quickly roll out new service and support of Service-Level-Agreement (SLA).

- **Quality of Service:**
  One other major concern is the challenging of QoS handling in VPN. MPLS tackles the problem of “how to make IP and data link technologies (e.g. ATM, FR, WDM) to interoperate by integrating the label swapping forwarding paradigm with network layer routing.

### 2.2.3.2 Specification of SLAs for VPN services

A SLA is used in an IP-VPN network to make assurances about the performance and availability of the network to a customer organisation. Although the network performance (including reliability and availability) is the only factor that the network operator can control, the customer may often be looking for slightly different aspect of performance -namely, the performance (reliability and availability) of the application that is running over the network. SLAs can be specified in terms of application performance or in terms of the performance of the network.
Even when the network performance is the only target, some control of application performance is required. The proper operation on an IP network requires supporting several applications, such as the domain name service. Without adequate performance and reliability assurances regarding these applications, network performance is hard to control.

**Application-level SLAs.**

An application-level SLA specifies performance requirements in terms of a specific application. Examples of such performance specifications may include statements such as the following:

- The database server will have a response time of less than 100 ms with a maximum load of 100 clients.
- The time to download a file of less than 1 MB will be less than 5 seconds.

**Network-level SLAs.**

A network-level SLA specifies its objectives in terms of network performance between one or more exchange points on its network. A network-level SLA may specify network performance using one of the following three approaches:

- **The tunnel approach:** Network-level performance is specified for two specific exchange points on the network.
- **The funnel approach:** Network-level performance is specified from the view of one exchange point only.
- **The Cloud approach:** Network-level performance is specified in a uniform manner across all the exchange points.

**2.2.3.2.1 SLA performance metrics**

The two main performance aspects covered in an SLA are system availability and responsiveness. *Availability* refers to the percentage of time that the application or the network is available and usable by the customer. *Response time* is a measure of the latency that is considered acceptable by the customer.

**Availability metrics**

Several metrics can be defined to measure service availability in an IP network. Some of the most common metrics include:

- **System Uptime** - is the measure of the number of minutes it is available and accessible to the customer. The system uptime can be measured by monitoring the server by using a specific monitoring tool, like PING or analysing the system log to determine whether the system was available and operational. This is more appropriate for an application-level SLA.

- **Network Connectivity** - Measure the percentage of time when one exchange point on the network is reachable from another exchange point, or a POP. This is the analog of uptime for application-level SLAs. A common method to measure network connectivity is to send ping packets from one site in the network to another, to ensure that the packets are not getting lost in the network.

- **Outage Count** - Outages in network connectivity can also be monitored and detected using network management tools. For example, IP routers can be configured to send an alert to a network management tool when an interface or an adapter becomes inoperable.

- **Outage Resolution Time** - A common performance metric used in SLA deals with limits on how soon network problems are resolved.
• **Error Rate** - An error in the network or an application server may be measured in several ways:
  - Measuring the number of failed transactions in the network in a given interval of time.
  - Measuring the number of packets dropped in the network in a given interval of time.
  - Measuring the number of packets corrupted in the network in a given interval of time.

• **Packet Lost Rates Within The Network** - The packet loss rate in the network can be measured as the fraction of packets that fail to reach their destinations.

### Responsiveness metrics

Some of the common metrics used in measure responsiveness of the network includes:

• **Response time** - is the interval perceived by the user of the system to complete a command given to the system. It is also the time the user typically needs to wait before issuing the next command to the system.

• **One-way delay and Roundtrip latency** - If the administrative authority of an organisation covers only part of the network between the client and the server, response times of applications cannot be assured. However, analogs of response time could be measured in terms of one-way delay and roundtrip latency of the network.

• **Delay jitter** - Whereas delays and roundtrip times measure how much time a packet spends in the network, delay jitter measures that the variation in the time spend in the network could be. For many applications, the delay jitter is a more important aspect of network performance than the actual delay.

### 2.2.3.3 Commercial products

Two approaches can be distinguished in commercial IP-VPNs solutions: Network-Based and Customer Premises Equipment based (CPE-based).

CPE-Based solutions are deployed as an overlay layer over the SP network. The corporation installs CPE equipment that connects to the public infrastructure, and the VPN solution relies on CPE functionality to provide tunnelling, security, routing and management of the VPN. In this type of realisation of a VPN, the SP network “cloud” is completely unaware of the VPN(s) that are being transported over the basic IP infrastructure. This type of VPN demand high technical skills on the corporate side, and, as an additional layer on a public infrastructure, is not possible for the corporate customer to demand QoS from the network, but has to find a way to provide traffic control based on CPE. Two products of this category are presented hereafter.

For scalability and economic reasons, providers who want to deploy wide scale IP-VPN services must “IP-VPN enable” their networks; that is, the forwarding mechanisms of the network infrastructure, whether router-based, switch-based, or a hybrid of both, must be integrally aware of IP-VPN partitioning, without having to use overlay models to establish connectivity.

Instead, traffic forwarding must actively participate in partitioning and inter-VPN membership control. Routed and switched backbones can be “IP-VPN enabled” through the use of label switching technology known as the Multiprotocol Label Switching (MPLS) standard. Network-Based solutions focuses on the provision of a complete transparent VPN solution to the Service Providers’ corporate customers, that is to say, customers’ CPEs just plug-in into the provided VPN access points regardless the underlying technology utilised by the SP. In this type of realisation of a VPN, the SP network “cloud” is completely aware of the VPN(s) that are being transported over the basic IP infrastructure. The customer can deploy the desired service based on standard IP routers, and the SP guarantees (based in a SLA) the service QoS. The SP carries out the management of the VPN.

**CheckPoint** [CHECKPOINT-01]
This product is CPE-Based. CheckPoint traditionally used to provide firewall-based security solutions for enterprise customers. With the emergence of Internet as a business platform, the companies soon realised that besides secure sites they needed to ensure traffic privacy and enterprise-wide management of the Intranet or Extranet.

This vendor fits into this market adding some traffic control features, based on proprietary QoS enforcement equipment. Being a CPE-based solution, the claim of policy-based traffic management lacks the complement of a “QoS enabled backbone” from the Service Provider, or the wide-open Internet, which constitutes the “weakest link” on the QoS “chain”.

The solution provided is composed by a set of software tools for management of the VPN, including tunnelling, encryption, access control, key (certificates) management, traffic enforcement, IP management and secure remote access.

In order to effectively utilise the Internet for wide-area communications between enterprise branch offices, mobile workers, and business partners, organisations typically need to implement a combination of intranet, remote access, and Extranet VPNs, each with a unique set of security, traffic control, and management requirements. Check Point’s VPN-1 product family addresses all of these requirements in a single, integrated solution, allowing organisations to integrate a VPN into their overall security framework. Once the security policy has been defined, it can be distributed to all network access points and managed from a single, centralised management console.

**Lucent/Xedia [XEDIA-01]**

This solution is very similar to CheckPoint’s. It provides IP-VPNs based on tunnelling and encryption between proprietary CPE-equipment. The result is a full mesh of secure tunnels over the public Internet. The VPN boxes also provide traffic enforcement features managed by a Policy-Based Management platform. The solution can also manage Dial-up tunnels to integrate them into the IP-VPN infrastructure.

This solution, as well as CheckPoint, is an overlay of the public Internet, and as it was previously mentioned, the promise of policy-based traffic management can only be fulfilled if the VPN equipment can co-ordinate its QoS features with the Service Provider’s backbone.

If this condition is met (i.e. Service Provider owns VPN equipment compatible with it backbone infrastructure), these solutions could be considered “Network-Based”, but will still have scalability problems.

**Cisco [CISCO-01]**

The Cisco solution is a network-based VPN making use of MPLS enabled networks. It comprises network equipment running MPLS as labelling/switching protocol, RSVP as signalling protocol for Label Switched Path establishment and control, OSPF or IS-IS as Interior Gateway Protocols with Constraint-Based routing extensions, and BGP for VPNs routing information distribution. These features are distributed over the network equipment, and not every router runs every piece of software. Needless to say, provisioning such an infrastructure on a network scale based on traditional CLI commands in a timely manner is not feasible. That’s why this vendors’ solution is complemented with a set of software tools to deal with VPN provisioning and management.

Compared to an overlay solution, an MPLS-enabled network can separate traffic and provide privacy without tunnelling or encryption. MPLS-enabled networks provide privacy on a network-by-network basis, much as Frame Relay or ATM provides it on a connection-by-connection basis. Where traditional VPN service offers basic transport, an MPLS-enabled network supports scalable VPN services that can deliver value-added, IP-based applications on top of the basic VPN transport network.

Logical ports on the provider edge router are associated with particular VPNs when provisioned.

During VPN provisioning, a unique identifier is assigned to each VPN, and the end user is unaware of this identifier. Thus, a packet can enter a VPN only via a logical port configured for that VPN. The provider, not the customer, associates a specific VPN with each interface when provisioning a VPN. Users can only participate in an Intranet or Extranet if they reside on the correct physical or logical port and have the proper identifier. This set-up makes an MPLS-enabled VPN virtually impossible to enter and just as secure as PVC-based Frame
Relay or ATM VPNs. For most VPN traffic this level of security is adequate. However, all packets are transmitted in clear text, so one cannot absolutely guarantee privacy. Some industries are required to encrypt all traffic passing into a public network, and some business customers may choose to protect sensitive, mission-critical traffic using encryption. Encryption can take place at the CPE before traffic enters the MPLS-based VPN for transport to another VPN site, and traffic is decrypted at the destination CPE. [CISCO-02]

Cisco completes its “VPN portfolio” with Access Servers “tunnel-enabled” (L2TP compliant) with RADIUS or TACACS+ AAA in order to provide Dial-up access to the IP-VPN, and tunnelling terminating equipment to support this dial-up tunnels.

**Orchestream** [ORCHESTREAM-01]

This vendor provides a software solution that can be categorised as Network-Based, because it relies on MPLS capabilities of the underlying network infrastructure to deploy VPNs, similar to Cisco. Furthermore, Differentiated Services can be implemented on the VPNs, managed by a Policy-Based Network Management platform.

The software solution has the advantage of being multi-vendor, on the network element side. In fact, it can manage Cisco, Lucent/Xedia and Bay routers via proxy agents. On the other hand, this product depends on equipment vendors’ compliance with the relevant standards used in the solution, or in proxy agents that generally have only partial functionality.

### 2.2.4. Application-oriented services: VoIP and MoIP

We are experiencing a technology revolution both in the telecommunication and Information Technology. Existing voice networks using circuit switching have dominated the telecommunication area. However, it is expected that the volume of data traffic will exceed that of voice in the near future due to the Internet explosion. This explosion will have a great impact in the operation of both telecommunication operators and Internet Service Providers. IP-based data networks will replace large parts of the today’s voice dominated networks where a number of existing (e.g. voice, Internet access) and future services (e.g. multimedia) will be provided to the end-user. However, this transition towards a data type of networks will be smoothly rather revolutionary [HAMDI], [PRYCKER].

During the past years both the traditional circuit-switched networks and the emerging IP data networks, deployed throughout the world, have been considered as two separate worlds [HAMDI], [INTCOMP]. These worlds have started to converge by examining the possibility of offering voice services over the Internet (VoIP). VoIP offers not only advanced voice services but it provides means of connectivity to the existing voice networks.

The major factor driving towards this convergence is the requirement and need towards service-driven networks. The benefits and advantage of IP is that it is a common transport for all types of networks. In the deregulated world of the near future, provision of advanced services is the major goal offered by the carriers to their subscribers.

#### 2.2.4.1 SLA specification for voice and multimedia services

VoIP is an alternative to the traditional telephony service, and as such, users expect from it a quality similar to that they have been provided for years. Although the quality of this service, as perceived by its users, is a very subjective question, it is well known that there are a number of measurable technical parameters with a strong influence on the final quality of the conversation. ITU-T has made important efforts to define desirable performance levels on the telephony network, and has published several recommendations addressing these issues.

A SLA between a customer and a VoIP provider would ideally take into account these recommendations on parameters such as end to end delay, or echo attenuation, and should also include quality goals usually applied to the telephony service, like establishment time or percentage of failed calls.
A bi-directional MoIP application has similar requirements to those of VoIP, in terms of jitter and end-to-end delay. If the communications is only unidirectional, requirements are not so strict, but still jitter and delay should be bounded. In both cases (bi-directional and unidirectional) audio and video are more bandwidth demanding and require less packet loss than telephony 3.4 KHz voice, in order to achieve a better quality. Accordingly, the contents of a MoIP SLA will not differ much from a VoIP SLA, the differences being in the magnitude of the parameters.

The SLA will contain both provider and customer commitments, how all considered parameters are going to be measured, and the actions to be taken if any of the commitments are not met (i.e. penalties, discounts, etc). The contents of the SLA may differ depending on the type of terminals used, whether traditional telephones (which need a gateway), IP telephones or PCs (which don't need the gateway).

The most common parameters of a VoIP/MoIP SLA would then be:

- **End to end delay**: ITU-T defines a maximum acceptable end-to-end unidirectional delay in its G.114 Recommendation [ITU-2] of 150 ms, but today networks typically work with smaller values. The SLA could specify, for instance, a maximum delay of 100 ms for 99.99% of calls, and an average delay of 60 ms. For unidirectional communications, this value could be greater, provided the jitter keeps under acceptable values.

- **Jitter**: when the VoIP service includes a voice gateway, the network jitter only amounts for the total delay, or the packet loss ratio, but it is not a parameter end users would care about. But if one of the terminals is not outside a gateway (a PC, an IP telephone), or the gateway is owned by the customer, jitter becomes a crucial parameter, that needs to be taken into account when receive buffers are set up. Therefore maximum and average values should be defined in the SLA. The same distinction applies for MoIP services, which may be provided through a gateway, or without it.

- **Packet loss**: voice/audio/video quality degrades significantly when the number of lost packets exceeds a certain percentage, percentage that depends on the voice compression algorithm (or no compression at all) in use. This degradation is aggravated by the fact that packet loss, usually due to network congestion, shows a bursty nature. So, meanwhile the average loss ratio keeps under acceptable values, the communication may get temporarily cut. The SLA should in turn define desirable packet loss values for small time intervals (i.e. 5 % during 3 seconds).

- **Bandwidth**: when any of the terminals are directly connected to the VoIP/MoIP network, the minimum bandwidth that the network will support should be agreed between the provider and the customer and is included in the SLA. This bandwidth, together with the compression applied to the voice, audio or video, determines the number of simultaneous sessions that can be established.

- **Failed calls**: this parameter is commonly used in traditional telephony networks as a means to measure its correct dimensioning. For a provider to guarantee a certain maximum percentage of failed calls (for instance, 2% of all calls), the customer must promise not to exceed a certain offered traffic load, and keep under what promised (for instance, no more than 200 calls per hour). Due to the heterogeneous nature of multimedia communications, this doesn't seem an appropriate parameter for measuring its quality.

- **Simultaneous calls**: the SLA may specify a minimum number of simultaneous telephone calls (or multimedia sessions) the VoIP (or MoIP) provider can always handle.

- **Voice, audio and video quality**: the quality of voice, audio or video, when delay, jitter and packet loss goals are met, depends mainly on the audio/video codec selected. The SLA should in turn list the algorithms supported, if the user can choose among them, or the one that will be used, if not. Some of the options for audio are those specified by the ITU-T: G.711 (audio coding at 64 kbps) [ITU-3], G.722 (64, 58, and 48 kbps) [ITU-4], G.723.1 (5.3 and 5.3 kbps) [ITU-5] and G.729/G.729A (8kbps) [ITU-6]. For video, H.261 [ITU-7] and H.263 [ITU-8], along with other competing solutions like MPEG or QuickTime, can be used. The user may prefer to request a subjective quality (high, medium
or low) and leave the provider the responsibility to map the request to the appropriate audio and video encoders.

- **Establishment time**: also known as post dialling delay (PDD). It is the time between the end of user or terminal equipment dialling and the reception of the appropriate network response. It should be kept to low values (for example, less than 3 seconds for 95% of calls). Two related parameters are the start dial signal delay (SDSD), time interval between off-hook and reception of start dial signal, and the call clearing delay (CCD), time interval between the clearance signal from the end users, and the networks return to ready to serve state. The ITU-T addresses these issues in its E.431 Recommendation [ITU-9].

- **Echo level**: echo is a phenomenon found in the telephone network that disturbs conversations. Imperfect decoupling when a signal traverses from a 2-wire to a 4-wire physical link causes it. Echo becomes a significant problem when the round-trip delay is greater than 50 milliseconds. To reduce echo, VoIP gateways must implement some means of echo cancellation. ITU-T recommendation G.126 [ITU-10] and G.131 [ITU-11] contain several considerations about listener and talker echo, showing acceptable echo attenuation level versus the total end-to-end delay. A possible value for this parameter could be an attenuation of 25 dB for 95% of calls. Echo is not a relevant issue in multimedia communications, unless a traditional telephone conversation is part of it. In this case, the same echo compensation mechanisms should be implemented.

- **Features support**: if services such as call waiting or call forwarding are supported by the VoIP system, its conditions of use could be part of the SLA as well. The MoIP SLA can also specify additional features the service support.

- **Offered traffic**: is a compromise reached with the customer, defining how much voice/multimedia traffic it will offer the VoIP/MoIP network. If the actual traffic offered exceeds that compromised, some (or all) of the provider commitments may not apply. For traditional telephone calls, the traffic offered is usually measured in Erlangs. For data terminals, the offered traffic will equal the requested bandwidth, and thus will be measured in megabytes per some unit (for example, a limit of 1.000 MB a day could be established).

- **Operational parameters**: reliability, time to repair, security, time to provision, etc. There are not special considerations to be made for VoIP/MoIP services.

### 2.2.4.2 SLA example for VoIP services

#### General conditions

This Service Level Agreement (SLA) specifies the commitments made by the service provider and by the customer for a telephony service transported over an IP network. The access to the IP network will be based on a gateway connected to a PBX located in the same premises, within a maximum distance of 200 metres. All the conditions expressed in the present document refer uniquely to the part of the network between two such gateways, both of them included. No commitments are made for what happens outside these boundaries. The SLA violations will be determined on a monthly basis.

**Service dimensioning:**

- The service will support a maximum of 60 simultaneous calls.
- The percentage of failed national calls will be less than 1%.
- The percentage of failed international calls will be less than 2.5 %.
- The average traffic offered by the customer during one month, and computed only during working hours (from 08:00 to 18:00) should be less than 20 Erlangs. If not, the traffic in excess of that limit won't be considered when determining the conformance of the service to the previous commitments.

**End-to-end delay**

- The total one-way end-to-end delay in both directions will be less than 150 ms for 99.999% of calls.
- The average one-way end-to-end delay will be less than 90 ms.
Voice quality
- The service will use G.729A voice codification, supporting 3.4 KHz speech at 8 Kbps.
- The average packet loss will be less than 1%
- The packet loss will never exceed 10% during 10 seconds intervals.

### 2.2.4.3 Network architecture

Figure 2.9 illustrates the functional devices of an IP network towards the provisioning of VoIP services. It is obvious that in this architecture, a distributed model has been adopted where different devices are responsible for the management and the operation of VoIP service [NARAY].

![Functional building blocks of an IP telephony service](image)

Such a network comprises the following devices:

**IP Telephony G/W.** The gateways perform functions for the media conversion of the voice samples and signalling conversion between SCNs (e.g. PSTN, ISDN, GSM) and the Internet. There are different types of gateways, which can meet diverse requirements related to cost, performance and deployment of services:

- Trunk gateways which interface between the circuit switched network and the Internet. These types of gateways can handle a large number of digital circuits simultaneously and are typically dedicated network elements.
- PBXs centers with IP interfaces and Key systems. These gateways coexist with PBX centres and Key systems and may have diverse capabilities such as call routing, trunk selection and QoS service. They have the advantage of offering the reliability and scalability of the traditional telephony systems.
- Routers and Remote Access Servers with capabilities of supporting voice services. Such gateways have the capabilities of routers and RAS such as routing, bandwidth control.
- Dedicated gateways. These are equipments of media and signalling conversion whose primary functionalities is the provisioning of gateway services.
- Gateways for analog centres. Their primary function is to allow to analog terminals such as phones, fax and modem to communicate through IP networks.

**IP terminals.** This equipment regards the end-users and may contain the following categories:

- Telephone handsets with communication capabilities using the IP protocol.
- Software-based telephone (Softphones). This is software for PC allowing voice and multimedia communication
• Telephone with USB (Universal Serial Bus) interfaces. These telephone devices can be connected directly to a PC. They allow to the remote users and telecommuters to use one desktop device supported by a single wiring plant and still have access to the familiar telephone user interface.

**IP Telephony Applications.** Such applications provide value-added services within the context of converged networking and include:

- Unified messaging applications, which integrate voice mail, email, SMS and fax.
- Call-centre applications which router and process calls to the appropriate person or agent according to specific sets.
- Interactive Voice Response (IVR) and applications, which are activated from the speech. They use the human voice and voice tones to recognise user inputs and provide text –to voice conversion.
- Mobility-related application allowing travellers and mobile workers to set-up and receive call.

**Service Control Units (SCUs).** These devices contain dedicated processors, which combines both the capabilities of a PBX processing a call and a gatekeeper controlling the resources of IP Telephony such as gateways, terminals and network capacity. A SCU device provides an API interface for the integration of advanced telephony applications in an IP telephony environment. There are two different categories for SCU devices:

- SCUs, which are independent from the hardware platforms and use the client-server model. There are software modules implemented in platforms such as Windows NT, LINUX etc. and enable to the IP networks to offer telephony as a service.
- PBXs with IP extensions και Key systems. It includes call processing combined with IP Telephony call control in order to integrate IP terminals and trunk gateways.

### 2.2.4.4 Products and standards

This section provides a description of the standardisation activities of different international organisation and fora in IP telephony:

- **ITU-T H.323**
  H.323 is a suite of standards, which have been adopted by ITU for real-time multimedia communication, in LANs and the Internet [H323]. The standard covers both single point and multipoint communications. More specifically, H.323 defines the following protocol suite for multimedia communications [H323]:
  
  - H.245 for control
  - H.225.0 for connection establishment
  - H.332 for large conferences
  - H.450.1, H.450.2 and H.450.3 for supplementary services

  Additionally, H.323 can be used with Internet Protocols such as UDP, RTP, RTCP and RSVP. Furthermore, Q.931 is used for call signalling. This can be accomplished by implementing Q.931 over TCP. Figure 2.10 illustrates the H.323 protocol stack over Internet.
H.323 standard defines four components, which are essential to be in a network. Such components are the following:

- **Terminals**: Terminals are used for the bi-directional multimedia communications. An H.323 terminal can be either a PC or an autonomous terminal «running» the H.323 protocol as well as the associated multimedia applications. The basic functionality of an H.323 terminal is the communication with other H.323 terminals as well as with terminals of the telephone network (e.g. H.320 terminals in ISDN, H.310 terminals in B-ISDN).

- **Gateways**: The basic functionality of the gateway is to provide connectivity capabilities between an H.323 compatible and a non H.323 compatible network. This connectivity is accomplished through the signalling for the call set-up and release, the media transformation of the encoded voice and the transportation of the voice data between the two networks.

- **Gatekeepers**: The gatekeeper is considered as the «brain» of the H.323. It is the focal point for all the calls towards an IP network supporting the H.323 protocol. The services offered by the gatekeeper include addressing, registration, authorisation and subscription, management of the bandwidth, billing and accounting. The gatekeepers can also handle call routing services.

- **Multi-point Control Units (MCUs)**: The MCUs provide mechanisms for the support of three or more H.323 terminals in a multipoint communication. All the terminals participating in a multipoint communications, are connected to the MCU. The MCUs manage the resources of the multipoint communication and negotiate with the terminals for the type of the audio/video codec, which will be used. Figure 2.11 illustrates the H.323 components in a VoIP infrastructure.

**ITU-T H.GCP/H.248**
The ITU-T SG16 has been working in parallel with ETSI TIPHON and IETF MEGACO WG towards the specification of the protocols for a modular architecture. ITU-T has also released a protocol for the control of the gateways in a VoIP network called MDCP [MDCP]. In April 1999, ITU-T SG16 adopted the MEdia
GAteway Control (MEGACO) protocol for the communication between the Call agent/Media Gateway Controller and the media gateways. This protocol was initially called H.GCP and later as H.248 (H-series Gateway Control Protocol) [H248].

IETF has defined the Working Groups Audio Visual Telephony (AVT), PSTN and Internet Internetworking (PINT), IP Telephony (IPtel), Multimedia Multiparty Session Control (MMUSIC) and Media Gateway Control (MEGACO) which are currently working on the protocols associated with Voice Over Internet [IETF]. There are two different groups of protocols, which are associated with the VoIP. The first group regards protocols, which are related to the transmission of IP packets containing voice samples. The second group regards the protocols, which are associated to the VoIP applications and services.

- **IETF Transmission-Related Protocols**
The protocols concerning with the transmission of IP packets for applications such as VoIP, are the following:

  - The RTP (Real Time Transport) [SCHUL] protocol is used for the packetisation of the bits of the real-time application (e.g. audio, video). This protocol is usually implemented over UDP.

  - The protocol RTP is associated with the control protocol RTCP (Real Time Control Protocol) that monitors the QoS of the connection and synchronises the different types of sessions (e.g. voice, video and data) through the use of timestamps. Both senders and receivers send in predetermined time periods RTCP packets. Each RTCP packet contains elements such as sender-ID, receiver-ID and sender session description. The periodicity of the RTCP packets depends on the number of participants in the session.

  - The RSVP protocol (ReSource reservVation Protocol) [BRAD] is a signalling protocol, which is used by a host to request specific QoS from the network for a particular flow such as video and audio. RSVP is also employed by routers to deliver QoS requests to all nodes along the path(s) of the flows, establish and maintain states to provide the requested service.

- **IETF VoIP-related protocols**
The IETF WGs IPtel, AVT, PINT, MMUSIC and MEGACO have adopted a set of protocols which can be used for the provisioning of telephony services in the Internet. These protocols are briefly presented below:

  - **Session Initiation Protocol (SIP)**: SIP [HANDL], [ROSEN] is a lightweight protocol, which is proposed by IETF for the call control of VoIP services. SIP offers a lot of advantages a mean of platform to offer voice services in an IP network. It is a client-server type of protocol and its model is illustrated in Figure 2.12.

    ![Figure 2.12: A model of Session Initiation Protocol (SIP) services](image)

    SIP functionalities include name translation, user location, and feature negotiation. The servers handling SIP requests can be either proxy servers (SIP requests are forwarded towards the next server) or redirect servers (the server sends a message to the user informing him for the location of the next server for the processing of the call).

  - **Session Description Protocol (SDP)**: The SDP protocol [JACOB] is used in order to describe multimedia type of sessions for voice applications and distribution such as Internet audio.
• **Real Time Streaming Protocol (RTSP):** The RTSP (Real Time Streaming Protocol) [HSCHU] is mainly used for the control of a server which different type of stored media (e.g. voice, video). Such a server has the capabilities to provide encoded information towards the networks and in the same time can store compressed information of different media. Such server can be used for storage of the content of a conference for future use and storage of voicemail messages.

• **Service Location Protocol (SLP):** In a VoIP network, there are a number of gateways, which provide connectivity to SCN. Each domain contains a number of users and Location Servers (LS). SLP [VEIZA] is an intra-domain protocol and is used in order to inform the IP users about the location of the gateways within the domain.

• **Gateway Location Protocol (GLP):** In cases where a user in an IP network wants to communicate with the subscriber of a PSTN network, he must communicate with a Gateway for signalling and media conversion in the appropriate format. The GLP (Gateway Location Protocol) [ROSC] is used for the selection of the appropriate gateway such that the distance between the PSNT and IP network is minimised. Additionally, GLP can be used by LSs to build up a database of gateways in other domains.

• **Media Gateway Protocol (MGCP):** MGCP is a new protocol, which can be used by the Call Agents for the control and management of the gateways [ARAN]. MGCP is a merge of SGCP (Simple Gateway Control Protocol) and IPDC (Internet Protocol Device Control) protocol. The philosophy of MGCP is that the “intelligence” of the “call control” resides in units outside the media gateways. Therefore, MGCP has introduced the concept of connections, end-points, events and signalling for end-to-end telephone connections. If two residential gateways participating in a two-party call are controlled by different Call Agents, then co-ordination between the two Call Agents is required for the call setup. MGCP includes the following eight commands: CreateConnection, NotificationReuest, Notify, ModifyConnection, DeleteConnection, AuditEndpoint, AuditConnection, RestartInProgress.

• **ETSI TIPHON**

The European organisation ETSI (European Telecommunications Standardisation Institute) through its project TIPHON (Telecommunications and Internet Protocol Harmonization Over Networks) is currently working towards the standardisation of VoIP architecture [25]. This project is funded by major vendors from both USA and Europe (e.g. Cisco, Lucent, Ericsson, etc) and from major telecommunication organisations (e.g. AT&T, DT, KPN, etc).

The ETSI TIPHON project aims to define mechanisms for the communication of users between IP and the Switched Circuit Networks (e.g. PSNT, ISDN, GSM, SS7). Such a reference configuration is illustrated in Figure 2.13.

![Figure 2.13: TIPHON Reference Configuration](image-url)
ETSI Tiphon has recently launched an initiative called TIPIA (Tiphon IP-Telephony Implementation Association) [TIPIA]. Its aim is to support interoperability of Tiphon-compliant products.

- **Standards from Other Fora**

  There are also other fora, which are currently working on VoIP related issues. Their work is briefly described below:

  - **IN Forum [INFOR]** examines inter-working issues between SS7 and IP for the provisioning of intelligent services in VoIP users. The Forum examines the development of Advanced Intelligent Network Services (AIN), which will lead in the convergence between the Internet and IN technology.
  
  - **The aim of the IMTC (International Multimedia Teleconferencing Consortium) [IMTC]** is to study the interoperability issues for H.320 and H.324 standards, which regards videoconferencing systems for ISDN networks and networks with low bit-rate connections such POTS connections.
  
  - **The VTOA group of ATM Forum [VTOA]** examines the transmission of telephone services over ATM networks. The VTOA group considers the use of AAL1 (uncompressed PCM 64Kbps) and AAL2 (compressed voice) protocols as well as the advantages offered by the use of ATM technology.
  
  - **The Telecommunications Industry Association (TIA) organisation [TIAI]** has set-up an effort towards the creation of standards for IP telephone terminals. This effort intends to examine through a common platform standards, which have been promoted by ITU and by IETF.
  
  - **The Interoperability Now (iNow) organisation [INOW]** provides solutions to the VoIP manufacturers so that there is interoperability among the different products. The organisation examines issues such as interoperability among gateways, gatekeepers and services and has defined the specifications regarding this interoperability.
  
  - **The Softswitch Consortium [SOFT]** is an international organisation for interactive and real-time communications in IP networks.

### 2.3. Enabling Technologies.

#### 2.3.1. Introduction

This chapter is devoted to review the technologies existing in the networks (IP and WDM) that can be used to provide the required services.

The first section addresses the topic of end-to-end QoS with the intention to point out that in order to reach such an objective is necessary to cross several network sections and hence to apply different mechanisms or technologies.

The section 2.3.3 presents the well-known approaches in the IP network, namely the Differentiated Service Approach (DiffServ).

Section four is entirely devoted to MPLS. After a description of the concept and an architectural overview we concentrate on the alternatives to take profit of such forwarding mechanisms to get end-to-end QoS. VPNs over MPLS are also considered here.

The last section of the chapter is devoted to the optical layer. More specifically we deal with the background concepts of WDM going in more detail with techniques allowing the control of service characteristics and the mapping between the IP layer and the optical layer. In this context, we present the Optical Differentiated Services mechanism and the MPLambdaS the version of the MPLS in the optical layer.

#### 2.3.2. End to end QoS in IP networks

IP connectivity may require packets to traverse many different network segments since the source terminal issues them until arriving to the destination terminal. In order to get end-to-end connectivity with QoS, all these segments must contribute to the overall performance. This contribution entails supporting specific protocols
and co-operation with other sections and layers. The purpose of this section is to outline the scope and functionality of these protocols, leaving for next sections a more detailed description of the most representative ones. Mapping of QoS between different administrative domains are not considered hereafter.

Starting at the host where the application resides, we can assume that a connection is requested to send packets to a remote host. The applications should be capable of signalling their requirements, which in turn should be mapped onto the appropriate QoS mechanisms of each relevant layer. A feasible approach consists of RSVP-aware applications that make reservations over the end-to-end network path to ensure QoS. However, if sender and receiver are connected at shared media LAN segments, the reserved bandwidth could suffer from degradation at the LAN level. That’s why a map between RSVP signalling at the IP layer and the underlying Data-Link layer is needed. Therefore end-to-end QoS comprises sender to receiver signalling and enforcement, but also top to bottom mapping of QoS mechanisms.

Once the packet leaves the host it usually goes through a local area network. The access LAN must enable QoS so high-priority frames receive high-priority treatment as they traverse the network media. LANs support OSI Layer 2. Therefore it is necessary to allow appropriate mapping between QoS support at layer 2 and all the above layers.

Some Layer 2 technologies like ATM are inherently QoS-enabled. However, other more common LAN technologies such as Ethernet were not originally designed to be QoS-capable. As a shared broadcast medium or even in its switched form, Ethernet provides a service analogous to standard “best effort” IP Service, in which variable delays can affect real-time applications. However, the IEEE has “retro-fitted” Ethernet and other Layer 2 technologies to allow for QoS support by providing protocol mechanisms for traffic differentiation.

The IEEE 802.1p, 802.1Q and 802.1D standards define how Ethernet switches can classify frames in order to expedite delivery of time-critical traffic (802.1D defines the Spanning-Tree Protocol, 802.1Q is a specification for VLAN tagging and 802.1p is a specification that permits to assign priority levels to frames at layer 2). The IETF Integrated Services over Specific Link Layers (ISSLL) Working Group is chartered to define the mapping between upper-layer QoS protocols and services with those of Layer 2 technologies, like Ethernet. Among other things, this has resulted in the development of the “Subnet Bandwidth Manager” (SBM) for shared or switched 802 LANs such as Ethernet (also FDDI, Token Ring, etc.). [QoS-01]. SBM is a signalling protocol that allows communication and co-ordination between network nodes and switches and provides a method for mapping an internet-level set-up protocol such as RSVP onto IEEE 802 style networks. [SBM-01]

Leaving the LAN and until they reach the far end, packets are sent through a transport network. Well known approach to allow QoS in this part of the trail consists in making use a hybrid Intserv-DiffServ approach [IntServ-DiffServ].

In this hybrid approach RSVP signals resources requested by applications, whereas DiffServ marks and prioritises traffic. RSVP is more complex and demanding than DiffServ techniques in terms of router requirements, so can negatively impact backbone routers. That’s why is commonly accepted that the use of RSVP should be limited on the backbone, and DiffServ should be the choice there.

In this hybrid approach end hosts may use RSVP requests with high granularity (e.g. bandwidth, jitter threshold, etc.); border routers at backbone ingress points can then map those RSVP “reservations” to a class of service indicated by a DS-byte (or source host may set the DS-byte accordingly also). At the backbone egress point, the RSVP provisioning may be honoured again, to the final destination. Ingress points essentially do traffic conditioning on a customer basis to assure that service level agreements (SLAs) are satisfied. DiffServ and IntServ can be used independently from end-to-end between sender and receiver. However, some sort of co-operation is needed between this mechanisms, applications and Layer 2 QoS protocols to provide top-to-bottom and end-to-end QoS between senders and receivers

With Differentiated Services (DiffServ), packets are classified at the edge of the network. The Differentiated Services-field (DS-field) of the packets is set accordingly. In the middle of the network, packets are buffered and scheduled in accordance to their DS-fields by different queuing mechanisms, namely Weighted Random Early Detection (WRED), Weighted Round Robin (WRR), Weighted Fair Queuing (WFQ), Class-Based Weighted Fair Queuing (CBWFQ). Important traffic such as network control traffic and traffic from premium customers will be forwarded preferably.
MPLS has been claimed to contribute also in the end-to-end provision of QoS. For instance, it can give support to DiffServ. MPLS makes possible to apply scalable QoS across very large networks because providers can designate sets of labels that correspond to service classes, or Layer 3 CoS. For example, a service provider network may implement three service classes: a high-priority, low-latency "premium" class, a guaranteed-delivery "mission-critical" class, and a low-priority "best-effort" class. The effect is that the physical network is divided into multiple virtual networks, one per class. These virtual networks may have different topology and resources. The end effect is that premium traffic can use more resources than best effort traffic. MPLS is also the base for traffic engineering. [TE-01]

2.3.3. DiffServ

2.3.3.1 DiffServ Architecture

Differentiated Services (DiffServ, or DS) is a protocol for specifying and controlling network traffic by class so that certain types of traffic get precedence over others. For example, voice traffic requires continuous uninterrupted flow of data, thus it could precede other classes of traffic. In contrast to the per-flow orientation of IntServ, DiffServ networks classify packets into aggregated flows or "classes", by affecting the bits in the TOS field of the packet’s IP header. Furthermore, QoS in DiffServ can be deployed using long-term provisioning rather than short-term reservations established by end-to-end signalling.

Differentiated Services is considered to be the most advanced method for managing traffic in terms of what is called Class of Service. Unlike the earlier mechanisms of 802.1p tagging and Type of Service (ToS), Differentiated Services avoids simple priority tagging and depends on more complex policy or rule statements to determine how to forward a given network packet. An analogy can be made to travel services, in which a person can choose among different modes of travel - train, bus, airplane - degree of comfort, the number of stops on the route, standby status, the time of day or period of year for the trip, and so forth. For a given set of packet travel rules, a packet is given one of 64 possible forwarding behaviours - known as per hop behaviours (PHBs). A six-bit field, known as the Differentiated Services Code Point (DSCP), in the Internet Protocol (Internet Protocol) header specifies the per hop behaviour for a given flow of packets.

Differentiated Services and the Class of Service approach provide a way to control traffic that is both more flexible and more scalable than the Quality of Service approach. The need for scalability arises from the fact that hundreds of thousands simultaneous traffic flows may be present at a core router. We will see shortly that this need is met by placing only simple functionality within the network core, with more complex control operations being implemented towards the "edge" of the network -- very much in line with the architecture of the current Internet. The need for flexibility arises from the fact that new service classes may arise and old service classes may become obsolete. The differentiated services architecture is flexible in the sense that it does not define specific services or service classes, as is the case with IntServ. Instead, the differentiated services architecture provides the functional components, i.e., the "pieces" of a network architecture, with which such services can be built.

DiffServ uses packet marking and per-class queuing to support priority services over IP-based networks. In the differentiated services architecture, a packet's mark is carried within the so-called Differentiated Services (DS) field in the IPv4 or IPv6 packet header. The definition of the DS field is intended to supersede the earlier definitions of the IPv4 Type-of-Service field and the IPv6 Traffic Class Field. The structure of this 8-bit field is shown below in Figure 2.14.

![Figure 2.14: Structure of the Differentiated Services (DS) field in IPv4 and IPv6 headers](image)
The 6-bit Differentiated Service Code Point (DSCP) sub-field determines the so-called per-hop behaviour (PHB) that the packet will receive within the network. Informally, we may think of this as the "class of traffic" to which the packet belongs. The 2-bit CU sub-field of the DS field is currently unused. Restrictions are placed on the use of half of the DSCP values in order to preserve backward compatibility with the IPv4 ToS field use; see for details. We need only note that a packet's mark, its "code point" in the DS terminology, is carried in the 8-bit DS field.

### 2.3.3.2 DiffServ Operation

![DiffServ edge and core router functions](image)

**Figure 2.15:** DiffServ edge and core router functions

#### 2.3.3.2.1 Edge functions

At the incoming "edge" of the network (i.e., at either a differentiated services (DS)-capable host that generates traffic and/or at the first DS-capable router that the traffic passes through), arriving packets are marked. More specifically, the Differentiated Service (DS) field of the packet header is set to some value. The mark that a packet receives identifies the class of traffic to which it belongs. Different classes of traffic will then receive different service within the core network. After being marked, a packet may then be immediately forwarded into the network, delayed for some time before being forwarded, or may be discarded. Many factors can influence how a packet is to be marked, and whether it is to be forwarded immediately, delayed, or dropped.

#### 2.3.3.2.2 Core functions

When a DS-marked packet arrives at a DS-capable router, the packet is forwarded onto its next hop according to the so-called **per-hop behaviour (PHB)** associated with that packet's class. The per-hop behaviour will influence the sharing a router's buffers and link bandwidth among the competing classes of traffic. A crucial tenet of the DS architecture is that a router's forwarding behaviour will be based only on packet markings, i.e., the class of traffic to which a packet belongs. In this respect, the differentiated service architecture obviates the need to keep router state for individual source-destination pairs - an important consideration in meeting the scalability requirement discussed above.
2.3.3.3 Interoperability issues

It is globally recognised that IntServ is an excellent medium for providing QoS in limited networks, but unsuitable for the Internet core. To the contrary, DiffServ was envisaged to scale well to large networks.

Thus, IntServ and DiffServ provide complementary solutions to the problem of providing QoS. Thus, these technologies should be able to coexist and interoperate. A possible model of Cupertino could be one in which DiffServ is used in core networks, while edge routers use IntServ. In such a manner, IntServ capable nodes connect to each other through a virtual link provided by the DiffServ transit network. A conceptual dissimilarity between the two strategies is that IntServ copes with a more quantitative approach of QoS, whereas DiffServ deals with it from a more qualitative aspect.

![IntServ - DiffServ interoperable network configuration](image)

**Figure 2.16: IntServ - DiffServ interoperable network configuration**

It is desirable that quantitative QoS is supported throughout the entire network, so the DiffServ infrastructure should shift to this direction. Although quantitative QoS applications are the minority in the application world, their requirements should be fulfilled to the greatest possible extent for reliability reasons. More tangible DiffServ services should be defined so that to be readily mapped to respective IntServ services. **Policing** might be a necessary step at the interface of the stub (IntServ) and transit (DiffServ) network.

Since transmitting and receiving hosts are attached to the edge IntServ networks, RSVP signalling is assumed to travel end-to-end transparently over the DiffServ transit network. Reference [IETF-2] discusses some mechanisms that serve this purpose without negative impact on core network performance. What is important is that it is not required for the core DiffServ routers to interpret RSVP messages.

The edge routers, which lie in the IntServ region, have dual functionality: they consist of a RSVP half and a DiffServ half, with full capabilities each. Border routers exist inside the DiffServ region and are just DiffServ capable. They implement the policing procedure and they don’t affect the DS-field, which is set by the hosts.

In an IntServ setting, quantitative service parameters known as “flowspec” are interpreted to a form that is meaningful to the underlying link layer. In a combined IntServ – DiffServ setting, the DiffServ network behaves as a virtual link layer, where the IntServ service description must be mapped to the DS field. There are two schemes that outline the mapping of IntServ service types to DiffServ ones:

- **Default mapping**
  
  In this scheme, the mapping is constant and standardised. The IntServ service type is mapped to an appropriate PHB that will command the suitable behaviour on the DiffServ part of the network. Since DiffServ features more granularity in service classes, it could be worthwhile to enrich the IntServ service request. This is implemented by adding qualifier objects to IntServ signalling messages that will carry the refined information.

- **Customer-specified mapping**
  
  Here, the service mapping is configurable by the customer, that is, the IntServ stub network. The edge router is responsible for compiling the content of a RESV message and determining the PHB code point that signifies the DiffServ service level. This code point is appended properly to the RESV message and pushed towards the sending end. From this point on, the sender starts marking outgoing packets with the particular PHB code point.
2.3.4. MPLS

IP networks offer unparalleled scalability and flexibility for the rapid deployment of value-added IP services. However, with the increasing demand and explosive growth of the Internet, carriers start to require a network infrastructure that can be dependable, predictable and can offer consistent network performance. Similarly to the solutions used in the traditional circuit switched communication networks, the adoption of traffic engineering policies and the provision of differentiated transport services seem to be the means required to satisfy the carriers requirement.

The Multi Protocol Label Switching (MPLS) networking technology being developed by IETF contributes to this end by introducing a new concept, the Label Switch Path (LSP), which enables IP datagrams with certain similarities to be modelled as one object that can be controlled by the network. By allowing the network to explicitly route a LSP, both the traffic engineering techniques and the provisioning of differentiated services are enabled in IP networks.

2.3.4.1 Architectural Point of view

In nowadays IP networks, a packet is forwarded from one router to the next until the packet reaches its final destination. Each router along the path makes an independent forwarding decision by analysing the (IP) packet header. The next hop for a packet is selected based (1) on the header analysis (e.g. IP destination address) and (2) on forwarding information that was obtained by previously run routing algorithms (e.g. RIP, OSPF, IS-IS or BGP-4). This approach is limiting the new integration role IP is expected to play in the coming years. First, the IP packet header, which contains much more information than the needed to simply select the next hop, is virtually unused. For instance, the ToS field that can be used to request differentiated services as suggested by the DiffServ IP extension. Second, the hop-by-hop destination–based forwarding constrains the services the network can provide. For example, it implies that packets with the same destination traversing the same point in the network will follow the same path from that point. This makes it difficult for the network architecture to support other services, such as providing paths that are specific to particular sources or to particular services at the destination.

MPLS overcomes these problems by combining the label-based technique already existing in other networks, e.g. the ATM virtual circuits identified by VPI/VCI addresses, with the traditional IP routing. The basic idea is to assign fixed length labels to IP packets at the ingress of an MPLS network domain, based on the concept of Forwarding Equivalence Classes (FECs). A Forwarding Equivalence Class represents a set of similar packets, such as the packets belonging to a flow demanding some quality of service from the network. In the MPLS domain, the labels attached to IP packets are used by the MPLS Label Switch Routers (LSR) to make forwarding decisions, without using of the original IP headers. In these routers, the label is used as an index into a table that specifies the next hop and a new label. The old label is then swapped with the new label and the packet is forwarded to its next hop. These fixed length labels, which are inserted between the IP and the data link-layer packet headers, determine the packet route across the network and are computationally less intensive and take less time to process by routers than the IP header counterparts.

The set of labels that are associated to a Forwarding Equivalence Class in the MPLS network effectively defines a Label Switched Path (LSP), that is, a virtual-circuit. By using Label Switched Paths in a manner similar to a connection-oriented architecture, MPLS can provide many of the same advantages of a connection-oriented network while still retaining the underlying efficiency and operation of datagram networks.

A Labelled Switched Path must be setup and labels assigned at each hop before traffic forwarding can take place. There are two kinds of Label Switched Paths based on the method used to determining the route: (1) control driven Label Switched Paths (also called hop-by-hop Label Switched Paths, LSP) and (2) explicitly routed Label Switched Paths, ER-LSPs. When setting up a LSP, each Label Switched Router determines the next interface for the LSP based on its IP forwarding table, which, in turn, was built, based on the traditional IP routing protocols, and sends the label request to the next-hop router. When setting up an ER-LSP, the route for the path is explicitly defined in the setup message, similarly to a circuit establishment. The setup message traverses all nodes along the specified route. At each node, a label request is sent to the next indicated interface.
Thus a LSP follows the path that a packet using default IP routing would have used. On the other hand, an ER-LSP may be defined and controlled by the network operator or a network management application to direct the traffic on a path independent of what is computed in IP forwarding.

IETF is currently standardising a new signalling protocol called Label Distribution Protocol (LDP) for setup and maintenance of LSPs. The Label Distribution Protocol is used mainly to overcome the problem of label local significance. It distributes the labels forming the LSP so that in each Label Switch Router the adjacent labels associated to the same LSP can be known. This association, as well as the IP prefix to label mapping, is carried out by an edge label node (a router interfacing with other nodes that are not MPLS-capable). Thus, the MPLS edge node exchanges routing information with non-MPLS capable nodes, locally associates and binds the addresses learned via IP routing to MPLS labels, and distributes the labels to MPLS peers. The forwarding table indexed by labels is constructed as the result of label distribution. When a Label Switch Router receives a labelled packet, it will use the label as the index to look up the forwarding table. Each forwarding table entry specifies how to process packets carrying the indexing label. This is faster than the process of parsing the traditional IP forwarding table in search of the longest match done in IP routing, and thus improves price/performance and scalability.

For setting up ER-LSPs, two approaches are being discussed: the constraint based routed LDP (CR-LDP), which requires a subset of LDP functionality enhanced to signal explicit paths, and extensions to the RSVP protocol.

### 2.3.4.2 MPLS and Traffic Engineering in IP networks

Traffic Engineering is concerned with performance optimisation of operational networks. In general, it encompasses the application of technology and scientific principles to the measurement, modelling, characterisation and control of traffic, and the application of such knowledge and techniques to achieve specific performance objectives. A major goal of Traffic Engineering in Internet is to facilitate efficient and reliable network operations while optimising network resource utilisation and traffic performance.

Datagram networks, such as IP, have exhibited poor efficiency because the only mechanism to redirecting traffic has been to change the link metrics presented in the interior gateway protocols (e.g., RIP, OSPF, IS-IS). This mechanism is awkward because changing the metric for a link can potentially change the path of all the packets that transverse the link.

MPLS provides better granularity when making this type of change because any particular LSP can be shifted from a congested route to an alternate route. This represents an efficiency improvement over the traditional operational methods for datagram networks because the network operator can run the network at much higher capacity under normal circumstances, secure in knowledge that before congestion occurs, some of the traffic can be deviated way from the congestion point. Further, the operator can make use of global optimisation algorithms to provide a mapping from the traffic demand to the physical links that could not otherwise be achieved using only local optimisation. The result is that the operator can achieve a much higher degree of link utilisation (efficiency) through the network and to provide differentiated services.

#### 2.3.4.2.1 Traffic Engineering using ER-LSPs

Although better scalability and faster packet forwarding performance are the most obvious motivations behind MPLS, attention is also focusing on traffic engineering for its obvious benefits in carrier networks.

The relevant aspect of MPLS is that it supports explicit connection control through the ER-LSPs introduced above, which are said to be the traffic engineering building blocks that can be manipulated and managed by the network administrators to divert traffic. Connection control permits the explicitly routed paths to be established through the MPLS network and make them independent of the traditional destination-based IP shortest path routing model.

When setting up an ER-LSP, the route for the LSP is specified in the “setup” message itself, and this route information is carried along the nodes the setup message traverses. All the nodes along the ER-LSP will follow the route specification and send the label request to the next indicated interface. While the hop-by-hop LSP
follows the path that normal IP routed packets will take, the ER-LSP can be specified and controlled by the network operators or network management applications to direct the network traffic, independent of the IP topology. Through ER-LSPs, a virtual circuit switching capability is superimposed on the IP routing model. In this way the ER-LSP may be used to achieve network traffic engineering capabilities in carrier backbone networks.

Two approaches have been debated in the MPLS working group to control (establish, terminate, re-route) ER-LSP, namely (1) extending the capabilities of the Label Distribution Protocol to include explicit paths via Constraint-based Routed LDP (CR-LDP) and extensions to the RSVP protocol (MPLS-RSVP). At an abstract level the functions of the CR-LDP and MPLS-RSVP are somewhat similar. They both allow a Label Switch Router to: 1) Trigger and control the establishment of a LSP between itself and a remote LER; 2) strictly or loosely specify the route to be taken by the LSP; 3) specify queuing and scheduling parameters to be associated with this LSP at every hop. The relative merits and demerits of these two schemes can be found in. It is sufficient to note that manageable traffic engineering and QoS control cannot be realised unless one of these two protocols is deployed.

### 2.3.4.2.2 Constraint Based Routing

The existing connectionless networks do not provide features that can easily balance loads, support dynamic failover to backup paths or re-optimise the traffic routes for better network resource utilisation and transport performance. Routing protocols, as currently defined, only advertise reachability information along with an administrative cost for the links, and thus uneven traffic distribution can be caused by these protocols because they always select the shortest paths to forward packets.

While MPLS provides a method to set up explicit paths and forward traffic on them, it does not address the issue of how to find paths with constraints. Because MPLS allows traffic engineering and explicit routing, there is keen interest in QoS routing that allows selection of routes subject to QoS requirements (and possibly additional constraints such as policy) instead of simply the least cost or shortest route found by traditional routing protocol.

In order to enable the computation of routes with constraints, IETF is extending the commonly used IGPs (OSPF and IS-IS) to carry additional information about links. The information includes maximum link bandwidth, maximum reservable bandwidth, current bandwidth reservation at each of eight priority levels, a default traffic engineering metric and the resource class or colour of the link.

Once this information about every link is available in the network, a path selection algorithm is needed. The source router is basically responsible for computing the complete path all the way down to the destination, and then initiating path setup using the signalling protocol. Thus, all routers in the network do not need to agree on the path selection algorithm and it is left to implementers to decide what path selection algorithm to use.

### 2.3.4.3 QoS in MPLS enabled networks

The problem of providing Quality of Service is solved on MPLS by establishing dedicated paths across the networks. Assigned with setup and holding priorities translated from DiffServ class-of-service levels, the paths essentially (1) reserve bandwidth and resources at the network elements for the traffic mapped to them and (2) can be associated to admission control mechanisms for shaping the traffic entering into the MPLS domain.

Since MPLS is primarily intended for IP over ATM networks, end-to-end QoS becomes a quite realistic scenario. The primary objective for MPLS, however, is to support the DiffServ QoS model. The ultimate benefit of MPLS QoS will be a unified or converged network supporting several classes of service.

### 2.3.4.3.1 Adding CoS and QoS

In MPLS, the data packets entering a MPLS domain are distributed among the various Forwarding Equivalence Classes by the LER devices of the network. Unlike conventional IP forwarding, this assignment of packets to a specific FEC is not based only on their address prefix but also can take into account additional information contained in the packet’s header. For example, packets can be assigned to a particular FEC based on the
combination of source/destination sub network, IP multicast group or a VPN identifier. Using this information, the network can establish different Classes of Service (CoS). Constraint based routing (CR) enables the network administrator to define the route and the Quality of Service (QoS) provided to each FEC. The MPLS header contains a 3 bit field (EXP field) that can be used for denoting the CoS of the packet, as shown in .

![Layer 2 Header](image)

**Figure 2.17:** IntServ - DiffServ interoperable network configuration

The network provider can take two approaches to support the MPLS-based Class of Service forwarding. The first approach is to provide Label Switched Routers with prioritised transmission queues and to assign the packets arriving through a LSP to these queues according to their CoS bits. The second approach is to provision multiple LSPs between each pair of edge Label Switched Routers, so that each LSP can be traffic engineered to provide different performance and bandwidth guarantees. The head end LSR can place high priority traffic in one LSP, medium priority to another, best-effort IP traffic in a third, etc.

The information about the explicit routes and the reservation of resources needed to achieve a certain QoS is distributed using the RSVP or CR-LDP protocols.

### 2.3.4.3.2 DiffServ and MPLS (end-to-end QoS)

One of the most straightforward possibilities for providing QoS in an MPLS domain is to map the 3 bits carried in the IP precedence of the incoming IP packet headers to a Label CoS field [TAG SWITCHING]. As IP packets enter an MPLS domain, the edge MPLS router is responsible for mapping the bit settings in the IP packet header into the CoS field in the MPLS header. As it has already been mentioned, several paths may exist from one end of an MPLS network domain to another, each of varying bandwidth and utilisation. Traffic labelled with higher CoS values could be forwarded along a higher-speed, lower-delay path, whereas traffic labelled with a lower CoS value could be forwarded on a lower-speed, higher-delay path.

When MPLS is combined with differentiated services and constraint-based routing, they become complementary abstractions for QoS provisioning in IP networks. In such architecture, LSPs are first configured between each ingress-egress pair. It is likely that for each ingress-egress pair, a separate LSP is created for each traffic class. DiffServ can support up to 64 classes while the MPLS shim label supports up to 8 classes. This shim header has a 3-bit field defined “for experimental use.” This poses a problem. This Exp field is only 3 bits long, whereas the DiffServ field is 6 bits.

There are two alternatives that address this problem called Label-LSP and Exp-LSP models. But they introduce complexity into the architecture. The DiffServ model essentially defines the interpretation of the TOS bits. As long as the IP precedence bits map to the Exp bits the same interpretation, the DiffServ model can be applied to these bits. In the case where additional bits are used in the DiffServ model, one can essentially use the label value to interpret the meaning of the remaining bits. Recognising that 3 bits are sufficient to identify the required number of classes, the remaining bits in the DiffServ model are used for identifying the drop priority and these drop priorities can be mapped into a Label-LSP in which case the label identifies the drop priority while the exp bits identify the Class that the packet belongs to.

When IP QoS is required in the network, the signalling protocol selected for ER-LSP must provide the support for differentiated services across the network. CR-LDP facilitates the integration of IP QoS to the core - it has been designed to support DiffServ and operator configurable QoS classes. Flexible mapping to existing ATM
service classes has also been specified. CR-LDP also provides the capabilities for strict and loose path setup with setup and holding priority, path pre-emption and path re-optimisation. CR-LDP signalling allows the specification of traffic parameters to express the packet treatment at each node, also known as per-hop behaviours (PHBs), for each differentiated service. It is this capability and the rules enforced at the edge, that provide a powerful and flexible QoS support for end-to-end differentiated services. In this way, CR-LDP has the flexibility to construct different services. Note that the specification of traffic parameters in CR-LDP is optional so, when they are not specified, ER-LSPs can be used to carry best-effort traffic.

### 2.3.4.4 Operational Point of View

The deployment of MPLS in the Internet has some profound consequences at the architectural level. It changes the basic forwarding model, which has remained essentially unchanged since initial deployment of the ARPAnet. In turn, it also impacts the routing architecture, requiring that routing protocols perform new and more complex routing tasks. While the initial deployment of MPLS will primarily provide benefits for traffic engineering, the many applications of MPLS within carrier network will provide many benefits in the years to come.

The exact ramifications of MPLS deployment are as yet unclear, but some of the immediate consequences are a more efficient transit core network, improved economy of scale, new connection oriented services and the ability for fast restoration of data traffic. Immediate applications are likely to be in intra-domain traffic, where a single network administrator drives the MPLS deployment. Over time, however, inter-domain MPLS deployment is likely to occur, where transit carriers could provide service to local or national ISPs.

From a pure functional point of view, MPLS can be modelled by the following basic functional components: 1) Path Management; 2) Traffic assignment; 3) Network state information dissemination; 4) Network management.

Path management concerns all the aspects related to the selection of explicit routes and the instantiation and maintenance of LSPs. A path management policy defines the path selection as well as rules for sustaining already established LSPs. Path management consists, for that reason, in the functions of selecting, placing and maintaining a path.

Traffic assignment consists on the assignment of traffic to a LSP previously established and concerns all the aspects related to the allocation of traffic which can be classified into two functions: portioning function and apportionment function. The first portions the ingress traffic according to some principle of division. The second lets the portioned traffic to establish LSPs according to some principle of allocation.

Network state information dissemination concerns the distribution of relevant topology state information through the MPLS domain. This is accomplished by extending conventional intra-domain gateway protocols to propagate additional information about the state of the network in the link state advertisements.

Network management is the key point in MPLS. The success of MPLS approach to traffic engineering introduced above eventually depends on the ease with which the network can be observed and controlled. Generally, an MPLS network management system includes a set of configuration management functions, performance and accounting management functions, and fault management functions. Collectively, these functions allow the state of managed MPLS objects (e.g., LSPs) to be acquired and their characteristics (and ultimately the network performance) controlled. Point to point traffic flow can be characterised by monitoring traffic statistics on LSPs. Path loss characteristics can be estimated by monitoring ingress and egress traffic characteristics at both points of LSPs and noting discrepancies. Sending probe packets through LSPs and measuring transit times can estimate path delay characteristics. Event notifications can be issued when the state of a managed MPLS object exceeds prescribed thresholds. Bulk retrieval of LSP traffic statistics can be used for time series analysis and capacity planning purposes. An operation requirement is the capacity to list, at any given point in time, all nodes traversed by a LSP and for each node to list to list all the LSPs originating from it, terminating from it or traversing it.
2.3.4.5 VPNs on MPLS networks

The deployment of MPLS in the Internet has some profound consequences at the architectural level. It changes the basic forwarding model, which has remained essentially unchanged since initial deployment of the ARPAnet. In turn, it also impacts the routing architecture, requiring that routing protocols perform new and more complex routing tasks. While the initial deployment of MPLS will primarily provide benefits for traffic engineering, the many applications of MPLS within carrier network will provide many benefits in the years to come.

The exact ramifications of MPLS deployment are as yet unclear, but some of the immediate consequences are a more efficient transit core network, improved economy of scale, new connection oriented services and the ability for fast restoration of data traffic. Immediate applications are likely to be in intra-domain traffic, where a single network administrator drives the MPLS deployment. Over time, however, inter-domain MPLS deployment is likely to occur, where transit carriers could provide service to local or national ISPs.

From a pure functional point of view, MPLS can be modelled by the following basic functional components: 1) Path Management; 2) Traffic assignment; 3) Network state information dissemination; 4) Network management.

Path management concerns all the aspects related to the selection of explicit routes and the instantiation and maintenance of LSPs. A path management policy defines the path selection as well as rules for sustaining already established LSPs. Path management consists, for that reason, in the functions of selecting, placing and maintaining a path.

Traffic assignment consists on the assignment of traffic to a LSP previously established and concerns all the aspects related to the allocation of traffic which can be classified into two functions: portioning function and apportionment function. The first portions the ingress traffic according to some principle of division. The second lets the portioned traffic to establish LSPs according to some principle of allocation.

Network state information dissemination concerns the distribution of relevant topology state information through the MPLS domain. This is accomplished by extending conventional intra-domain gateway protocols to propagate additional information about the state of the network in the link state advertisements.

Network management is the key point in MPLS. The success of MPLS approach to traffic engineering introduced above eventually depends on the ease with which the network can be observed and controlled. Generally, an MPLS network management system includes a set of configuration management functions, performance and accounting management functions, and fault management functions. Collectively, these functions allow the state of managed MPLS objects (e.g., LSPs) to be acquired and their characteristics (and ultimately the network performance) controlled. Point to point traffic flow can be characterised by monitoring traffic statistics on LSPs. Path loss characteristics can be estimated by monitoring ingress and egress traffic characteristics at both points of LSPs and noting discrepancies. Sending probe packets through LSPs and measuring transit times can estimate path delay characteristics. Event notifications can be issued when the state of a managed MPLS object exceeds prescribed thresholds. Bulk retrieval of LSP traffic statistics can be used for time series analysis and capacity planning purposes. An operation requirement is the capacity to list, at any given point in time, all nodes traversed by a LSP and for each node to list all the LSPs originating from it, terminating from it or traversing it.

2.3.5. Optical transport

2.3.5.1 WDM

The tremendous increase in the network traffic, mainly in the IP traffic, forced many network operators to seek for methods to increase the bandwidth of the existing fibres. Wavelength Division Multiplexing (WDM) is a technology that allows the bandwidth upgrade of existing fibres and can be smoothly integrated with the deployed SONET/SDH networks and equipment.
Its principle is the same as in frequency division multiplexing (FDM). Every signal is transmitted using different carriers, occupying non-overlapping parts of the available optical spectrum. Thus, various optical signals can be transmitted by a single fibre. Advances in the optical technology allow the multiplexing of more than 100 optical channels into the same fibre.

2.3.5.1.1 WDM Components

The main WDM component is the Optical Multiplexer, which allows the multiplexing of two or more optical signals into the transmission fibre. Optical amplifiers are used to restore the signal to the appropriate power, compensating for the system losses. Similarly to the SDH ADM, Optical ADM (OADM) allows the grooming or splitting of the optical signal along the transmission path. These devices allow the construction of more complicated networks topologies giving more flexibility. WDM ring can be constructed in a way similar to the SDH rings.

Optical Switch is a component of primary importance, allowing the switching of optical signals from a given input to a given output. Moreover, wavelength converters perform the translation of an incoming signal at a given wavelength into a different signal at a different wavelength. Optical switches with wavelength converters provide more flexibility in routing optical signals and permits the construction of optical networks with complicated mesh topologies.

Finally, Optical Crossconnects (OXC) are the key components for the construction of an optical transport network. Optical multiplexers/demultiplexers, optical switches and wavelength converters usually compose OXCs. Figure 2.18 depicts the architecture of an OXC node. The incoming optical signal is demultiplexing into several wavelengths that are sent to the optical switches. The optical switches route the wavelengths to optical multiplexers, where are multiplexing into one optical signal suitable for transmission. Wavelength converters can be used between the optical switches and the multiplexers, providing more routing flexibility.

![Architecture of an Optical Crossconnect (OXC) node](image)

**Figure 2.18:** Architecture of an Optical Crossconnect (OXC) node

2.3.5.1.2 Optical Transport Networks (OTNs)

Optical Transport Networks (OTNs) are WDM networks that provide transport services via lightpaths. A lightpath is simply a high-bandwidth pipe, carrying data up to several gigabits per second. An OTN is composed of several OXC, which are connected in a certain topology of choice. Lightpaths are setup and torn down either by the management system of the network or by means of signalling. The network should be also capable of monitoring the network elements, facilitating recovery from possible faults, etc.

Some interesting features of OTN are:
• Transparency. Lightpaths can transfer any variety of bit rates and protocols and can be considered as protocol insensitive. It allows the support of a variety of higher layers concurrently. Some wavelengths can carry SDH traffic, whereas others can carry ATM cells.
• Wavelength reuse. The number of the lightpaths that the network supports is greater than the number of the wavelengths as the wavelengths can be spatially reused.
• Reliability. The network should be capable in the event of a fault to reroute the affected traffic over alternative paths. A high degree of reliability is of vital importance for network operators.
• Virtual topology. Virtual topology is the topology that is seen by the higher layers. It is a graph consisting of network nodes, with an edge between the nodes if there is a lightpath between them.
• Circuit switching. The lightpaths can be established and torn down in a way similar to the circuit-switched networks.

The optical transport network is divided into 3 sub-layers: the optical channel, optical multiplex section and optical transmission section, as illustrated in Figure 2.19.

**Optical channel layer network**: This layer network provides end-to-end networking of optical channels for transparently conveying client information of varying format (e.g. SDH STM-N, PDH 565 Mbit/s, cell based ATM, etc.). To provide end-to-end networking, the following capabilities are included in the layer network:

– optical channel connection rearrangement for flexible network routing;
– optical channel overhead processes for ensuring integrity of the optical channel adapted information;
– optical channel supervisory functions for enabling network level operations and management functions, such as connection provisioning, quality of service parameter exchange and network survivability.

**Optical multiplex section layer network**: This layer network provides functionality for networking of a multi-wavelength optical signal. The capabilities of this layer network include:

– optical multiplex section overhead processes for ensuring integrity of the multi-wavelength optical multiplex section adapted information;
– optical multiplex section supervisory functions for enabling section level operations and management functions, such as multiplex section survivability.

These networking capabilities performed for multi-wavelength optical signals provide support for operation and management of optical networks.

**Optical transmission section layer network**: This layer network provides functionality for transmission of optical signals on optical media of various types (e.g. G.652, G.653 and G.655 fibre). The capabilities of this layer network include:

– optical transmission section overhead processing for ensuring integrity of the optical transmission section adapted information;
– optical transmission section supervisory functions for enabling section level operations and management functions, such as transmission section survivability.
2.3.5.1.3 Transport Services

The optical layer can be seen as a server layer that can carry different client signals (IP, ATM, SDH). From this point of view, the optical layer provides transport services to the upper layers. Three categories of transport services can be identified:

- Leased OCh service. An OCh can be setup either using the network management resulting in a permanent setup or using signalling and routing protocols, which is more flexible.
- Leased Wavelength. The customer of this service should be equipped with coloured line terminals.
• Leased Dark Fibre. In this case the provider of this service has no control of the fibre and the customer can use it according to its needs.

From a carrier’s point of view, the provision of lichgates to the customers can be classified into the following categories:

• Lichgates that must be protected in the optical layer. In this category the client layer relies on the optical layer for protection and no restoration/protection mechanism itself.
• Lightpaths that must not protect in the optical layer. An example in this category is SONET/SDH signals that relies on the protection/restoration mechanism of the SONET/SDH layer
• Lightpaths that are indifferent to protection. IP traffic fits in this category, since IP has its own mechanism for protection by rerouting the traffic around the affect segment of the network. However, protection at the optical layer is welcome since it takes less time the protection in the optical layer instead of the rerouting in the IP.
• Lightpaths that may be protected on a best effort basis. Here, protection is provided only when bandwidth is available in the optical layer
• Low priority lightpaths, that are used protection bandwidth and are pre-empted by protection of other lightpaths in the event of a fault

2.3.5.1.4 Routing

There are two categories of wavelength routing networks:

• Static, where the set of lightpaths are fixed
• Reconfigurable, where the set of lightpaths can be changed by changing the states of the OXC

Given the network topology and a set of end-to-end lightpath requests, the Routing and Wavelength Assignment (RWA) problem tries to determine the route and the wavelength for every request, minimising at the same time the number of the wavelengths. The RWA problem is tightly coupled with the virtual (logical) topology design. The later resolves the connectivity between different nodes, taken into consideration a set of restrictions like traffic demands matrices, nodal resources, policy constraints, over provision constraints, etc. However, it can be assumed that the virtual topology design generally deals with long-term concerns and it provides input to the RWA problem. The RWA problem itself can be further decomposed into two sub-problems: route selection and wavelength assignment.

The RWA problem is further complicated by different constraint such as wavelength continuity limited wavelength conversion capabilities, power limitations, etc, which are imposed.

Many optimisation schemes have been proposed (integer-linear, mixed-integer) which tries to minimise various cost function such as the number of wavelength, blocking probabilities, etc. Usually, a complete set of desired source-destination requests is given and routing decision are made for all source-destination pairs in order to achieve a global optimum solution. These solution requires complicated computation and as the network size increase they become excessive time-consuming. Therefore, they can be used only for static demands.

To cope with real time requests, which may arrive at random time, dynamic routing schemes have been proposed. These approaches lead to greater blocking probabilities, since channels are routing as they arrived. Channel rerouting is then needed in order to optimise the network utilisation by reallocating the network resources. Currently, there is no standardised RWA algorithm.

2.3.5.1.5 Protection/Restoration

Network survivability is an important factor in the design of a network and network operators take special actions in order to minimise the disruption of the network traffic in case of a fibre defect or an equipment failure. Taking into consideration that a fibre cut affects thousand of connections, the existence of a flexible, efficient and robust protection/restoration mechanism is of primary importance.
Protection in the optical layer has many similarities with the SDH/SONET approach and achieves fast restoration/protection within 50msec. Protection can be performed either at an optical channel level, protecting individual lightpaths or at the aggregate signal level, which corresponds to the OMS.

We can further distinguish protection at the link or the connection/path level. At the link level the traffic is diverted where the affected link is located, whereas at the connection level, the protection is occurred between the 2 end points of the path.

2.3.5.1.5.1 Link Protection/Restoration

In case of link protection the network will be protected at a link connection level. The protection or back-up path is disjoint from the working path at the particular link over which protection is required. When the protected link fails, traffic on the working path is switched over to the protection path. This is a local repair method that can be potentially fast.

At the OCh layer similarly to the SDH protection we have the following schemes (for ring topologies):

- OChDRRing, where each wavelength demand is protected by using main working path along one side of the ring and a protection path along the other side of the ring
- OChSPRing, where the protection channels in the other fibre protect the working channels in one fibre, travelling in the other direction around the ring.

Whereas at the OMS layer we have the following schemes (again for ring topologies):

- OMS DPRing, where one fibre on the ring is used for working traffic, the other fibre is used for protection traffic
- OMS SPRing, where in a 2-fiber OMS-SPRing, half of the wavelengths on each fibre are reserved as protection channels. In the event of a failure condition, the nodes adjacent to the failure will loop back all the affected lightpaths at once on the protection channels of the ring.

2.3.5.1.5.2 Connection Protection/Restoration

In case of network connection protection the endpoints of the network connection is protected. Under network connection protection, the protection path is completely disjoint from the working path. The advantage of network connection protection is that the protection path protects the working path from all possible link and node failures along the path, except for failures that might occur at the ingress and egress nodes. The Optical sub network connection Protection (O-SNCP) belongs to this category.

2.3.5.1.5.3 Linear Protection/Restoration

For linear configuration, there are the following protection schemes at the OMS level:

- 1+1 protection
- 1:1 protection
- 1:N protection
- M:N protection

2.3.5.2 Differentiated Optical Services

Quality of Service in a network capable of supporting terabits of data is a controversial issue. Many believe that there is no need at the time to spend effort for developing an effective mechanism for supporting QoS in the optical networks. However, as WDM becomes a mature technology and many problems are tackled successfully the support of different optical quality between a single source destination pair becomes important. For example the existence of different protection schemes, one for high priority traffic and the other for low priority traffic gives greater flexibility to operators and more cost effective approaches can be followed.
Moreover, intense research for integrating the IP layer with the optical layer is carried out, since IP seems to be the dominant protocol for data transmission. For a smoothly integration of the two worlds the issue of the QoS should be studied as well.

In the IP community, the issues of QoS have been studied extensively and different QoS model has been proposed providing the necessary support for quality to application that requires more than the traditional best effort service. IntServ and DiffServ are two well-known models that have been studied extensively. The IntServ model can deliver packets belonging to a certain traffic flow with a guaranteed QoS. Each packet is processed to determine its class. IntServ use a signalling protocol to reserve resource along the path. RSVP is one of the signalling reservation protocols that can be used for that purpose. However, the major drawback of the IntServ framework is that each router should maintain and management thousands of flows, thus making the IntServ approach unable to scale in large networks.

DiffServ was born by the scalability problem of the IntServ model. DiffServ aggregates packets with similar QoS requirements to coarser-grained flows, thus it is more scalable that the IntServ model. Moreover, the existence of a resource reservation protocol like RSVP is not needed since flows are enforced locally on a per-hop basis.

On the other hand, only lately the issue of QoS in the optical network has started to be studied. The idea of a QoS aware optical network shifting ideas from the existing IP models is appealing. In [GHANI] a framework for aggregating the DiffServ flows into coarser-grained flows, which can be directly mapped into optical channels, is proposed. The quality of the DiffServ flows must matches the quality providing by the optical channels.

### 2.3.5.2.1 Optical Differentiated Services Model

Scalability and especially at high rates becomes an important design issue of an optical QoS model. Computation at the edge of the optical network must be kept as light as possible. The DiffServ framework, with the appropriate modifications is a major candidate since scalability was a key role in its design.

Figure 2.20 depicts an optical QoS aware domain with the appropriate interfaces between the IP sub-networks. The optical network has a mesh topology and is composed by OA, OADM and OXC. At the edge of the optical network, there is electrical to optical conversion. The latter is needed since for the calculation of some parameters like BER electronic processing is required.

The interface between the Optical network and the IP sub-networks implements the necessary QoS aware functions for aggregating and mapping the DiffServ flows into equivalent optical flows. Optical flows have capacity equal to the commercial available capacity (e.g. OC-12, OC-48, etc).

![Figure 2.20: Interconnection of DiffServ-aware IP sub-networks with a QoS-aware Optical Network](image)

### 2.3.5.2.2 Service Parameters

Each differentiated optical service class is defined by a set of parameters that characterised the quality of the optical signal. The parameters can be either quantitative such as delay, BER, jitter, bandwidth or based on functional capabilities such as monitoring, protection, etc. In the following subsection some important parameters are further analysed.
2.3.5.2.2.1 Lightpath Properties

Every component in the route of a lightpath introduces some impairment such as jitter, wander, etc. Some of these impairments are accumulated, resulting in degradation of the quality of the signal at the destination. Such physical effects permit only a limited number of network elements to be cascaded in order to comply with the performance figures of the service class. Thus, the quality of a lightpath is heavily dependent on the quality of the components along its path. Defining thresholds for every parameter (BER, jitter, etc) allows the association of these thresholds with a certain class of optical service. For example in the case of the BER, the definition of multiple thresholds can be used for the classification of the offered optical service into classes of service. Thus, class 1 can have BER less than $10^{-10}$ and class 2 can have BER between $10^{-10}$ and $10^{-8}$.

2.3.5.2.2.2 Lightpath Protection

SDH as well as existing WDM networks provide protection equally to all connections. Protecting all the connections is not a cost-effective approach since not all connections require protection. The classification of the flows based on the desired level of protection gives great flexibility and permits the implementation of different survivability schemes for every class of service. DiffServ flows that require a certain degree of protection belong to a certain class of Optical QoS.

In case of optical network failures, the optical network should firstly restore high-priority lightpaths. That same also applies in cases where there are no sufficient network resources and decisions have to be made about which lightpath should be restore.

2.3.5.2.2.3 Lightpath Supervision

Lightpath supervision is the process of monitoring the integrity, validity and quality of a given trail. The monitoring capabilities should be offered on a class-based basis.

The following types of monitoring techniques are considered

- Processing the overhead section of the lower layer performs inherent monitoring. For example in case of an IP client layer, processing the overhead section of the OCh layer performs inherent monitoring. Performance information cannot be obtained by inherent monitoring since existing overhead section does not provide the appropriate performance data.
- Non-intrusive monitoring is achieved by listening the original data and the overhead as well.
- Breaking the original trail and introducing a test trail perform intrusive monitoring. It is mainly used for testing fibre continuity and for fault localisation.

2.3.5.2.2.4 Transparency

Transparency is depending on the signal regeneration. Three type of regeneration exists

- 1R regeneration performs amplification and equalisation (frequency and dispersion)
- 2R regeneration includes the operations of 1R plus digital reshaping and noise suppression
- 3R regeneration includes the operation of 2R plus reshaping and retiming. 3R regeneration eliminates transparency.

At present, there is no simple optical performance monitoring techniques without regeneration available.

2.3.5.2.3 Optical Services Mapping

The optical services are access via the interface between the optical QoS aware network and the IP sub-network. At the boundary between the optical and the electrical domain the following functions are performed:

- Aggregation of the incoming DiffServ flows into bigger flows with capacity equals to the commercial available (e.g. OC-12, OC-48, etc)
- Mapping of the coarser-grained flows into the following defined service classes which have equivalent QoS characteristics
• Admission control and policing function to ensure that there are always enough resource in the optical network to support the requested services.

Figure 2.21 depicts the mapping between IP DiffServ flows and optical classes in a border OADM. The set of wavelengths assigned in each optical class are presented as well.

![Figure 2.21: Mapping of IP DiffServ flows to Optical Classes](image)

It is assumed that after the mapping of the DiffServ flows to optical flows is done, the optical network should be capable to enforce the agreed parameters using signalling as well management functionalities. For example through the signalling entity, end-to-end set-up requests can be initiated, allocating the necessary resources along the path.

The following table depicts 3 classes of optical services. The quality of the optical service for lightpaths belonging to the same class is the same.

![Table 2.3: Example of classes of optical services](image)

### 2.3.5.3 MPLambdaS

#### 2.3.5.3.1 Introduction

The MPLambdaS framework was an attempt to bring the MPLS framework into the optical layer. Since the IP layer appears to be the dominant layer for the transport of data, the motivation behind the MPLambdaS was to shift the concepts of the label switching to the optical layer. As the MPLS was used to perform fast forwarding of layer 3 flows, the lambda approach can be used to perform layer 1 forwarding of layer 3 flows.

The concept of the Label Switched Path (LSP) is now applied to the optical lightpath channels and the MPLS labels are analogous to the MPAS wavelengths. The MPLS control plane can be now attached to the Optical cross-connects/ Wavelength router/switches (OXC/WRS) devices, which are termed as optical lambda switch
router (O-LSR) nodes. However, there is a difference between the electrical and the optical MPLS nodes, since the optical node can perform only a subset of the operations that the electrical nodes can do.

When a connection using IP-MPLS traverses a full optical network where MP\(\lambda\)S is implemented, there are no significant changes because the wavelength(s) used at the optical network can be seen as a label added to the label stack and OXC with MP\(\lambda\)S control plane has also assigned an IP address. One difference may be that the label is not stacked over the other labels but it is the wavelength itself used at the transmission.

### 2.3.5.3.2 Technical Approach

Similar to the electrical MPLS, the forwarding at each O-LSR node is done based on the label of the incoming optical LSP. The MPLS forwarding routes can be specified via two different methods: hop-by-hop and Explicit Routing (ER). Hop-by-hop allows each LSR to independently choose the next hop (as existing IP), whereas ER pre-determines the next hop of each LSR. In ER, the routes are chosen for each label and not for each packet, what increments the efficiency of the method.

Electronic MPLS performs different operation in the incoming label. These operations are depicted in Figure 2.23 and include:

- Label add-drop. It is performed to incoming LSP termination or outgoing LSP origination
- Label swapping. The label swapping operation involves label read and write operations (new label retrieved during lookup operation)
- Label merging. It is done by combining two or more labels into one aggregate LSP label
- Label stacking/de-stacking. Label stacking/de-stacking is an important operation since it allows traffic flow aggregation. Two smaller LSP, smaller in terms of bandwidth, can be stacked together to a larger LSP. This is similar to the ATM concept of the VP/VC.
2.4. Integration of IP and WDM.

2.4.1. IP over ATM over SDH for WDM transmission

There are many flavours of IP over ATM, e.g. classical IP over ATM, LAN emulation, Multiprotocol over ATM. For long-haul transport over WDM, the most standard transmission format currently is to use the SDH frame. Figure 2.24 shows a possible IP/WDM network architecture that uses IP over ATM over SDH encapsulation.
Figure 2.24: Example of IP over ATM over SDH encapsulation for transport over a WDM network

In this scenario, IP packets are segmented into ATM cells and assigned different Virtual Connections by the SDH/ATM line-card in the IP router. The ATM cells are then packed into an SDH frame, which can be sent either to an ATM switch or directly to a WDM transponder for transport over the optical layer (simplified in Figure 2.24 to a single OADM ring).

At present, one of the ways of trying to ensure a given QoS for an IP service is to guarantee a fixed bandwidth between pairs of IP routers for each customer (layer 2 QoS management). ATM provides a way to do this with variable granularity by the Permanent Virtual Channels (PVC) using the ATM management system or Switched Virtual Channels (SVC) dynamically set-up, all within Virtual Paths (VP). It can also use statistical multiplexing to allow certain users to access extra bandwidth for short bursts. This can help to guarantee a fixed and arbitrary bandwidth from less than 1Mb/s to several hundred Mb/s to many different customers. In addition the fine granularity can enable IP routers to be connected into a logical mesh easily, thus minimising delays from intermediate routers. One other benefit in using ATM protocol is the possibility to have differentiated traffic contracts, which offer various quality of service depending on the application requirements. For IP traffic, which is by essence connection-less, the UBR (Unspecified Bit Rate) traffic contract is mainly used within ATM networks. Nevertheless, if IP applications require a particular QoS, especially for real-time constraints, it is possible to use other ATC (ATM Transfer Capability) such as CBR (Constant Bit Rate) or VBR-rt (Variable Bit Rate – real-time) which are suitable for real-time applications. However, mapping variable length IP-packets on fixed length ATM-cells impose additional overhead if fragmentation of the packet is necessary, this is sometimes called cell tax. Difference in size may also raise the need to pad empty space in the cells, which also gives additional overhead. One solution to prevent padding is to put packets directly after one another, but this means a potential risk of loosing two consecutive packets in case of cell-loss. IP over ATM can also be used as an implementation of MPLS. In this situation, the PVCs are not set up by intervention of the ATM management system, but dynamically by the MPLS protocol. For MPLS based on ATM, the label can be stored in the ATM VCI.

2.4.2. IP over ATM directly on WDM

It is possible to have a scenario where ATM cells are transported directly on a WDM channel. This scenario is identical to the previous one, from an architectural point of view. The only difference is that ATM cells are not encapsulated into SDH frames, instead they are sent directly on the physical medium by using an ATM cell-based physical layer. Cell-based physical layer is a relatively new technique for ATM transport. Cell-based physical mechanisms have been developed specifically to carry the ATM protocol; this technique cannot support any other protocol except if these protocols are emulated over ATM. Some benefits of using a cell-based interface instead of SDH as described in the previous section are:

Simple transmission technique for ATM cells as cells are directly sent over the physical medium after scrambling.
Lower physical layer overhead (around 16 times lower than SDH)

As ATM is asynchronous, there is no stringent timing mechanism to be put on the network. However, the drawbacks are that the overhead i.e. cell tax is the same as for transport on SDH, the technology has not been endorsed by the industry yet and this transmission technique can only carry ATM cells. ATM cell-based physical layers are defined by several standardisation bodies, 155 Mbit/s and 622 Mbit/s in ITU and recently the ATM Forum completed the specification for 622 Mbit/s and 2488 Mbit/s.

2.4.3. IP over SDH, Packet over SONET (POS)

A network transporting IP in an SDH frame over WDM is shown in Figure 2.25.

![Figure 2.25: example of IP over SDH over WDM network](image)

It is possible to simply use SDH formats to frame encapsulated IP packets for transmission over WDM, probably using a transponder (wavelength adapter), or it is also possible to transport the SDH-framed IP over an SDH transport network along with other traffic, which may then use WDM links.

SDH can currently be used to protect IP traffic links against cable breaks by automatic protection switching (APS) in a variety of guises. The line-card in the IP router performs the PPP/HDLC framing. The optical signal is then suitable for transmission over optical fibre either into an SDH network element, a neighbouring IP router, or a WDM transponder for further transmission. There are also different types of IP over SDH interfaces:

- VC4 or Concatenated VC4 “fat pipes” which provide aggregate bandwidth without any partitions between different IP services which may exist within the packet stream.
- Channelised interfaces, where an STM16 optical output may contain 16 individual VC4s, with a possible service separation for each VC4. The different VC4s can then also be routed by an SDH network to different destination routers.

The version of IP over SDH examined here uses PPP encapsulation and HDLC framing. This is also known as POS or Packet over SONET. PPP is a standardised way to encapsulate IP and other types of packets for transmission over many media from analogue phone lines to SDH, and also includes functionality to set up and close links (LCP). HDLC is the ISO-standardised version of SDLC, a protocol developed by IBM in the 1970s. The HDLC framing contains delimiting flag sequences at the start and end of the frame, and also has a CRC checksum field for error control.

2.4.4. IP over SDL directly over WDM

Simple Data Link (SDL) is a framing method proposed by Lucent Technologies Inc, and can replace HDLC framing for PPP-encapsulated packets. Compared with the HDLC frame the SDL frame has no delimiting flag sequences. Instead the SDL frame is started with a packet length field. This is advantageous at high bit rates where synchronisation with the flag sequence is difficult. The SDL format can be inserted into an SDH payload for transmission over WDM or SDH equipment. The SDL format can also be encoded directly onto an optical carrier: SDL specifies the bare minimum functionality to be able to do this.
It uses a 4-octet header, which includes the packet length, as shown in Figure 2.26. The packet can be up to 65535 octets long. Additional error checking codes (CRC-16 or CRC-32) can optionally be used for the packet, and which may be placed after the packet. All bits except for the header are scrambled using a x48 scrambler. The sender and receiver scramblers are kept synchronised by occasional transmission of special packets.

![SDL header structure](image)

**Figure 2.26**: SDL header structure

SDL does not include any extra bytes dedicated to protection switching protocols (like the K1 & K2 bytes of SDH). The use of the optional payload CRCs could enable bit error rate monitoring.

### 2.4.5. IP over Gigabit Ethernet for WDM

The new Gigabit Ethernet standard can be used to extend high-capacity LANs to MANs and maybe even WANs, using Gigabit line-cards on IP routers, which can cost 5 times less than SDH line-cards with similar capacity. For this reason, Gigabit Ethernet could be a very attractive means to transport IP over “metropolitan” WDM rings, for example, or even over longer WDM links. Furthermore, 10Gbit/s Ethernet ports are likely to be standardised in the near future.

Figure 2.27 shows an example of an IP network based on Gigabit Ethernet interfaces. The Gigabit Ethernet line-cards may be used on IP routers only, or fast layer 2 Ethernet switches may also be used to network several IP routers together.

Lower bit rate Ethernet networks (e.g. 10Base-T or 100Base-T) have been used a lot in a half-duplex mode, where the bandwidth available for transmission is shared between all users, and between both directions of transmission. To police access to the shared bandwidth, CSMA-CD is used. This imposes limits on the physical size of the network, where the transit time cannot exceed the “slot time” which is the minimum frame length (512 bits for 10Base-T and 100Base-T). For a bit rate of 1Gb/s, using a minimum frame length of 512 bits would imply an Ethernet network only roughly 10m long, and for this reason the minimum frame length has been redefined to be 4096 bits for Gigabit Ethernet. However, this still limit the network size to 100m, so full-duplex mode is more attractive when using Gigabit Ethernet.

When Gigabit Ethernet (1000Base-X) is used in full-duplex mode, it becomes simply an encapsulation and framing method for IP packets, and the CSMA-CD functionality is not used. Ethernet switches can also be used to extend the network topology beyond a point-to-point link.

![Example of IP being transported over a WDM ring with Gigabit Ethernet framing](image)

**Figure 2.27**: Example of IP being transported over a WDM ring with Gigabit Ethernet framing
Gigabit Ethernet provides some support of CoS as defined in the standards IEEE 802.1Q and 802.1p. These standards facilitate CoS over Ethernet by providing a means for “tagging” packets with an indication of the priority or class of service desired for the packet. These tags allow applications to communicate the priority of packets to internetworking devices. RSVP or DiffServ support may also be achieved by mapping into 802.1p service classes.

2.5. Business Drivers for Inter Domain Management Systems

2.5.1. Technical Analysis

2.5.1.1 Single Technology Management Systems (IP, ATM, SDH, WDM)

2.5.1.1.1 Overview

Next generation networks built by the service providers will span multiple transport technologies. The choice of the backbone transport protocols depends on the offered mix of services. So, for example, public multi-service providers with customer Service Level Agreements will utilise a full stack of technologies, while National Internet Service Providers will probably choose direct IP over WDM networks. Figure 2.28 illustrates a number of network technology combinations that will be utilised to provide voice, data and video services by different types of service providers.

Figure 2.28: Multi-Layer and Multi-Service Network
The equipment in these networks will be provided by more than one vendor. This situation creates network environments where one technology is provided by one vendor and another technology by a different vendor. Technology providers supply element and network management systems to manage their technologies, causing a creation of a “smoke-stack” network management environment to the service providers. For example, the TDM voice network and its Operation Systems (OS) can be regarded as one domain, while an ATM data network and its related OSs can be regarded as another domain.

The network management situation is further complicated by multi-vendor support within a single technology domain (i.e. SDH), and service provider business needs to partition the management of their growing network. Thus the definition of management domains is driven by the mix of technologies, vendors, and business needs present in the given service providers environment. Lack of integration and sheer complexity of the tools themselves has become a barrier to the development of new applications as well as the exchange and sharing of data captured by these individual network management and provisioning tools.

This document describes the current situation in most national incumbent operators’ networks where different management systems exist for the different available technologies. In most cases a national network consists of multi-technology, multi-vendor equipment and corresponding management systems capable of managing in most cases the equipment of the specific vendor providing the equipment. Below the single domain management systems briefly describing the different management layers and functionality taken from real situations (incumbent national operators in Greece, Spain, etc) are presented, making evident the need of an Inter Domain Management System.

### 2.5.1.1.2 SDH Management Systems

Usually the network consists of multi-vendor equipment. Also different types of elements are present with different capabilities as terminal multiplexers, add/drop multiplexers, digital cross-connects etc. Each vendor provides the corresponding management system, covering Element Management Layer (EML) functionality of the TMN pyramid (Element Managers) and potentially also the Network Management Layer (NML) functionality (Network Managers). Sometimes the Element managers provide a sub-network view module, being able to visualise a whole sub-network. In other words it provides a partial Network Management Layer view. It is usual that each vendor group of equipment covers a specific geographical area. The EMs have complete configuration management capability. Any cross-connection in any of the managed types of network elements can be performed through the EMs. In addition, fault and performance management functionality is supported, mainly dealing with alarm reporting, filtering and logging relevant to fault and performance data.

The situation and structure of the Network Management Layer (NML) is similar to that of Element Management Layer (EML). The NML is capable of having a network view for each of the vendors’ equipment and able of creating end-to-end paths inside geographical area that a single vendor is covering. The NMs are able to configure and activate circuits and paths. Circuits are selected by using a variety of algorithms and rules. In order to create an end-to-end path crossing the different geographical/multi vendor areas a special order should be delivered to one of the management platform vendors. If all the vendors provide northbound interfaces in their NMSs which comply to specific standards then it might be possible to integrate all the vendors in one platform, but usually with mediocre results and semi-automated procedures. As far as network layer fault and performance management functionality, NMSs usually support additional functions such as fault correlation, analysis, localisation and recovery and performance analysis and threshold-cross reporting for end-to-end connections.

In the Service Management Layer, provisioning of SDH connectivity, evaluation and automated process of requests for connectivity and service monitoring is realised through the above integrating process, if available. Otherwise it is applicable only for a specific vendor and semi-automated procedures need to be adopted to provide end-to-end connectivity spanning multiple vendors’ equipment.

### 2.5.1.1.3 ATM Management Systems

The above cases, illustrating multi-vendor equipment inside a national network, are exactly the case for ATM networks, especially for the incumbent operators. The management information is sent in-band across the ATM network and then to the supplier specific EMS from one single node for each supplier. A separate Data Communications Network (DCN) will be used for remote terminals of the EMS management applications and if necessary to get management information from additional nodes. Dial-up or Ethernet ports on each ATM NE may be used by the EMS or NMS to reach an NE if there is a communications failure. The NMS must be
capable of establishing a dial-up connection to each NE so that communications to the NEs may be re-established in cases of EMS failure. Similar functionality to the SDH management systems in the different management layers is supported by the ATM management systems regarding configuration, fault and performance management; hence it is not reported once more.

2.5.1.4 WDM Management

Similarities especially to the SDH case, illustrating multi-vendor equipment inside a national network, have been reported for WDM networks. The only difference is that the vendors have shown significant efforts in integrating the SDH Management systems with the WDM management systems. As a first step, a light integration in management systems is performed by encapsulating the WDM management functionality in the SDH management systems. In other words, a common look and feel environment is given for both platforms under a single umbrella system. This integration can be considered high-level integration, incorporating 2 management systems in one platform, with common look and feel. The functionality integration, e.g. providing SDH connectivity in a mixed SDH/WDM network creating a mixture of SDH and optical paths, is still under development in most cases.

2.5.1.5 IP Management Systems

For many years IP networks has evolved deploying a complete set of protocols and applications suitable for delivering best effort traffic. Management wasn’t a priority since networks relayed on very powerful control-plane protocols to re-configure themselves in response to the addition of new elements or in case of failures. In this context, Simple Network Management Protocol has been developed with monitoring capabilities in mind, but with poor support for configuration management.

The above cases, illustrating multi-vendor equipment inside a national network, can be the case for IP networks, especially for the incumbent operators. Nevertheless, competitive operators, which are more flexible, usually have greater homogeneity in their national networks. In addition the SNMP protocol dominates the management of IP equipment, giving the opportunity to management platform vendors to be able to manage multi-vendor equipment, provided that the corresponding MIBs are supported or can be loaded upon demand. Despite the fact that the philosophy of the IP networks was connectionless and up to now QoS was not supported, IP Management Systems provide complete monitoring of service provisioning, fault and performance data collection covering more or less the three management areas relevant to WINMAN.

2.5.1.6 Conclusion

As a conclusion, integration is starting to appear for a single vendor among its different technologies products, such as SDH/WDM integration. Nevertheless this is a high level integration, providing only a common graphical interface for two technologies, than a real functionality integration in terms of configuration, fault or performance management. The latter is either not yet performed in commercial products or still under the development process. In the next chapter future trends in the integration of multi-domain management systems will be elaborated, illustrating the efforts in providing standard management interfaces in all technology domains towards the service management layer (northbound interfaces of the Network Management Layer) and the need for an Integrated Network Management System (INMS). The framework of WINMAN effort in this direction will be also presented.

2.5.2 Future Trends in IP- and Inter Domain- Management Systems

2.5.2.1 Overview

Nowadays a series of changes in the telecommunication network services industry are placing unprecedented demands on service providers. These changes include among others: De-regulation. De-regulation demands organisational flexibility. That, in turn, requires integration and uniformity of procedures to allow personnel to move from one assignment to another with a minimum of training. New entrants to the industry compete with each other to obtain skilled personnel. This “back-office”
competition places demands upon the management systems to help to concentrate rare skills into centralised operations centers while reducing the need for training in many other jobs.

**Competition.** There is tremendous pressure on service providers to attract and retain new customers. The market demands shortened intervals for the deployment of new services and for the delivery of existing services. It demands that the service representatives be well informed and proactive with respect to network problem resolution. Service quality has become a differentiator in the marketplace. Competition places new demands upon the management systems to collect and organise information concerning the quality of services provided.

**Technology and Services.** The rapid evolution of telecommunications technology places increased demands upon the operations staff to integrate and utilise new techniques for delivering end-user (customer) services. This, in turn, places new demands upon the management system to evolve rapidly and to reduce the cycle time needed to accommodate new network elements (telecommunications equipment). Management systems must support the industry trend toward more “intelligent” network elements by providing applications that operate at a level of functional abstraction that transcends the technical details that are increasingly managed by the elements themselves.

The challenge for the Network Operators/Transport Service Providers is to manage networks in a way, which allows the various technologies from the different vendors to inter-work in harmony and to make the differences in a heterogeneous network environment transparent to the user. And in order to be competitive in this market they must have the capabilities to provide a broad range of services to the end-customers at a competitive price with a proven quality of service in the shortest possible time.

To surmount these challenges, service providers are asking for a solution to manage these diverse and large-scale networks, which are based on ATM, SDH, WDM and IP. The solution is an Integrated Network Management System that manages the network across the different technology/vendor domains.

This means that these providers need an advanced Operations System that supports them in providing

- Best in class provisioning times,
- Guaranteed service subscriber service levels,
- Tools to operate their network effectively,
- Tools to provide the services cost effectively,
- Efficient utilisation of the network resources (return on huge investments)
- Reduction of required human intervention and thus reduction of labour costs
- The flexibility to evolve with the rapid changes in the market
- Reduction of the network management complexity

The advantages of an Inter-Domain Management solution for service providers/network operators are:

**Flexibility**

- Integrated support in terms of Configuration, Performance and Fault Management for multiple technologies and multiple vendor domains
- Common presentation layer
- Modular design of inventory, end-to-end design, implementation and the logical view permits 3rd party applications to be integrated at the data level.

**Extendibility**

- Scaleable architecture for inter-domain services.
- Improved life cycle costs through the transition of existing architecture into Inter-domain Management Architecture.
- Cope with the ever increasing complexity of their networks: The solution must be able to evolve rapidly and reduce the cycle time needed to accommodate new network elements
- Ability to incorporate new domains whether by technology or vendor.

**Cost control**

- Reduce duplication of functionality amongst products to promote current and future integration.
- Manage their diverse and large-scale networks in a centralised operation centre
- Integrate and uniform their internal procedures: Less skilled personnel needed
- Interval reduction in design, implementation, fault isolation and correlation due to the single Inter-Domain access point.
Competitive advantages

- Single view of network data across multiple domains/technologies supports integration of OSs. This benefits Service Providers regarding work realignment issues that may occur during deregulation and mergers and customers by providing an abstract, service-ready view of their network.
- Deliver their services in a timely manner: Automated procedures
- Deliver their services with quality, because competition is severe
- Customers receive improved service-impact analysis.

In this direction the efforts of the different manufacturers, standardisation organisations and the WINMAN solution will be elaborated hereafter.

### 2.5.1.2.2 Trends in IP management systems

Nowadays, IP networks support a diverse range of services, including VPNs, extranets, application hosting, e-commerce as well as voice and video traffic, and the range of services is growing daily. These new IP services demand more and more network resources.

There is a strong trend among ISPs to move away from being a generic service, and become a Network SP (NSP) and/or Application SP (ASP).

Another trend is to provide VPN service, and other specialised services. At the same time, traditional Telcos are trying to figure out how to cope with the network convergence phenomena. Providers need to be able to guarantee consistent and predictable business level quality of service over IP networks to be successful in these transitions.

On the enterprise side, a typical corporate network has wide variety of applications and user groups competing for service on the same physical network. Mission critical applications such as trading, e-commerce, ERP, and VoIP are competing with less important but resource-consuming web browsing, downloads, FTPs, e-mails, etc. Furthermore, everybody receive the same service level, without user differentiation.

#### 2.5.1.2.2.1 Policy Based Network Management applications

The obvious approach of over-provisioning to sustain Quality of Service is clearly not acceptable. Network resources have become more valuable and need to be managed efficiently to deliver maximum performance. That’s why new management frameworks are being developed to cope with this problem.

Policy Based Network Management applications are devoted to solve these new challenges. They provide management of Quality of Service (QoS), based in the COPS (and/or CLI or HTTP) protocol for Provisioned and Signalled QoS, Security management, and MPLS support. Many of these applications support IP-VPNs based in BGP/MPLS (draft-rosen-rfc2547bis-02.txt).

Another common feature is modularity and integrability, for example via CORBA-based APIs. The DEN/CIM model is also being supported in some of these software suites.

In order to illustrate the above, real world applications from three different manufactures are presented here: Nortel, Orchestream and Cisco.

#### 2.5.1.2.2.1.1 Nortel

Nortel has two different management applications product lines: Preside for Service Providers and Optivity for Enterprise.

The most relevant for WINMAN purposes is Preside, in particular the subset Preside Policy Services. This IP network service management application integrates quality of service policies, access policies, and security policies.

Preside Policy Services is directory-enabled. It provides centralised data management for the domain name server (DNS), dynamic host configuration protocol (DHCP), Quality of Service (QoS) policy, RADIUS and IP VPN provisioning servers.

The modular architecture permits service providers to choose the combination of functionality that suits their needs.
The component applications and products that comprise Preside Policy Services include the following:

- **Management console**: The management console is the graphical user interface used by the root admin user or customer admin user to view and manage the IP networks. Authorized users can access the management console remotely using JAVA-enhanced Internet browsers.

- **Application server**: The application server manages the user interface, and the data communications between the user interface and the data repositories. This communication includes:
  - Processing user requests for display of information from the data repositories.
  - Passing input information to the repositories for storage, such as imported DNS files.

- **Directory**: Preside Policy Services uses the directory for central storage of quality of service policy information, Radius user profiles and VPN data. The QoS, configuration and Radius server communicate with the directory using Lightweight Directory Access Protocol (LDAP).

- **Database**: The central repository for DNS and DHCP information is the database. The application server and server manager communicate with the database in Structured Query Language (SQL) to provide data updates, or to request stored information required by a control-plane server.

- **Server manager**: The server manager controls communications (messaging and data transfer) between the servers (for example, DNS server, DHCP server) and the data repositories. The server manager has an alternative server manager with a backup function.

---

**Figure 2.29**: Nortel System Architecture

![Diagram of Nortel System Architecture]

**Legend**:
- PMC: Preside management console
- LVP: LDAP validation proxy
- DNS: Domain name server
- RADIUS: Remote authentication dial-in user service
- SQL: Structured query language
- TCP/IP: Transmission control protocol/Internet protocol
- DHCP: Dynamic host control protocol
- VPN: Virtual private network
- QoS: Quality of service policy
- NAS: Network access server
- LDAP: Lightweight directory access protocol
- CLI: Command line interface
- COPS: Common open policy service

---
• DNS server: The DNS server responds to domain name requests from clients. DNS servers are configured as either primary or secondary servers, where the primary server is authoritative for the zone information.

• DHCP server: The DHCP server manages IP address requests from clients. The DHCP server has a backup server that is configured and provisioned in the Preside Policy Services GUI. 

  Note: the dynamic nature of DNS and DHCP servers operation imposes a big transaction rate (read/write), not suitable for LDAP, which is more read-oriented; that’s why these server uses an Oracle database for information storage.

• IP VPN Provisioning server: The VPN provisioning server distributes MPLS VPN configuration information to routers. The configuration information (created from the Preside Management Console) is obtained from the directory through LDAP. The configuration information is translated into command line interpreter (CLI) data and installed on the router. This application implements the MPLS/BGP model defined by RFC 2547.

• Preside Radius server: The Preside RADIUS server authenticates requests for Internet connections from subscribers. It also authorises access based on the authority level of the subscriber. The RADIUS server does this by matching data from the subscriber’s Profile with entries in the directory.

• QoS Policy server: The QoS Policy Server manages the quality of service (QoS) policies and maintains logical maps for associating specific policies to routers. The QoS Policy server downloads policy information in the form of IP traffic filters (data packet filters) to the network routers within its scope of policy management. The QoS Policy server allows the setup, configuration and management of different QoS levels based on traffic profiles. The QoS Policy server is based on the IP Differentiated Services (DiffServ) model.

  When communicating to Nortel Networks routers, the QoS Policy server uses the IETF-standard Common Open Policy Service (COPS) protocol. The QoS Policy server uses an ASCII command line interface to communicate policy information to other routers, such as Cisco routers.

  By reading the data packet header, an IP traffic filter can direct the data traffic in several ways:

  • drop the data
  • mark the data
  • direct the data to the appropriate pre-defined differentiated service (DiffServ) queue
  • rate limit

  A temporary loss of the QoS Policy server does not cause network traffic loss. The routers that are dependent on an out-of-service QoS Policy server continue to function using the last filter downloaded from the QoS Policy server.

  The QoS Policy Server also supports the application of QoS policies to a VPN site access, through the assignment of policies to the customer edge (CE) or provider edge (PE) interfaces of the applicable routers.

• LDAP Validation Proxy (LVP): The LDAP validation proxy (LVP) is an application layer that validates data provisioning and data access to the directory.

• Integration with external applications: Preside Policy Services allow external applications to integrate in the managed environment by database and directory access through APIs interfaces to these repositories.

2.5.1.2.2.1.2 Orchestream

This company is devoted to the development of Policy Based Network Management applications. The last product release Orchestream 2.1, is composed by the following modules:

• Multi Protocol Label Switching (MPLS) control module for implementing network-based IP-Virtual Private Networks (IP-VPNs)

• QoS control module for managing the Quality of Service (QoS) levels of specific traffic

• Security control module for managing access to specific parts of the network

• Integration Module for integration with other IP network management software. A set of CORBA-based APIs allow the software to integrate with other network management applications that service providers are using in their IP-OSS, including billing, fault management and performance measurement applications.
Modules can be added incrementally to the management infrastructure. For example the MPLS module can be used for VPN connectivity setup, and later the QoS control module can be added to offer VPNs with guaranteed service levels for specific traffic.

**Figure 2.30: System architecture**

MPLS Control Module: Current trends in IP-VPN provisioning are based in the BGP/MPLS approach. MPLS configuration is very complex to achieve manually, requiring specialised network engineering effort to set up a VPN. Orchestream provides high-level definition of VPNs, which requires no complex manual scripting or calculations to configure the devices. Device-level configuration commands are issued automatically by the application. The system supports both fully meshed Intranet, extranet and hub-and-spoke VPNs.

QoS Control Module: Orchestream uses a rules-based interface that enables network manager to set priorities for different traffic types. It follows the IETF's DiffServ framework. Services can be automatically turned on or off at different times of the day, week or month, or event-driven. The software can report on current and future bandwidth reservations on selected points in the network, alerting potential problems where SLAs may be broken.

Integration Module: Orchestream's software core comprises the Service Activation Engine, which contains a model of the network, including a listing of all the devices in the network, their capabilities and connections. It also contains the definitions of the services currently active on the network, and the rules used to build those services. The software has a set of APIs that form the Orchestream Integration Module, which allow information about the services created on the network to be shared with other systems or allow other applications to instruct Orchestream to activate services in the network. For instance, a billing application can direct Orchestream to add a new VPN site, and bill for that new service once activated. On the other side, for example, Orchestream can send fault traps to a fault management package. They can be reported as SNMP traps or accessed via a CORBA interface.
Security Control Module: This module allows rule-based management of access control lists (ACLs) in routers. Rules can be applied global or partially, and can be activated at specified times of the day, week or month, and also on demand.

2.5.1.2.2.1.3 Cisco

Cisco has a vast set of management applications, with some frameworks for OSS integration, namely Cisco Element Management Framework, Cisco Assure Policy and Cisco Networking Services.

Cisco Networking Services (CNS) is a set of Directory Enabled Networking (DEN) software development tools, including an implementation of DEN and DEN-derived Cisco Lightweight Directory Access Protocol (LDAP) schema, LDAP client software, and scalable event services. CNS 1.0 is a controlled release, available to selected network independent software vendor (ISV), systems integrator (SI), and service provider partners, as well as internal Cisco engineering teams. End-user customers will benefit from CNS via directory-enabled products that Cisco and Cisco partners bring to the market.

In practise a Software Development Kit is provided for application development compliant with Cisco’s vision of DEN/CIM models. The intended roadmap for this SDK follows:

Cisco Assure Policy Networking is a long-term project for integration of QoS and Security applications, for example QoS Policy Manager (QPM), Cisco Secure Policy Manager and Cisco Network Registrar. Cisco plans a three-phase approach to the policy package. Actual Phase I comprises unrelated management applications acting directly over the network infrastructure; next phases will integrate the directory concept to these applications with LDAP support and DEN/CIM integration.

2.5.1.2.2 Conclusions

New PBNM applications are being developed to cope with new requirements for IP networks. Modularity and integration is one common feature of these applications, in order to perform the intended QoS functionality without leaving the traditional FCAPS management. MPLS is explicitly supported in many of these applications, and more precisely the BGP/MPLS VPN provisioning model.

The QoS framework of choice is Differentiated Services.

2.5.1.2.3 Trends in Inter domain management systems

As mentioned above, the first level of light integration in multi-technology management systems is performed by encapsulating the WDM management functionality in the SDH management systems. This integration is more a high level one (integration), incorporating 2 management systems in one platform, with common look and feel, mainly with Element Layer and Network Layer functionality. We should say that real management integration in terms of configuration, fault or performance is still under development or not yet performed in commercial products.

Real integration of the different layers is being developed nowadays by manufacturers in co-operation with platform vendors or external parties. Effort is being made on ATM/SDH/WDM integrated management systems or even IP/ATM/SDH/WDM integration, capturing the strong demand for IP oriented services.

The necessity of an Inter Domain NMS is software technologies independent. Nevertheless, the technology adopted by most manufacturers for the integration is the CORBA architecture. In other words, the Inter-Domain management system will come as an umbrella on top of the single domain management systems, providing southbound IDL interfaces towards them. Such efforts have been reported from Lucent, Marconi, Siemens and other manufacturers. Similar efforts have been reported for a 4-layer network, adding the IP one, being in the design phase. This approach is also being adopted by the TeleManagement Forum (TMF), which will be described in the next section.

Significant effort though has been reported in the integration of IP directly on top of WDM networks mainly using the control plane. The efforts are mainly drawn by the IETF and OIF, and manufacturers such as
CISCO, CIENA and NORTEL have already reported lab implementations of protocols integrating the two network layers mainly in terms of path provisioning functionality. Cisco reported the implementation of Wavelength Routing protocol (WaRP) trying to adopt the MPLS functionality in the optical layer (MPlambdaS). Siemens is reporting that integration will happen in the management plane as a first step, and then through the proper interaction between the management and control plane. In other words the management plane will trigger some processes in the control plane, such as end-to-end optical routing. The interaction in the control plane will happen in a few years time, when the need for capacity and MPlambdaS will be really needed.

On the other hand, WINMAN is proceeding in the integration in the management layer dealing not only with configuration management functionality, but also with fault and performance management. WINMAN will keep track of the efforts being reported in the control plane, but focus in the management layer. Interaction and re-use of the work that is in progress in the control plane, is being evaluated within WINMAN. The WINMAN effort will be analysed in a next section.

2.5.1.2.4 The TMF NGOSS approach

With the gap between the worlds of data communications and telecommunications rapidly disappearing, the complexity and size of the networks supporting the emerging information and communications industry is increasing at a corresponding rate. The ability of existing OSS solutions to manage information and communication networks is being rapidly outpaced by the growth of the networks. A new generation of operations support system is needed to manage the new generation of networks.

The Operational and Administration Management system of today has become a roadblock to innovation, instead of a business tool for competitive success. The pace of change in the industry is such that incremental improvements will not work effectively. Service providers must increase their operational efficiency by an order of magnitude to compete, while software developers and systems integrators need to find completely new ways of quickly producing profitable solutions.

This calls for a re-thinking on the part of communications and information service providers on how they manage their business. It also calls for software developers to embrace a new way to developing management software. This is the impetus for the TM Forum’s New Generation Operations Systems and Software (NGOSS™) initiative.

NGOSS™ is based on commercial off-the-shelf information technologies, instead of technologies unique to the telecommunications industry, as many recent management systems have been. This approach significantly reduces costs and improves software reuse and operational flexibility, enabling NGOSS-based systems to support a range of new services and new technology environments more easily. In addition, NGOSS™ is designed to easily integrate with other systems, eliminating stovepipe solutions, or the need to deploy an overlay network for management.
Among the key concepts of the NGOSS™ is the separation of process flow control from functional application, giving service providers the ability to tailor and differentiate their business processes, while still using off-the-shelf software. Equally important are the concepts of common information, a registry, and a common information 'bus' that makes plug and play a practical reality. The NGOSS™ will provide a systems framework that is capable of supporting a collection of interoperable, reusable, off-the-shelf components. These components will implement standardised contract defined interfaces, supported by a registry (containing among other things, the shared data model), and driven by a process workflow engine to enable the service provider to integrate the OSS directly into their business processes.

Today's telecommunications market is characterised by the competition among traditional incumbent operators and new entrant-competitive operators/service providers struggling to provide new, flexible, consistent, fully-automated services adapting to the customers demands. In this context the TMF has launched a significant activity to capture the needs of network operators and service providers and thus enable the “end-to-end process automation of telecommunications and data services operations processes” [TOM2.1]. The Telecom Operations Map (TOM) is the framework for accomplishing the above mission. The TOM defines the business processes and their interactions used by Service Providers in the Customer, Service and Network Management areas. The specific building blocks where the TOM model has focused are the following:

- An industry driven business process model characterising a service provider
- Descriptions of the above processes
- High-level information to perform each process (capturing and satisfying requirements, developing information models, etc.)
- A framework for highlighting the priorities in which processes need to be integrated/automated especially by the industry

The above process as will be shown in chapter 4 has been adopted also by WINMAN after analysing the processes that need to be covered by WINMAN followed by the proper adaptation and customisation.
Finally, TMF is nowadays adopting a transport technology independent common management interface from the Network Management Layer towards the Service Management Layer, called the “Open CORBA” interface. The transport technologies under consideration are the connection-oriented ATM, SDH, WDM with the focus of attention given to WDM, being soon the new dominant in transport networks, while main consideration is also given in the IP technology either connection oriented (MPLS Paradigm) or not. In any case the necessity of an Inter Domain NMS is pointed out by most manufacturers serving as an umbrella on top of the single domain management systems, providing southbound IDL interfaces towards them.

2.5.1.2.5 The TMF CASMIM project

An on-going TMF activity in accordance with the NGOSS is a project called CaSMIT- Connection and Service Management Implementation Team operated by a Catalyst project group. CASMIT is working on the CaSMMI -Connection and Service Management Information Modelling and I/Fs which is relevant to WINMAN northbound I/F and/or southbound I/F. TMF CASMIM i/f will not be provided earlier than the next 10 months. CaSMIT will demonstrate in Nice, France 7-10 May, 2001, automated end-to-end ordering/provisioning/activation of IP based services, and end-to-end service management/trouble ticketing in event of a failure/outage, over a live ADSL/ATM infrastructure. In so doing, CaSMIT will utilise the CORBA-based Information Model being developed by CaSMIM and show that its plug-and-play approach is an appropriate basis for the rapid development of the interfaces needed to integrate tomorrow's operations support systems (OSSs). The telecommunication world of today is especially dealing with all types of services. This trend led to a business approach of a quick & easy service creation, seeing the impact of network fault on services and having a visibility of the problems on several layers. These goals led to the creation of the CASMIM Approach, when the CASMIM acronym means: Connection and Service Management Information Model.

The CASMIM goal is to create a standard model, in order to automate end-to-end service provisioning & monitoring across multi-technology, multi-vendor, multi-layered networks. This involves translation of service requests at SMS into network connection provisioning actions. Another cause is to be “service-oriented” but “technology independent” and in this way be flexible to cater for connectivity set-up across networks using current and future technologies.

CASMIM sees its usage in the following scenarios:

- **Service provisioning** - Automatic provisioning of service whatever the underlying technology implied.
Service assurance - CaSMIM model can be used to export the visibility of a given network layer and to show the impact on service of a given network failure

2.5.1.2.6 The WINMAN solution

The WINMAN management systems will be designed by applying the TMN and ODP principles. The TMN architecture structures the management complex by layering the management applications, defining a common data model, enabling re-use of management data, and specifying system interfaces. ODP goes one step further, it enables design of management applications that are independent of distribution, the underlying infrastructure and management protocols.

In principle one can see Network Management solutions for all types of domains, i.e., the TMN model is applied separately for each transport domain. But, such an infrastructure certainly does not support the idea of integrated management required for end-to-end IP QoS support. The solution is the Inter-Domain Network Management System (INMS) concept, which introduces a network management layer that allows for coordination of management tasks and correlation of management information flows. Such a system will be very complex, huge, and thus may have poor performance with respect to responses and tasks executions. Although it seems highly desirable to have a single Inter-Domain Manager handling all of these functions, in reality it would be impossible to implement it that way, both for technical as well as economical reasons – at least in the case of a large network. This may lead to the decision that, for large networks, at least three INMS systems are required, the Configuration Manager, the Fault Manager, and the Performance Manager.

The initial target architecture consists of the following primary applications:

- Inter-Domain Configuration Manager
- Inter-Domain Fault Manager
- Inter-Domain Performance Manager

These applications and their separable modules interface with domain management systems and service management layer applications via publishable CORBA APIs IIOP specifications. To position the WINMAN solution Figure 2.33 is given were the Inter-Domain Network Management layer is in between the Service Management layer and the Network Management Layers of the different transport domains.

All components of the inter-domain target architecture are required to publish a standard CORBA API to eliminate pair-wise interfaces. When the standard API cannot be met, a Technology Object Adapter (TOA) is developed so that this interface can be met without changing the core applications. This isolates pair-wise interfaces to an adaptation process that translates requests and responses from the external interface protocol into the published CORBA API. This mechanism can also be used to interface to legacy and non-compliant systems.

At the center of the architecture is the information database of all the physical and logical network configuration data needed to manage the network. This database can be viewed as a large directory with open, standard interfaces. The integration approach, which is basically the one adopted by the WINMAN solution, can be summarised in the following figure.
2.5.2. Financial Analysis

2.5.2.1 Introduction

One of the most important business processes network providers must face is network and service management. Services are supported by networks that usually are heterogeneous, consisting of different network elements from different manufacturers and employing different technologies. But customer's perception of services must be homogeneous and not depend neither on geographical nor on technical conditionings.

Network and service providers existed in a stable environment for a long time, but in recent years the telecommunications market has expanded and diversified dramatically. New network and service providers have entered the market and competition has been strongly increased.

Two main factors are affecting the telecommunications market:

- **Liberalisation** implies the evolution towards a competitive market with shorter services development cycles where companies need to keep an outstanding position.

- **New technologies** that allow the development of new services which are the basis for differentiation from other companies.

This competitive environment allows customers to choose among different companies and, because of the apparent lack of differentiation, the degree of customers fidelity has significantly decreased.

Although quick new services development provides a way of differentiation, it is only a temporary advantage as other competitors immediately imitate any strategy. Therefore, *front-office* activities, that have traditionally produced the majority of the perceived value for customers, though still very important in a competitive environment, are loosing weight in favour of *back-office* activities.
It is in this context that network and service management, traditionally considered as back-office activities, allows the development of skills that cannot be easily imitated and thus gives companies a way to differentiate from other competitors. Having an adequate management infrastructure is of crucial importance for network and service providers in order to keep a predominant position in such a competitive environment.

This management infrastructure must cover all the functional areas (fault, configuration, performance, security and accounting) and address all levels of management (network-element, network, service and business levels). Besides, networks are becoming more complicated, shifting from PSTNs to hybrid networks that must inter-work properly, so the management infrastructure must also cover all the technologies in use, providing a common view to network and service providers.

Some of the most important issues in which management systems can be useful for network and service providers to reach and maintain an advantageous position in the telecommunication market are:

2.5.2.2 Revenue increase

One of the most important problems network and service providers must face to maximise profit is revenue increase. Management systems can help to achieve this goal in several ways:

- **Maximising the usage of existing products.** To do so, network and service providers can increase the average usage of existing customers and/or acquire new customers. In order to make customers perceive a differentiation in the services offered, network operators must offer higher quality, as a business which provides perceived better quality than its competitors will get a bigger market share. Managing service quality is therefore an essential activity for all network and service providers. Because of that, one of the main objectives in WINMAN is to provide the management of end-to-end QoS services.

- **Avoiding congestion and temporary networks crashes.** Network unavailability results in revenue loss, as it impedes service operation. It also has a deferred effect by damaging a company's image. Moreover, network operation and maintenance will be influenced by the growing layering of network technologies (e.g., IP over ATM over SDH over WDM or IP directly over WDM). Such layering increases complexity and time needed to identify problems because each layer can impact a higher layer in elaborate ways. Therefore, network providers need management systems that are able to effectively operate hybrid networks, providing facilities to quickly identify and locate problems.

- **Charging higher prices.** Service providers can charge higher rates for services that require a high degree of quality. For this kind of services higher prices are acceptable but they must comply with Service Level Agreements (SLAs). SLAs are used to record commitments on the quality of service offered. As a consequence, service providers need tools to manage and measure network quality and performance so they can provide some evidence of the quality of service offered.

- **Introducing new services.** Basic services (voice traffic, bandwidth, etc.) tend to become undifferentiated standard products (commodities). Service providers need to introduce new services that yield greater added value and profit margins, as customers are willing to pay more for them. These complex services are supported by multi-service networks that need an adequate management. Furthermore, service development and life cycles are continuously shortening. It is of utmost importance for service providers to have the ability to swiftly develop and deploy new services to respond to the ever-changing environment and demand.
2.5.2.3 Cost containment

Traditionally, in a monopolistic environment and due to tariff regulation, it was enough not to increase costs by a greater proportion than revenue to keep up the expected profit. Therefore, cost management was not a critical issue for network and service providers.

Nevertheless, this situation has dramatically changed with the advent of competence, as cost management has turned into a key factor to keep up margins and offer competitive prices. Consequently, network management has become very important to efficiently operate the existing infrastructure and contribute to cost containment.

Network infrastructures employing equipment from different manufacturers permit network providers to negotiate better prices but at the same time result in greater network maintenance complexity. On the other hand, network complexity is deepened by the diversity of technologies used, producing hybrid multi-layered networks.

In this context, management systems have enforced operational costs reduction, contributing to the following changes:

- **Integration.** Network management must integrate in a unique system the operation of networks provided by different manufacturers and based on different co-operating technologies. In this way network management systems offer a unique management interface that reduces operation costs by integrating in the same system the information and management facilities that otherwise would be dispersed, making management tasks easier and faster and allowing to have a unified work force in which all operations staff can handle all types of network elements. At the same time, integration makes operational staff training easier and faster. Integration also permits a unique interface to service level, reducing development costs and complexity in service level management systems.

- **Distributed environment** Network management should allow remote operation on networks and network elements in a fast and efficient way, thus minimising the impact of network failures on services. In addition, a distributed environment allows for a suitably organisation and location of the operational staff where it is more cost effective to do so. To meet these requirements, network management systems must provide remote and distributed operation.

- **Network evolution.** Network management systems must deal with networks that are increasingly complex. These networks are continuously changing, incorporating new equipment and technology domains. Therefore, network management systems must adapt easily to changes in the network and be able to incorporate new network elements.

- **Interoperability.** Another important issue that must be considered is the integration with other management systems. Consequently, network providers need management solutions that are easy to extend and offer standard interfaces to network elements and other management systems.

2.5.2.4 Standardisation

Standardisation has become very important in the telecommunications market, as it provides a common framework for network and service management. Network and service operators' interest in the development of management standards stems from the fact that the adoption of standards by the telecommunications industry makes the integration of network elements and management systems much easier. Thus, the use of standard solutions, as the TMN model, results in lower costs and more efficient management solutions.
Therefore, network management systems should be based on standard solutions. Using standard architectures and interfaces has the following advantages:

- **Standard interfaces to network elements/network element level management systems.** As standards are being accepted by industry, most manufacturers include standard interfaces in their network equipment and network level management systems. Therefore, network management systems should include standard interfaces to the network element layer so new equipment can be easily added.

- **Interface to network management systems.** WINMAN offers a solution to network management integrating several network level management systems covering different technology domains in a common inter-domain management system. The interfaces with the individual network management systems will be based in standard solutions, thus allowing an easy evolution an integration of new technology domains and the corresponding network management systems.

- **Interface to service level management.** Offering standard interfaces to the service level management allows for a smooth integration of applications at that level. Therefore, services can be effectively created and managed using service level management systems that communicate with the integrated network management system via these interfaces. WINMAN provides interfaces to the service level that can also be used by third parties interested in offering more sophisticated management solutions.

### 2.5.2.5 Constraints from Regulators

In order to ease the introduction to the telecommunications market of new service providers, regulation authorities are defining conditions on incumbent service and network providers. Usually these conditions imply the provision of information on customers and network resources and the permission of electronic access to Operations Support Systems.

Requirements on electronic access to OSSs translates into transactions across an interface between the service and network providers dealing with orders, customer information, trouble administration, network usage data, etc.

As it has been already mentioned, the WINMAN solution considers the possibility of third parties (companies interested in building up their own management solutions on top of WINMAN) access via standard interfaces to the network management functionality provided. In this way WINMAN will help in the fulfilment of constraints imposed by regulators.

### 2.5.2.6 Conclusions

Network and service providers need management solutions, usually based on standard models, to help them in the operation and maintenance of their services and infrastructure. In this sense, WINMAN aims at an integrated solution to the management of IP services deployed over a DWDM network.
3. Business Model and System Requirements

3.1. The WINMAN Methodology

Figure 3.1 depicts the methodology adopted regarding the production of the Functional Architecture (FA) and Functional Interfaces (FIs) of the WINMAN system.

The WINMAN Methodology consists of the following major phases, which may be viewed as steps of a logical sequence towards the specification of the Functional Architecture and Interfaces between components:

- **Business Reference Model**
  The identification of different roles that may be undertaken by certain business entities and the actual business (enterprise & end-user / customer) entities that participate within the WINMAN context are the goals of this initial phase, that effectively triggers the overall procedure.

- **Requirements Identification**
  In this phase all requirements are brought together. The requirements are the rules the WINMAN solution should adhere to. The requirements describe the 'what' of the system. They are categorised, elaborated and prioritised, until an agreeable and feasible set of requirements is achieved.
  The approach adopted by WINMAN is to link the actors of the Business Model with the requirements of the WINMAN solution. In other words the actors of the Business Model pose their requirements as external actors to WINMAN, and then the WINMAN system requirements are derived from the needs of the WINMAN actors.

- **Use Cases Specification**
  Following requirements definition, the next step is to proceed with the definition of appropriate Use Cases. Use Cases describe the interaction between the actors (both human and non-human entities) and the system under
design. A Use Case mainly consists of a number of steps and their variations - effectively scenarios - describing the relevant interactions. This is done in plain (human language) text and complies to a pre-defined template. Within a Use Case there is always a successful scenario description and at least one scenario describing an error condition. Both error and success scenarios have a “most likely” context, thus avoiding description of extreme occurrences. Use cases may describe different parts of the system but they may also describe the system on different levels. See also [UC1]
The sum of all the Use Cases should cover all the requirements. There should be a feedback from the UCs to the WINMAN requirements. UCs are coupled with the appropriate set of requirements at a later stage.

- **UML Message Sequence Charts**
  
  A more detailed examination of a scenario reveals interactions between the actors and the system and within the system itself. These interactions may be specified in a formal manner using Message Sequence Charts (MSCs). A sequence diagram shows an interaction arranged in time sequence. In particular, it shows the objects participating in the interaction by their lifelines, and the messages that they exchange arranged in time sequence. It does not show the associations among the objects. It represents an interaction, which is a set of messages exchanged among objects within a collaboration to effect a desired operation or result. A sequence diagram has two dimensions: the vertical dimension represents time and the horizontal dimension represents different objects. Normally time proceeds down the page (although the dimensions may be reversed if desired). Usually only time sequences are important but in real-time applications, the time axis could be an actual metric. There is no significance to the horizontal ordering of the objects. Within these charts, messages and included parameters can be described in detail. Moreover internal actors (i.e. WINMAN sub-systems) are introduced at this stage.

- **UML Class Diagrams**
  
  When MSCs are defined for all scenarios, then those charts can be converted to class diagrams. A class diagram describes the types of objects in the system and the various types of relationships that exist amongst them. Furthermore, class diagrams show the attributes and operations of a class and the constraints that apply to the way the objects are connected. This is easily done, if the messages in the message sequence charts become class methods, and the vertical lines become the classes. The attributes of each class, can be identified from the messages exchanged (MSC message attributes - components in brackets - become class attributes). The class diagrams are out of the scope of this deliverable and will be studied and documented in a separate one.

### 3.2. WINMAN scope

The architecture of the next generation Internet will take advantage of the provision of integrated IP over WDM services consisting of an IP network layer directly on top of WDM transport layer. In other words, we refer to a network consisting only of IP- and WDM-technologies equipment, while the encapsulation of IP over WDM can be accomplished in different ways with simplified network stacks deploying protocols such as Packet over SONET/SDH, Gigabit Ethernet or Simple Data Link, as described in detail in [3]. The basic guidelines for the integrated IP/WDM architecture is that WDM is considered as a core/backbone technology and IP is interconnected to the WDM equipment at the edges of the core network. Such a network is mainly considered by Internet Service Providers and in particular, Competitive Operators, deploying optical infrastructure, leased or of their own, willing to provide IP services on top of it using IP Points of Presence (PoPs). Optical infrastructure, gradually evolving from ATM/SDH, consists of different topologies of WDM equipment in the metropolitan or backbone areas. Incumbent operators could also deploy such a network, where in that case they integrate their existing ATM and SDH infrastructure with the WDM equipment by using the WDM backbone or core to carry the ATM and SDH traffic. WINMAN interest focuses on managing the IP over WDM part of the network without implementing any functionality for ATM/SDH networks.

Three main areas are considered in this integrated IP over WDM network architecture:

- **Access area**, where main Business/Enterprise customers or smaller Residential/Small Office/ Home Office IP customers are interconnected to the ISP acquiring Internet access.
- **Metro area**, consisting of an optical WDM metro core with ring topologies dominating, and metro access area, where the IP PoPs are located. IP PoPs can be of 2 categories. The edge level ones are the gateways to the Customer Premises IP equipment and the core or transit ones are used to groom traffic and forward it to the IP backbone.
• Backbone area, consisting of core level IP PoPs, which are interconnected via the WDM backbone network. WDM backbone network topologies heavily depend on the distances of the IP PoPs. For long distances with significant power losses (partial) mesh networks or concatenated rings of point-to-point WDM systems are most common, while for smaller distances similar topologies with the metro area (e.g. rings) are applicable.

The main idea behind an ISP’s network is that IP PoPs are interconnected with the optical WDM backbone. IP PoPs can be located either in a metro area, or in smaller PoPs for smaller cities, where a metropolitan network is not present. In a metropolitan area usually the concatenation of IP PoPs forms a distributed Giga-PoP with multiple edge routers interfacing customers, called Provider Edge (PE) routers following the BGP/MPLS VPN terminology [4], while in smaller PoPs there is only a limited number of PE routers. The IP PoPs, either GigaPoPs or smaller PoPs, use the core high-end routers, called, in accordance to PE routers, Provider (P) routers, to interconnect to each other. Each PoP is usually equipped with only one P router, but more than one can be accommodated if this is required. The above are further elaborated in the following sub-chapters with the appropriate figures, covering both the Metropolitan Area and the Wide Area Network or Backbone network.

Figure 3.2 depicts a potential ISP’s metropolitan network consisting of a WDM optical metro core and IP metro access. The IP section is composed by a number of IP points of presence (PoPs), where customers can access the IP network services and traffic is groomed and forwarded to other PoPs or networks through the backbone. Access is facilitated to customers through the interconnection of the ISP’s Provider Edge (PE) IP routers with the Customer Edge (CE) IP routers. The ATM and SDH equipment are shown for completeness purposes and are out of the WINMAN scope. Provider equipment can be collocated or not with the customer equipment, depending on the distance between customer and provider premises and on the amount of traffic generated by the customer and the tele-housing policies.

Figure 3.2: Metropolitan Area IP over WDM Example

The optical WDM Metro core is usually composed by a ring of re-configurable Optical Add-Drop Multiplexers (OADMs), while additional point-to-point WDM links with Terminal Multiplexers (TMs) can be considered for big customers. OADMs offer management interfaces so that they can be remotely re-configured to add and drop wavelengths (optical channels) to the ring through the tributary cards and multiplex them in the form of optical line signal in the corresponding line cards of the ring in each direction of the latter (east and west). 3R Regeneration (Regeneration, Reshaping and Retiming) inside the ring depends on the ring physical length and
the number of elements inside the ring and can be definitely decided upon, with the proper measurements of the so-called power budget. In any case, most WDM manufacturers cannot exceed the number of ten elements inside a single ring. 3R cannot be accomplished yet without Optical to Electrical conversion and vice-versa. Such O-E conversions will be used by manufacturers to perform the so-called digital wrapper functionality currently being standardised, adding some client independent overheads similar to the SDH ones, to support Operations, Administration and Maintenance (OAM) functionality in the Optical Layer. Although such a notion could be considered to be against the all-optical vision, it should be noted that the digital wrapper could be configured to be set/not set upon demand. It could be used to troubleshoot performance degradation or faults introduced in the optical layer.

On the other hand, the IP metro access is composed by a set of PE routers interconnected via optical interfaces with the OADMs. The PE routers usually support one or multiple optical interfaces, which interconnect to the corresponding tributary card of the OADM. In case the router provides coloured interfaces (working in 15xx nm) for transmission and reception, there is no need for a transponder in the OADM. The usual case though is that the routers’ optical interfaces work in 1310 nm and there is a need to adapt this wavelength to the 15xx, which is done by the corresponding two-way transponder. The transponder converts the optical signal of 1310 nm to electrical and back to coloured optical. In addition, the transponders are capable of providing the 3R regeneration, as well as to support wavelength conversion through the O-E-O conversion, since this is not easily managed in the all-optical domain with optical means.

Note that in some cases a single PE router cannot handle in terms of interfaces or capacity the traffic inserted by the customers and a bundle of IP routers are used to serve such a situation. These IP routers are usually interconnected in a high-speed LAN (e.g. via Fast or Gigabit Ethernet interfaces), as shown in the ellipsis on the bottom of Figure 3.2. Each metropolitan area network uses a high-end Provider (P) Router, which is responsible of grooming all the traffic from PE routers of the metro area and forwarding it to the backbone or another P router of a foreign domain (Network Access Point for Internet Connectivity or peering with another ISP).

In case there are two WDM rings, then an optical cross-connect is needed, to route wavelengths from one ring to the other supporting all-optical networking. Such cross-connects are the “kings” and the most expensive of optical networking equipment capable of performing additional tasks, such as wavelength switching and conversion for hundreds of ports in an all-optical form without O-E conversion.

Figure 3.3: Wide Area IP over WDM Network Example
The Wide Area network is usually composed of a partial mesh-type optical WDM network. In case the distances do not allow the sole use of re-configurable OADMs -since the power budget is not sufficient for bigger distances than 1000km without 3R regeneration-, then point-to-point long-haul Optical Terminal Multiplexers (OTM) are being used as shown in Figure 3.3, in co-operation with the proper line 1R Optical Amplifiers and electrical 3R regenerators. Optical Amplification is deployed either to boost the aggregate multiplexed optical line signal (e.g. with Erbium Doped Fibre Amplifier) or to separately regenerate each optical channel at the corresponding tributary. 3R regenerators interconnected back-to-back between two WDM elements regenerate/reshape/retime the client signal after converting each optical channel to electrical and vice-versa. This is usually accomplished using Transponders as explained above. OTMs can co-operate with the proper fixed or re-configurable OADMs to add/drop wavelengths in the desired spots between the point-to-point OTMs supporting end-to-end optical paths, which is usually called the light-paths.

P routers are usually needed in all the geographical areas, where access IP traffic through the corresponding PE and CE routers is inserted. Note that in case optical TM's are used, then the P routers are placed between two co-located WDM TMs, which can be also interconnected back-to-back with optical fibres (patches) or transponders, as shown in Figure 3.3.

The WINMAN management system intends to manage all the IP Provider Edge (PE) routers and the IP Provider (P) router of each metropolitan area as well as the IP collocated routers interconnected with pure IP interfaces (e.g. Ethernet). All the WDM equipment used to interconnect the IP PoPs, as well as the WDM backbone linking several metropolitan areas to constitute a wide area network (WAN) are inside the WINMAN scope. Customer edge routers are considered out of the scope of WINMAN scope.

### 3.2.1. The Control Plane vs Management Plane approaches in the IP/WDM Integration

#### 3.2.1.1 State of the art

The term control plane is used in the literature to refer to the set of real-time mechanisms and algorithms needed for call or connection control. It deals mainly with the signalling to set-up, supervise and release calls and connections [2]. Although, a detailed de-composition of the control plane and a description of each component is not the purpose of this deliverable, we can safely assume that the signalling protocol for connection set-up and the routing protocols supporting network discovery are the most significant features of the control plane. In that respect, it is significantly easier to follow all the recent advances and proposals about the integration of the IP-electrical world and the WDM-Optical world control planes.

Many standardisation bodies as well as international fora have addressed the issue of integrating the control plane of the IP MPLS-capable and the WDM network elements (NE). The IETF has proposed the MPLambdaS framework [5], which extends the MPLS ideas to the optical domain, allowing the re-usability of the existing Internet protocols with the appropriate extensions. The OSPF as well as the IS-IS routing protocols have been enhanced to disseminate information relevant to the optical domain ([6], [7]). On the other hand, the OIF and the ODISI forums have made one step towards the definition of the appropriate signalling messages ([8], [9]), which will allow the dynamic set-up of end-to-end connections between IP routers spanning the optical network. Furthermore, [10] and [11] present the mapping between the signalling messages defined in [8] and existing IP/MPLS signalling protocols, namely RSVP-TE and CR-LDP.

The latter is an important issue, since the automatic provisioning of end-to-end connection between two IP routers by means of signalling permits to Network Operators (NO) and Network Service Providers (NSPs) to provide rapidly the requested IP connectivity to customers. This is the main requirement of the NO and NSP in order to cope with the changes in the service-chain and to be able to provide in a cost efficient way the requested services.

Different business models [12] have been identified for the IP/WDM integration, each one imposing different requirements, mainly in the areas of security and dissemination of topology information. In [13] and [14] three inter-working scenarios are described:

- **The Overlay model**, where the routing algorithm, topology distribution and connection set-up signalling protocols of the IP and the WDM networks are independent.
- **The Peer model**, where the IP network has full topological view of the optical network and just a single routing algorithm instance is running in both the IP and the WDM network.
The Augmented model, being a combination of the previous two models. Each layer has its own protocols, however routing information exchange is allowed between the two layers. The overlay model is the one that allows an easy migration from the existing situation to the deployment of optical network elements (ONEs) for the transport of the IP directly over WDM. However, the implementation complexity of this model is a burden and it does not promote the integration of the control plane of the IP and the WDM networks. On the other hand, the peer model, which promotes the integration of the control plane of the IP and the WDM networks, is simpler in implementation, but its operation is far more complex than the overlay. Finally, the augmented model is the golden mean, combing the advantages of the peer and overlay model and minimising their disadvantages at the same time.

In our opinion, such an automatic provisioning of end-to-end IP services cannot yet be provided by signalling means in a real network environment, since all the above-mentioned proposals are still immature. The interoperability of ONE belonging to different manufactures has not been solved and is not foreseen in the instant future. Additionally, the different approaches or proposals are currently implemented and evaluated in field trials and have not been deployed in real network environments, which are much more complex than the lab environments.

On the other hand, the term management plane is used in the literature to refer to the set of near real-time field management mechanisms and algorithms related to the system as a whole and to the OAM [2]. It deals mainly with the procedures related to five functional areas, namely Configuration, Fault, Accounting, Performance and Security (FCAPS). The three main management functions of the FCAPS which have similar functionality with the control plane functions, thus competing each other, are path provisioning with routing and QoS support, in the Configuration Management Area, and automatic recovery of failures or performance degradations in the Fault and Performance Management Areas. Network Management functionality mainly exists independently for the IP-electrical world and the WDM-Optical world rather than for the integration of the two worlds. In this direction we will first deal with the advances and proposals for IP layer and WDM layer separately and then we will provide the limited efforts on their integration.

The appearance of the MPLS protocol makes the IP network resemble a connection-oriented network. This paves the way to extend network and service management to the IP layer. On the other hand, the paradigm of Policy Based Network Management enters also into the arena with the Common Open Policy Service (COPS) ([15], [16]). The motivation is to be able to automate the management of the network devices and hence to allow scalable network management solutions. Very recently, the IETF has already provided policies and mapping mechanisms for networks based on MPLS. The opportunities of MPLS and PBNM will be further investigated in our research work.

On the other hand, Management is dominant in the Optical Layer. WDM equipment manufacturers have adopted the SDH-style Management Systems although standardisation efforts are not yet paving very fast. They provide TMN-layered sophisticated management systems in the Element Management Layer (EML) -Element Managers- and Network Management Layer (NML) -Network Managers. Service Management Layer (SML) functionality, although still limited in commercial products, is complementary to the Network Layer providing end-to-end path provisioning services (light-paths, optical VPNs) and limited fault or performance management to identify service impact for client signals, mainly ATM and SDH.

In addition some lightweight integration is performed by including the WDM management functionality inside the SDH management systems. Real multi-technology management integration in terms of configuration, fault or performance is still under development in commercial product releases. Efforts have been reported on SDH/WDM, which will be available soon, providing integrated end-to-end SDH paths over WDM light-paths. As a next step ATM/SDH/WDM or even IP/ATM/SDH/WDM integrated management systems will appear, capturing the strong demand for IP oriented services.

TMF has launched a series of programs along with the supporting catalyst projects to capture the needs of network operators and service providers to enable the “technology integration” and the end-to-end “process automation” of telecommunications and data services operations [17]. The Telecom Operations Map (TOM) is one of the main frameworks for accomplishing the above mission. The TOM defines the business processes and their interactions used by Service Providers in the Customer, Service and Network Management areas. In this context, TMF has adopted and started specifying a transport-technology-independent common management interface from the EML towards the NML and from the NML towards the SML, sometimes called the “Open CORBA” interface. The transport technologies under consideration are ATM, SDH and WDM domains with the focus of attention given to WDM, being soon the new dominant in transport networks. As a next step, the technology neutral management interface will be adapted to an IP connection-oriented model characterised by MPLS technology. In this view, the necessity of an Integrated or Inter-technology-domain NMS (INMS) is pointed out by most manufacturers serving as an umbrella on top of the single technology domain management.
systems, providing southbound IDL interfaces towards the technology-domains and northbound interfaces to the Service Management Systems (SMS).

3.2.1.2 WINMAN management approach

The integration of the IP and the WDM technologies is an issue of intense research. However, the majority of these proposals are based on the concept that such integration should be done through the control plane. On the other hand, little attention has been given to the integration of the IP and the WDM technology using the management plane. A hybrid solution has been proposed in [18] as the outcome of analysing the technical maturity and customers needs of the two approaches. The solution proposed a very limited use of the immature control plane, mainly supporting automatic error detection and protection, and an extensive network management plane, supporting automatic discovery, set-up of connections and a series of other FCAPS management functions. The WINMAN solution is more or less in this direction focusing on the management plane. The use of the control plane will not be possible until standard interoperable routing and connection set-up signalling protocols appear and extensive tests provide performance assurance. Automatic protection switching in the optical or IP layer provided by the elements is the most important feature of the control plane that management functions such as restoration cannot compete. In addition, the digital wrapper type of OAM functionality should be put into practice when standardisation reaches maturity levels. In this view, there could be a period of time where management plane could dominate integrated IP over WDM service provisioning taking advantage of the lack of standards and performance proof by the control plane. The transition period should allow the stable features of control plane to interact with the management plane and finally signalling mechanisms will gradually dominate the IP over optical area, provided that the QoS-enabled Internet vision has shown the anticipated results. The above ideas are captured by WINMAN research project, specifying and developing a Network Layer Management System providing southbound interfaces towards the IP and WDM Network Management Systems and northbound towards the Service Management Systems. The architecture is technology neutral and can cope with other transport technologies like ATM or SDH, but these will not be implemented, since our focus of attention lies on the direct integration of IP over WDM networks.

3.2.2. Control & Management Plane Interaction

The WINMAN solution could take advantage of the mature features of the control plane, such as automatic protection and discovery of nodes. In this view, there could be a transition period where the control and management plane will interact with each other, to provide the best of the two worlds.

3.2.2.1 Discovering the network topology

3.2.2.1.1 IP Network elements

Usually IP network elements run an IGP protocol (OSPF, IS-IS) that allows them to discover the network topology. This feature is very common and crucial to our system. All network elements maintain databases with all the information about the topology of the network. The content of the network databases is periodically refreshed (by means of hello messages) in case that some network element or link went down without sending any notification. Information about new network elements and links is spread across the network by the routing protocol. In a certain amount of time (convergence time of the routing protocol) all the elements in the network have their databases actualised.

A failure on a network element or on a link will immediately trigger update messages from the routing protocol. Nevertheless these messages must reach all the network elements so they all have the information they need to update their network databases. This process takes its time. With modern IGPs, and for traditional data networks (best-effort networks) the process is usually very fast but for real time services the delay may be
acceptable. And there is the possibility that some network element fail without being detected. This way the change of the network would only be detected when the last hello messages sent by the failed element aged out.

The mechanisms used to discover the network topology, dissemination of information and fault management supported by the network elements are very well developed for data networks. They are the result of decades of experience and lots of people’s hard work. But new challenges to these protocols are appearing. The integration of different types of traffic in these networks requires new features that are completely new to them. A lot of work is taking place at standardisation groups (specially in IETF) to give these protocols the means to cope with the actual challenges, but it seems that maybe we will have to wait some time until we have the desired performance.

### 3.2.2.1.2 Management system

The management system can listen to the messages sent by the IGPs. It can have a passive participation on the IGP algorithm getting from it all the information that the IP network elements have. This means that this way the management system can easily be as well informed about the network topology as any other network element.

Along with the routing protocols the management system can have other sources of information. The management system can query any network element to learn about its status.

Usually all the IP network elements send annunciation messages when they are turned on. The management system can use these messages to learn about new network elements and then can query them to learn more about their status links and adjacencies.

When something wrong happens in a network the affected network elements send alarms to the management system. The management system can use this alarms to locate the fault and then query the affected network elements to actualise its network databases. This way, and contrary to what happens with the network elements, the management system doesn’t need to wait for the IGP messages.

In principle the management system has better means of having an actualised network map. It is in better position of making decisions that are based on the network topology and resources.

### 3.2.2.2 Creation of IP paths

In order to transport real time traffic and to limit jitter is necessary to reserve resources on the network and to make a traffic flow follow always the same route. For this reason we create IP paths on the network. An IP path is similar to a virtual circuit. Each path has its own resources reserved and packets send through a path follow always the same route.

Therefore to create an IP path a system must first choose a route. This route may not be the shortest one has it happens in traditional IP routes. The route is chosen having in consideration constraints like bandwidth and QoS parameters. The process that does this job must know the network topology to be able to make a good choice. We called this the design phase.

After the path is designed the routers that support them must be configured and the necessary network resources must be reserved. We called this the implementation phase.

### 3.2.2.2.1 The Control Plane approach

Huge efforts are being made to standardise protocols to perform this approach. One of the pioneers standard fora in this subject is the IETF.

The idea in this approach is that all the work is done by the control plane (network elements running routing and signalling protocols).

The ingress router makes the design of the IP path. To do this job it needs to know the topology of the network and the free resources on the links it needs to use. The knowledge of the topology of the network is granted by the routing protocol. This is a common feature on actual networks, it seems that it will not be a problem.
The big problem in this phase is how to get the information necessary to perform the traffic engineering. Extensions to traditional routing protocols (like OSPF) are being developed to enable them to transport this kind of information [25]; maybe this problem will be solved soon.

But there is another aspect that should be mentioned. In this approach all the edge routers can create IP paths reserving and freeing resources on links all over the network. It is likely that the status of the links are continuously changing at a very high rate, at least when comparing with frequency of link and node failures or the appearance of new elements (that was for what the routing protocol was made for). Having in consideration the time it takes for a change in the network to become the knowledge of all the network elements, it is highly probable that the edge routers never have actualised information about the network resources. They will probably fail some calculations.

Ways to attenuate this problem were studied in [23].

On large networks this calculations can be a high CPU consuming process. The edge router that has to perform the calculation still has to perform its basic function: forward traffic. To put all this functions on the ingress router can harm its performance.

The implementation is made by a signalling protocol. The edge router that performed the first phase triggers the process. The packets sent by the signalling protocol follow the route of the IP path making the resources reservation and the path establishment. The great advantage of this process is the speed. The paths are established very fast.

To perform this task the signalling protocol must support constraint based routing and explicit routing. The problem with this process is the compatibility between different vendors. There aren’t standardised protocols to perform this task. Many people are working on this subject but the problem seems far to be solved. There are two protocols that are candidates to this task: CR-LDP and RSVP-TE. Right now there are vendors that support one and others that support the other. After everybody reaches an agreement about which one to use there still is all the work of standardisation and to make all the implementations compatible.

If we consider multi-vendor and vendor-independent systems, by now this approach is almost impossible to implement.

3.2.2.2 The Management approach

This approach is the opposite of the Control Plane approach. In this approach the Control Plane has no responsibility. There is a management system that can do all the work.

If we pretend to implement a multi-vendor or vendor-independent network this is, considering the actual state of the standardisation process in this area, the easiest way to reach our goals.

The management system makes the design of the IP path. To do this job it needs to know the topology of the network and the free resources on the links it needs to use.

The management system has the means to access the same information that the network elements do. Therefore it can have at least a view of the network as good as any network element. In fact, if we consider that the management system can query the network elements, receive alarms and other messages sent by network elements it is easy to conclude that the management system has much more chances of getting a best picture of the network.

If it is the management system that does the IP path design then it can keep track of all free and reserved resources on links. This way the information of free and reserved resources on the network is always up to date with no need for any additional effort.

Another immediate advantage of this approach is that we are freeing routers (edge routers) from the high CPU consuming task of designing paths. The routers will also no longer need to have a very accurate view of the network. They don’t need to know nothing about reserved resources, the knowledge about network topology can be the one given by traditional IGPs. This way routers can concentrate all their capacity on their primary function: forwarding traffic.

The job of designing IP paths still is high CPU consuming but no longer interferes with traffic forwarding and can be performed by powerful servers optimised to this task, not by routers that have other concerns.

On this approach the implementation of the IP path is also performed by the management system. The management system creates the label mappings and makes the reservation of resources in every router of the IP
path. To do this job the management system needs to have access to a MIB that allows all the necessary configuration. MIBs for MPLS are proposed in [21] and [22]. The great advantage of this approach is that we completely avoid all problems with compatibility between the signalling protocols implemented on the network elements. The great drawback is the speed. The management system must access each router that participates in the IP path and configure the label mappings and the reservation of resources. This process is much slower than the control plane.

3.2.2.2.3 The Control & Management approach (best of two worlds)

It seems clear that comparing the two previous approaches we see that one of them is clearly better in the first phase and the other one is clearly better in the second phase. What we try to do in this third approach is to define a mixed solution that combines the best of the previous approaches.

The design phase is performed by the management system. The management system has a clear picture of the network and it has access to information about network topology changes faster than any other network element. Because it is the management system that makes the design of the IP paths, it is the management system that decides what resources should be reserved and when. So the management system knows where and what resources are reserved on the network. The information about resources is always up to date on the management system. By making the design in the management system we maintain the advantages of freeing the edge routers from this high CPU consuming process and having the best-informed entity making the design of the IP paths.

The implementation phase is performed by the control plane, but triggered by the management plane. After the management system has designed the path it triggers the signalling protocol. The signalling protocol must support explicit routing and must be able of performing resource reservation. The information transmitted by the signalling protocols is given by the management system when it triggers the signalling. This way we have paths designed by the management system implemented as fast as the ones designed by the control plane (and implemented much faster than the ones implemented by the management system). The major problem in this approach is the same that we had in the implementation phase of the control plane approach. All the network elements must use compatible signalling protocol implementation. This is very hard to provide in multi-vendor networks nowadays.

We think that with this approach we are using the best of the previous two. We can have IP paths established by the best-informed entity (management system) and implemented by the fastest mechanism (signalling protocols of the control plane).

3.2.2.3 Network Restoration

3.2.2.3.1 Switching to protection paths

When designing the IP paths we can design in parallel with the main IP path a protection path. That protection path can be shared with other IP paths or can be reserved just for one path. The objective is to have an alternative path to forward traffic in case the main IP path fails. Since the protection path is already designed and reserved to switch traffic to the protection path is very fast. This is a fast and effective way of making network restoration. When the switching to the protection path is needed the protection path is already created, so the way we create that path (management or control plane) is not relevant.

Another important thing about switching to protection paths is how the ingress edge router knows when to switch to the protection path. If the control plane created the path, when a link or network element fails and that fail breaks one IP path a message must be sent to the ingress edge router. The router that noticed the fail sends the message using the signalling protocol. The edge router then switches to the protection path.
In parallel with this process other thing happen. The network element that detected the fail sends an alarm to the management system. The management system automatically checks what services where affected by the fail. The management system can then send commands to the ingress edge router of each affected service to switch to the protection path.

The action that reaches the ingress edge router first makes the switching.

If the management system created the IP path we only have the last mechanism. If it is fast enough we not notice any difference.

### 3.2.2.3.2 Redesigning IP paths

Another step to take when a network fail occurs is to redesign the affected IP paths. The breakdown of a link can affect a large number of established IP paths, so it can trigger the redesigning of a large number of IP paths all at the same time. This can emphasise the drawbacks of each approach.

If the control plane approach was used each edge router has to redesign all the affected paths that it designed. If this process is high CPU consuming for the design of one IP path it is even worst for more than one IP path. This can seriously compromise the performance even of the services that were not affected by the fail. On the other hand at failure event there should be lots of edge routers redesigning IP paths at the same time. This means planning to reserve resources on the network. Probably there will be edge routers reserving resources that were already been reserved by others edge routers. Even if it doesn’t happen at the same time it will be very difficult to broadcast information about reserved resources at a high enough speed in a situation like this.

In the implementation phase the paths are established by a signalling protocol. The fastest way there is to do that. A large amount of affected IP paths can just increase the traffic generated by signalling protocols, which we think is not that significant. This process is studied in [20]

If the management plane was used all the work must be done by the management system. The positive side of this approach is that we take out all the pressure from the edge routers, saving their performance on non-affected IP paths. The management system is just one entity so we should not have problems regarding resources that were reserved twice. The time it takes to the management system to recalculate all the IP paths is just a question of computation power of the machine that supports it and software efficiency.

The big drawback on this approach is the time the implementation phase may take. If we say that the implementation of one IP path is slow on this approach, the implementation of several IP paths should much worst. And we are talking about time critical operations.

Again, the control & management approach seems the best one. We use the management system calculations, that look more stable, and we try to use the control plane speed to implement the paths.

### 3.2.2.4 Conclusions

The management system has better possibilities than the control plane to have a correct and permanently actualised view of the network. This fact determines that the management system is the entity that can make the better decisions on performing traffic engineering. Besides, making the design of the IP paths in the edge routers can compromise their performance specially in failure situations when can be necessary to redesign several paths at the same time.

In the implementation phase we have shown that the control plane is the best choice, essentially because of its speed compared to the management approach. The Control & Management approach looks like a good choice because it can combine the best of the other two approaches. The major problem in the Control & Management and in the Control Plane approaches is the need for a universal signalling protocol. The actual absence of a standardised universal signalling protocol is our major implementation problem. The two candidates for this are RSVP-TE and CR-LDP, but it seems very difficult to have an understanding on what protocol should be used. Each vendor implements the one he thinks it’s better, and compatibility is not assured even between vendors that use the same protocol.

While this situation lasts it will be very difficult to implement the Control Plane approach and the Control & Management approach in a multi-vendor network.
3.3. Business Reference Model

3.3.1. Business Roles

Business entities, which are considered users and producers of services in today’s information market, may play different business roles. The following types of business roles were identified as having some kind of interaction, either directly or not, with the WINMAN system:

- Customers and end-users:
  - Customers are legal persons, humans or companies, which have contracts with VASPs about the right to use telecommunications services and the obligations to pay for this right and the usage of these services according to the tariffs.
  - End-users are entities, which interact with the VASPs to obtain the effect of the service. End users may be humans or an automated piece of application software.
  - This is the only business role consuming services, not trying to make profit out of them.

Examples of these roles include:
- VPN Business Customer is the one who orders a VPN among its different customer premises or endpoints.
- VoIP Business Customer is the one who orders a VoIP service for its premises or endpoints.
- MoIP Business Customer is the one who orders multimedia services for its premises or endpoints.
- VPN Business End-user is the one who makes use of the VPN among the different customer premises or endpoints.
- VoIP Business End-user is the one who makes use of the VoIP service ordered by the VoIP Business Customer for its premises or endpoints.
- MoIP Business End-user is the one who makes use of the multimedia services ordered from the corresponding customer.
- VoIP Residential Customer is the one who orders a VoIP service for its home premises.
- PSTN Business and Residential User and Customer - such users could interact either direct through the VoIP Service Provider, if this is supported by the VoIP SP using diallers, gateways, etc, or through another VASP who provides this functionality.

- Value Added Service Providers (VASPs), whose role is oriented towards customer management and value adding. Customers buy services from VASPs, which act as retailer of telco services, providing other services than connectivity.

Examples of these roles include:
- VPN Service Providers: offer to their customers the capability to define a logical view of the connectivity among customer’s endpoints. They provide the mapping from the view of the private network seen by the user to the network managed by the service provider;
- VoIP Service Providers: offer to their customer the capability to use the alternative VoIP service by utilising a packet-based connectionless network instead of the POTS synchronous connection-oriented network. Customers buy services from VASPs, which act as retailer of telco services, providing other services than connectivity.

- Network Service Providers playing both the role of Network Provider and Network Management Provider. Their role is to support VASPs to provide their services, acting basically as managed connectivity providers. They provide an interface to VASPs, which enable them to request connections between arbitrary end-points in the global network.

Network Providers are responsible for managing resources involved on service and network provision, thus providing connectivity services. The services offered by the Management Service Providers aim at fulfilling the management needs of their customer organisations, such as VASPs and business customers.
Note that these business roles play the role of user and provider towards other business roles, for example, a VASP provides services to customers – the provider role, but uses services from Network Service Providers, playing the user role. Several business relationships can be established among the different roles.

Several roles can be performed by a single business entity – the actor or player or stakeholder, at the same time. For example, a player in the Network Service Provider role can also be in the role of a VASP. This is the case for companies who own their own transport network and want to make revenue in the new markets, for example, an incumbent telecom operator.

The following actors could be defined:

- Incumbent/traditional telecommunications providers that were used to be called PNOs: these players have an incumbent position in the market, since they have extended fixed network infrastructure, both in the local loop (copper) and in the backbone all over the country. They can play multiple roles in once, such as the roles of Network Service Provider and VASP.
- Competitive/new entrant telecommunications providers: these providers are new entrants following the market deregulation and are competitive to the above. They usually don’t have extended network infrastructure in the local loop and build their backbone on top of the incumbent ones. They use the local loop either by the unbundling deregulation procedure or by owning licenses in the Wireless Local Loop. They gradually build their own backbone network. Examples of such carriers are the Internet Service Providers, who extend their activities in the telephony and other services.

### 3.3.2. Business Model Relationships

This section describes the relationships among the various roles identified in the previous section. The following picture shows the key roles and their interdependencies. The different roles are placed in the figure in a vertical positioning according to the identification of role played, as identified in the previous sections, i.e. Customers/End Users, Value Added Service Providers and Network Service Providers. In that context the different Customers/End users are presented in shades of blue, VASPs in yellow and Network Service Providers in orange. Note the important central role that the INMS takes in this figure.
3.3.3. Business Case

Operators wish to build carrier scale multi-service IP networks to provide a wealth of new multimedia services to their customers. All of these services have to be cost effectively, rapidly and reliably deployed to millions of customers.

In parallel to the deployment of these new networks, operators are faced with the challenge of upgrading or replacing their existing OSSs. Puts in sentence carriers must simultaneously deploy carrier scale network management.

The next paragraphs describe the competitive advantages that WINMAN will provide to the network service providers.

3.3.3.1 Configuration Management

- Service Provisioning
To date, service providers have opted to offer one-size-fits-all, flat-rate plans that allow customers unlimited use of standard services. The ability to offer premium-grade services such as guaranteed bandwidth, voice over IP, and VPN are dependent on the ability to define personalised user policies that map users to the class of service to which they are subscribed.

WINMAN will make policy-based service provisioning a reality, thus empowering service providers to quickly bring new differentiated services to market. In addition WINMAN will pave the way for flow-through automation in conjunction with other operations support system (OSS) applications, by means of standardised interfaces. WINMAN should consider parameters derived from Service Level Agreements – the Network Level Parameters (NLAs), and monitor their quality.

- Network Provisioning

Today, network provisioning, like the network infrastructure itself, has become increasingly complex. The manual configuration of service “pipes” on the network has become costly, time consuming, and error prone. As the demand for timely delivery of more innovative services increases, the requirement to automate the provisioning process is stronger than ever. Automating the provisioning process allows service providers to scale their operations and improve quality of end-to-end deployment for new customers.

The main two functions are:

- The Provisioning of end-to-end IP paths over light-paths using MPLS technology. In this context the Inter Domain Management System should be capable of calculating, designing and creating MPLS Label Switch Paths (LSPs) over the corresponding light-paths in the optical domain. For completeness reasons as well as for backwards compatibility the provisioning of IP plain connectivity (connectionless mode) might also be considered and is left for further study.
- Support traffic and QoS parameters for MPLS LSPs derived from Service Level Agreements – called the Network Level Parameters. Policies and other requirements will be applied in the path-provisioning request.

Secondary functions supporting the above are the ones to discover network resources, topology and maintain an inventory of all the network resources with their status and their hierarchical relationship, to notify the Service Management about service status and network parameters identified in the SLAs, to provide updates to the Fault and Performance Managers reflecting the changes in the network configuration and new services.

WINMAN will provide an integrated, automated provisioning solution for network service providers that offer IP-based network services (IP-VPN and voice services), transparently/independently from the underlying transport network (ATM, SDH or WDM).

### 3.3.3.2 Fault Management

In today’s service provider environment, it is no longer acceptable to simply monitor for a network’s device-specific events. In a world where SLAs impose penalties for network down time, network events must be correlated with effected customers. Additionally, service providers can differentiate VPN services by offering customers a “window” into events associated with their VPNs.

WINMAN will provide a service-level alarm monitoring and diagnostics tool that provides network fault monitoring, network trouble isolation, and real-time service-level management for service provider networks.

WINMAN is designed to help operators focus on important network events by offering a combination of alarm processing rules, filtering, and customisable alarm viewing. It will offer Web-based interfaces, that allow service providers to select-specific views of events and availability reporting associated with particular services.

The main two functions are:

- Report faults in the IP or Optical Domain in an intelligent and integrated way. Reporting of primary faults, either in the electrical or optical domain should be supported after the corresponding filtering,
analysis and correlation of the multiples alarms that are propagated in case of a single fault. The report should include all the attributes of the anticipated alarms together with the list of affected LSPs.

- Recover faults in the IP or Optical Domain in an integrated way. In case that automatic protection switching is not applicable in any of the domains or it fails partially or completely, then an automatic fault restoration mechanism should apply to try to restore all the affected LSPs or IP plain connectivity. This automatic procedure is triggered by the inter-domain management system after the integrated analysis and correlation of the propagated alarms.

Secondary functions supporting the above are the ones to interface with the Inter-Domain topology database to obtain network topology information, keep an inventory of user-defined correlation rules, maintain a fault topology database which contains the alarm status of the network resources and services, design management rules for the multi-layer survivability aspects (avoidance of conflicting actions, protection and restoration, service differentiation) and handle service requests from Service Management systems for the fault status of the connectivity services.

### 3.3.3.3 Performance Management

Service providers face the challenge of the ability to measure user experiences through their networks, not just network element throughput.

WINMAN will monitor key NLA metrics such as response time, availability, jitter (inter-packet delay variance), connect time, packet loss, and application performance. Also important is the ability to measure performance within a particular customer VPN.

- The main function is to monitor, filter and report performance data. The INMS shall monitor the basic traffic and QoS network parameters of the LSPs and report service degradations in case of performance gauges or counters threshold crossings.

Secondary functions are the ones to set threshold crossing alerts on the available route capacity between any two-service locations for all provided transport services/facilities or on the equipment capacity, handle notifications of capacity threshold crossings, obtain periodic and on-demand reports of the monitored capacity (traffic load) and use the obtained traffic monitoring data for identifying hot spots in the network and take actions to prevent network congestion.

To sum up, WINMAN will provide a robust, multi-product, multi-service provisioning system, able to integrate and correlate fault and performance information across technological domains (IP, ATM, SDH and WDM).

### 3.3.4. Business Processes addressed by WINMAN

As stated in chapter 2, TMF has launched a significant activity to capture the needs of network operators and service providers and thus enable the “end-to-end process automation of telecommunications and data services operations processes” [TOM2.1]. The Telecom Operations Map (TOM) is the framework for accomplishing the above mission. The TOM defines the business processes and their interactions used by Service Providers in the Customer, Service and Network Management areas. The Business Process defined in TOM Business Process Framework (BPF) are identified in Figure 3.5.

The above methodology of business processes decomposition has been adopted also by WINMAN followed by the proper adaptation and customisation. The processes in blue shade will be addressed by the WINMAN solution, either in detail or briefly, in order to show a complete demonstration case. The requirements capture following in the next chapter will be based in the TOM Business Process decomposition as will be presented in detail.
The project has adopted a semi-formal procedure, partially based on the Rational Unified Process (RUP) [24]. In detail, the approach adopted is that first the business model with all the actors interacting with the system and the high level architecture has been identified. The actors of the Business Model impose their requirements as external entities of the system, and subsequently the system requirements will derive from the needs of the external actors, as will be elaborated later in this chapter. The actors in turn initiate the so-called “use cases”, showing the different steps that the system has to follow to accomplish a task. The outcome of the use cases is the system functionality. Finally, the use cases will be further elaborated with internal actors and corresponding message sequence diagrams or scenarios, showing the interactions among the different components, having as a result the system software design.

3.3.5. **High Level Management Architecture**

In the above context the WINMAN scope is defined through a model following the TMN principles and layers. WINMAN scope focuses in the Network Management Layer of the TMN pyramid. The reference points for interactions with the outer world is the northbound interface towards other Service Management Functional Systems, such a VPN or VoIP functionality systems and southbound to the Element Management Functional Systems. In WINMAN scope are the IP and WDM technologies under consideration, since ATM and SDH functional systems will not be implemented. Finally, a presentation functional system or Workstation function in TMN terms providing a workstation reference point is provided to the WINMAN operator, as a means of controlling and monitoring the WINMAN functionality. The WINMAN solution will provide Inter-domain functionality, as well as network layer functionality for WDM and IP functional systems. For testing or demonstration purposes service layer functionality will be provided, being in principle out of the scope of WINMAN. The above are summarised in Figure 3.6.
3.4. Requirements Specification

3.4.1. Introduction

The requirements are the rules the WINMAN solution should adhere to. The requirements describe the 'what' of the system.

The approach adopted by WINMAN is to link the actors of the Business Model with the requirements of the WINMAN solution. In other words the actors of the Business Model pose their requirements as external actors to WINMAN, and then the WINMAN system requirements will derive from the needs of the WINMAN actors. The WINMAN actors are also present in the use cases, which will lead to the WINMAN system functionality and functional architecture. The use cases are coupled with the appropriate set of requirements.

The WINMAN actors related to the Business Model are depicted in the following figure as already described above. The Service layer Management Actor is represented by a Service Management System, whereas the WDM/IP Element layer Management Actors are represented by the corresponding systems.

The WINMAN requirements will be captured according to needs of these 4 external actors, namely the Service Management System, the WINMAN operator and the IP and WDM Element Management Systems.
3.4.2. Requirements Attributes

Hereafter follow the requirement attributes, on which the collection of requirements will be based. Note, that the mandatory requirements will form the requirement tables for the WINMAN solution.

**ID:**

*mandatory*

This attribute is a unique tag so it can easily be referenced to in other documents.

The ID field has a fixed format in order to group requirements by management function and management area, this format is as follows: [SYSTEM]_[AREA]_[NUMBER]

Valid values for SYSTEM are:

<table>
<thead>
<tr>
<th>SYSTEM</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATM</td>
<td>The ATM transmission layer</td>
</tr>
<tr>
<td>INM</td>
<td>The Inter domain network management layer</td>
</tr>
<tr>
<td>IP</td>
<td>The IP transmission layer</td>
</tr>
<tr>
<td>SDH</td>
<td>The SDH transmission layer</td>
</tr>
<tr>
<td>WDM</td>
<td>The WDM transmission layer</td>
</tr>
<tr>
<td>WIN</td>
<td>The WINMAN Solution in general</td>
</tr>
<tr>
<td>NM</td>
<td>The Domain Network Manager layer (common for all technology domains)</td>
</tr>
</tbody>
</table>

Valid values for AREA are:

<table>
<thead>
<tr>
<th>AREA</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CM</td>
<td>Configuration Management</td>
</tr>
<tr>
<td>COM</td>
<td>Common</td>
</tr>
</tbody>
</table>

---

**Figure 3.7:** Requirements’ capture context
The number field should have a three-digit number and should be unique through all requirements. Using the notation IP_PM_005 is requirement 5 of performance management of the IP management system. This helps human to locate quickly the requirement.

Requirements related to the Network Provisioning functionality of the TOM, will start from number 101, while requirements related to the Network Inventory Management functionality of the TOM, will start from number 201, so as to be differentiated from other requirements. All other requirements will start from 001.

**Need:**
- **mandatory**
  This attribute defines how important it is to meet a defined requirement

| M|andatory | The requirement must be met it is not negotiable |
|---|---|
| D|esired | The requirement indicates a desired target for the next release, subject to negotiation |
| T|arget | The requirement indicates a long term target value |

**Priority:**
- **optional**
  This attribute defines the priority for implementing the requirement for incremental delivery so that the developer can decide in which release it will be implemented. Although WINMAN does not consider incremental delivery (only two releases are foreseen: R0 and R1) there could be a use for this attribute if later on in the project we cannot implement all requirements and we have to decide which will be implemented and which are not.

| H|igh | This requirement should be considered first during design |
| M|edium |
| L|ow | This requirement should be considered last during design |

**Stability:**
- **mandatory**
  This attribute defines how stable the definition of the requirement is. As WINMAN will evolve in time requirements will change as well.

| F|inal | The requirement is well understood and stable |
| P|reliminary | The requirement may be changed soon |
| U|nclear | The requirement is not understood is changed with high probability |

**Type:**
- **mandatory**
  This attribute differentiates requirements from other text, which helps to understand the requirement. To make the requirements more readable I do not propose to put plain text in the table as well. Plain text can be placed outside the requirement tables.

| R|equirement | The pure requirement |
| D|escriptive | Text which gives additional information to a requirement |
| T|est | The text gives more information in what environment the requirement is valid |

**Class:**
- **optional**
  This attribute differentiates between the different classes of requirements. It should be used to check if all aspects are covered with requirements testing.

| A|dministration | How the system can be managed and configured |
**Conservation** Specifies how failures will be handled

**Documentation** Needs for the user documents

**Evolution** Conditions on how the system must evolve

**Functional** The functions of the system.

**Installation** Defines the way to install the system

**Interface** Defines elements with which the system has to interact.

**Maintainability** Repair and detection of faults

**Operational** How the system interacts with human operators.

**Performance** Numerical values for measurable variables. Quantitative specification. May be represented as a range of values: acceptance level, target, long term target

**Portability** Operating systems and platforms the system can be portable to.

**Quality** Quality attributes

**Reliability** Reliability attributes, like mean time between failures.

**Resource** Resource needs of the system

**Security** Protection of the system against intruders and external attacks

**Undefined** Not defined

**Upgrade** Needs for a future upgrade to a new version

**Verification** Constraints how the system is to be verified

**Question:**
**optional**
This attribute may store any question to the requirement

**Release:**
**mandatory**
This attribute indicates for what release the requirement is targeted. This can help to give already indications what will be necessary in future releases. It has a close relation with the Priority attribute. In WINMAN there are only two releases: R0 and R1. The use of the priority attribute should be avoided.

**Source:**
**mandatory**
This attribute indicates which partner is in charge of the requirement (i.e. the one who proposed or wrote it). It could be helpful in case a partner has doubts about a requirement and thus there is a need to clarify its meaning.

**Actor:**
**mandatory**
This attribute indicates which actor being around the WINMAN solution imposes the requirements. In this way the actors identified in the Business Model are linked with the requirements of the WINMAN system. The WINMAN system requirements will derive from the needs of the WINMAN actors. The WINMAN actors are also present in the use cases, which will lead to the WINMAN system functionality and functional architecture. Finally there will be a feedback from the use cases to the WINMAN requirements.

<table>
<thead>
<tr>
<th>Operator</th>
<th>The WINMAN Operator</th>
</tr>
</thead>
<tbody>
<tr>
<td>[SMS]</td>
<td>Service layer Management System (upper layer)</td>
</tr>
<tr>
<td>[IP EMS]</td>
<td>IP Element Management System</td>
</tr>
<tr>
<td>[WDM EMS]</td>
<td>WDM Element Management System</td>
</tr>
<tr>
<td>[All]</td>
<td>All the above impose this requirement</td>
</tr>
<tr>
<td>Other</td>
<td>Combinations of the above could be also present [SMS/Oper]</td>
</tr>
</tbody>
</table>

### 3.4.3. Requirements Identification

The requirements specification is split in two main categories:

- Functional requirements, describing the requirements of the external actors of the WINMAN solution related to configuration, fault and performance management functionality
- Non-functional requirements, such as interoperability, implementation and integration specific requirements etc.

As described in section 3.3.4, the requirements identification will be based on the TMF Telecom Operation Map (TOM) approach. In more detail the requirements are categorised per functionality belonging to one of Business Process defined in TOM Business Process Framework (BPF) from TMF. The requirements capture methodology used in the WINMAN project is documented as Appendix A.

### 3.4.3.1 Functional Requirements

#### 3.4.3.1.1 High Level Requirements

These are the high level functional requirements, which define the context of the WINMAN solution. High-level non-functional requirements will be addressed in a separate section.

<table>
<thead>
<tr>
<th>ID</th>
<th>Req Type</th>
<th>Requirement</th>
<th>Actor</th>
<th>Need</th>
<th>Stab</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>WIN_COM_001</td>
<td>All Req</td>
<td>The WINMAN solution shall perform the following functions: - Managing WDM Domain- Managing IP Domain- Managing the Inter Domain Network</td>
<td>All</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td></td>
<td>Des</td>
<td>Management of the WDM domain consists of all the Management Functions (MF) that pertain to the management of the WDM network, and supporting functions required for the execution of all related processes and communications with other MFs</td>
<td>All</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WIN_COM_002</td>
<td>All Req</td>
<td>The WINMAN solution shall be updated according to the managed network changes</td>
<td>EMS</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td>WIN_COM_003</td>
<td>All Req</td>
<td>WINMAN should monitor the status of its components and the inter and intra-systems DCN</td>
<td>Oper</td>
<td>D</td>
<td>Fin</td>
<td>Adm</td>
</tr>
</tbody>
</table>
3.4.3.1.2 WINMAN Requirements

In this section the WINMAN solution requirements are presented in detail. In order to clarify different entities in the following requirements, a small introduction is present clarifying the different WINMAN data repositories.

3.4.3.1.2.1 Data Repositories

The WINMAN solution shall consist of 3 types of databases used for network configuration purposes at different levels and another 2 types for fault and performance data:

- **Network Inventory or Physical database**
  The network inventory database is managed by the Network Inventory Management (NIM) subsystem (according to the TOM processes), and maintains the physical inventory of the network resources. The database repository resembles a hierarchical structure, representing the equipment in the network. At least the following attributes are found for each network resource of the network inventory database:
  - Cards
  - Ports
  - Fibre conduits
  - Location

- **Logical database**
  The logical database is managed by the Network Provisioning (NP) system. The logical database maintains the logical inventory of the network resources, and consists of entries representing logical paths, which are supported by the physical resources of the network. Each path is specified by an entry in the database having at least the following attributes:
  - Protection of the path
  - Available bandwidth
  - Network quality
    - Delay
    - Delay variation

- **Connectivity database**
  The connectivity database is managed by the Network Provisioning (NP) subsystem. All active or to be activated connections in the network are stored with their attributes in one entry in the connectivity database. If a particular connection is released, the corresponding entry will be removed from the connectivity database. A connection has at least the following attributes:
- Connection status
  - Pending
  - In-Effect
- The logical route through the network
- Connection bandwidth
- Connection quality
  - Delay
  - Delay variation

**Fault topology database:**
The Fault topology database is managed by the Network Management and Restoration subsystem. The Fault topology database has an entry for each root cause problem within the network, the root cause has at least the following attributes:
- Fault cause
- Fault severity
- Location

**Performance management database:**
The Performance management database is managed by the Network Data Management subsystem. The Performance management database stores the quality levels in each of the logical paths within the network.

### 3.4.3.1.2.2 Network Provisioning

The aim of the Network Provisioning Process is configuration and installation of the logical and physical network. The process ensures that network capacity is ready for provisioning and maintenance of services. It carries out the Network Provisioning, as required to fulfil specific service requests, network and information technology additions, changes, deletions and configuration changes to address network problems. The network provisioning process administers the logical network and interfaces with the network inventory management process.

<table>
<thead>
<tr>
<th>ID</th>
<th>Rel</th>
<th>Type</th>
<th>Requirement</th>
<th>Actor</th>
<th>Need</th>
<th>Stab</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>WIN_CM_101</td>
<td>All</td>
<td>Req</td>
<td>The WINMAN solution shall be able to receive provisioning requests</td>
<td>SMS/Oper</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Des</td>
<td>Paths shall be point-to-point or point-multipoint, uni or bi-directional.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Des</td>
<td>There are the following types of &quot;high level&quot; or summary level provisioning requests: Provision-Modify-Delete connectivity service</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Des</td>
<td>The user would be able to trigger &quot;second level&quot; actions, such as Create, Implement, Activate, Verify, Release, Remove, Deactivate connectivity service</td>
<td>Op</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WIN_CM_102</td>
<td>All</td>
<td>Req</td>
<td>The provisioning requests shall support QoS-related parameters</td>
<td>SMS/Oper</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Des</td>
<td>The WINMAN solution shall support QoS specifications in an abstract view which is independent of the underlying transport technology (IP/ATM/SDH/WDM)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The WINMAN solution shall be able to receive connectivity requests. The connectivity request shall refer either to the client layer (IP/MPLS) or to the server layer (WDM). The latter is useful to pre-configure the WDM network according to network planning activities, so as to enforce the desired initial IP topology.

The connectivity request should specify which QoS parameters are mandatory (shall strictly be met) and which ones are desirable (might accept a lower QoS level).

For a defined service to work properly, some parameters can be critical and other parameters can be desirable. For example, for VoIP delay is a critical parameter but bandwidth can vary provided that it is greater than 8 kbps.

The provisioning requests shall support the inclusion of a desired time for performing the associated operations (scheduled request). It should be possible to schedule requests by any combination of time-of-day (to the minute), day-of-week, day-of-month, weekly, monthly, and yearly.

The WINMAN solution shall process provisioning requests. The WINMAN solution shall select the network resources according to the requested QoS.

The WINMAN solution shall receive responses to connectivity request from the EMS systems, conveying the degree of success of the request.

It should be possible to cancel a scheduled request.

The WINMAN solution shall maintain a connectivity database for all connections inside its own network.

| WIN_CM_103 | All Req | The WINMAN solution shall be able to receive connectivity requests | Oper | M | Fin | Fun |
| WIN_CM_104 | All Req | The connectivity request shall refer either to the client layer (IP/MPLS) or to the server layer (WDM). The latter is useful to pre-configure the WDM network according to network planning activities, so as to enforce the desired initial IP topology. | Oper | M | Fin | Fun |
| WIN_CM_105 | All Req | The connectivity request should specify which QoS parameters are mandatory (shall strictly be met) and which ones are desirable (might accept a lower QoS level). | SMS/Oper | D | Fin | Fun |
| WIN_CM_106 | All Req | For a defined service to work properly, some parameters can be critical and other parameters can be desirable. For example, for VoIP delay is a critical parameter but bandwidth can vary provided that it is greater than 8 kbps. | SMS/Oper | M | Fin | Fun |
| WIN_CM_107 | All Req | The provisioning requests shall support the inclusion of a desired time for performing the associated operations (scheduled request). | SMS/Oper | M | Fin | Fun |
| WIN_CM_108 | All Req | It should be possible to schedule requests by any combination of time-of-day (to the minute), day-of-week, day-of-month, weekly, monthly, and yearly. | SMS/Oper | M | Fin | Fun |
| WIN_CM_109 | All Req | The WINMAN solution shall process provisioning requests. | SMS/Oper | M | Fin | Fun |
| WIN_CM_110 | All Req | The WINMAN solution shall select the network resources according to the requested QoS. | SMS/Oper | M | Fin | Fun |
| WIN_CM_111 | All Req | The WINMAN solution shall receive responses to connectivity request from the EMS systems, conveying the degree of success of the request. | IP/WDM EMS | M | Fin | Fun |
The WINMAN solution shall maintain the status of all connections in the connectivity database.

The WINMAN solution shall maintain a logical database storing the logical representation of network information.

The connectivity states as defined in X.721 should be supported.

The WINMAN solution shall support the management of protocols and network technologies that could be required by a client system in order to establish Real Time Services and VPNs.

The WINMAN solution shall be able to reserve resources for a connection prior to setting it up when receiving deferred/scheduled reserve request. Useful when a provisioning request defines the time at which the connectivity request has to be set up.

The WINMAN solution shall be able to activate the connectivity service upon request. Useful when the automatic activation according to a specific schedule included in the provisioning request fails.

The WINMAN solution shall be able to deactivate the connectivity service upon request. Useful when the automatic deactivation according to a specific schedule included in the reservation request fails. It is also useful when a request is received to deactivate the end-to-end path before the scheduled time.

The WINMAN solution shall be able to modify the connectivity service upon request. Useful when the automatic deactivation according to a specific schedule included in the reservation request fails. It is also useful when a request is received to deactivate the end-to-end path before the scheduled time.

The WINMAN solution shall be able to delete the connectivity service upon request.
<table>
<thead>
<tr>
<th>ID</th>
<th>Type</th>
<th>Req/Des</th>
<th>Description</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>WIN_CM_120</td>
<td>All</td>
<td>Req</td>
<td>The WINMAN Solution shall send provisioning status notifications</td>
<td>SMS/Oper M Fin Fun</td>
</tr>
<tr>
<td>WIN_CM_121</td>
<td>All</td>
<td>Req</td>
<td>The WINMAN solution should allow manual, semi-automatic and automatic routing of end-to-end paths</td>
<td>SMS/Oper D Fin Fun</td>
</tr>
<tr>
<td>WIN_CM_122</td>
<td>All</td>
<td>Req</td>
<td>The WINMAN solution shall calculate routes based on specific constraints (hops, protection, policies, etc) given by the upper layer</td>
<td>SMS/Oper M Fin Fun</td>
</tr>
<tr>
<td>WIN_CM_123</td>
<td>All</td>
<td>Req</td>
<td>The WINMAN solution shall allow the manual routing of connections in failure cases</td>
<td>SMS/Oper M Fin Fun</td>
</tr>
<tr>
<td>WIN_CM_124</td>
<td>All</td>
<td>Req</td>
<td>The WINMAN solution shall be able to offer a view of the current state of the network</td>
<td>Oper M Fin Fun</td>
</tr>
<tr>
<td>WIN_CM_125</td>
<td>All</td>
<td>Req</td>
<td>The WINMAN solution should be able to display a Network Map</td>
<td>Oper D Fin Fun</td>
</tr>
<tr>
<td>WIN_CM_126</td>
<td>All</td>
<td>Req</td>
<td>The Network Map shall provide the topology of the managed network resources (nodes, links, etc)</td>
<td>Oper D Fin Fun</td>
</tr>
<tr>
<td>WIN_CM_127</td>
<td>All</td>
<td>Req</td>
<td>The network map shall be configured either manually or automatically or semi-automatically by the operator. In the latter case consistency/synchronisation mechanisms shall be available to audit the network map information</td>
<td>Oper D Fin Fun</td>
</tr>
<tr>
<td>WIN_CM_128</td>
<td>All</td>
<td>Req</td>
<td>The network map shall allow several network layer partition view. The network map shall provide the ability to structure a complex network by means of subnetworks (or network areas).</td>
<td>Oper D Fin Fun</td>
</tr>
<tr>
<td>WIN_CM_129</td>
<td>All</td>
<td>Req</td>
<td>The WINMAN solution shall respond to every request including information about the degree of success of the request</td>
<td>All M Fin Upg</td>
</tr>
</tbody>
</table>
Des

If a request itself is not (fully) implemented a NOT_IMPLEMENTED reply shall be sent. This requirement will ensure that each defined request can be sent to each domain manager even if not all request are (fully) implemented. This will allow for early testing of the applications.

WIN_CM_128 All Req
The WINMAN solution shall send reserve and release requests to the network element management system via the southbound interface. IP/ WDM EMS M Fin Fun

WIN_CM_129 All Req
The WINMAN solution shall be able to support protected and non-protected optical paths SMS/Oper M Fin Fun

WIN_CM_130 All Req
The WINMAN solution shall be notified of all changes or additions/deletions of network resources. IP/ WDM EMS M Fin Fun

Des
This includes activation of a protection mechanism and notification of the protection results. IP/ WDMEMS M Fin

3.4.3.1.2.3 Network Inventory Management

The aim of the Network Inventory Management Process is to install and administer the physical infrastructure of the network. The responsibilities of the process concern the installation and acceptance of the equipment, the physical configuration of the network, managing the spare parts and the part return/repair sub-processes and software upgrades of the equipment.

<table>
<thead>
<tr>
<th>ID</th>
<th>Req Type</th>
<th>Requirement</th>
<th>Actor</th>
<th>Need</th>
<th>Stab</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>WIN_CM_201 All Req</td>
<td>The WINMAN solution shall build a Network Inventory (or Physical) database</td>
<td>Oper M Fin Fun</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WIN_CM_202 All Req</td>
<td>The WINMAN solution shall provide access to the Network Inventory database</td>
<td>Oper M Fin Fun</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WIN_CM_203 All Req</td>
<td>The WINMAN solution shall update its Network Inventory database upon receiving network changes and/or additions/deletions from each EMS.</td>
<td>IP/WDM EMS M Fin Fun</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WIN_CM_204 All Req</td>
<td>The WINMAN solution shall include a synchronisation mechanism for Network Inventory database.</td>
<td>IP/WDM EMS M Fin Fun</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3.4.3.1.2.4 Network Maintenance & Restoration

The aim of the Network Maintenance and Restoration Process is to maintain the network, to eliminate or minimise the impact of problems to services caused by the infrastructure, to restore or repair customer-affecting troubles quickly, and to identify problems in the network prior to the problems becoming customer affecting
problems. This process concerns the maintaining the operational quality of the network, in accordance with required network performance goals.

<table>
<thead>
<tr>
<th>ID</th>
<th>Rel</th>
<th>Type</th>
<th>Requirement</th>
<th>Actor</th>
<th>Need</th>
<th>Stab</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>WIN_FM_001</td>
<td>R1</td>
<td>Req</td>
<td>The WINMAN solution shall receive state and attribute value changes notifications from the network elements managed</td>
<td>IP/ WDM EMS</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td>WIN_FM_002</td>
<td>R1</td>
<td>Req</td>
<td>The WINMAN solution shall process state and attribute value changes notifications</td>
<td>IP/ WDM EMS</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td>WIN_FM_003</td>
<td>R1</td>
<td>Req</td>
<td>The WINMAN solution shall inform the operator of changes in the current operational status of every network element</td>
<td>Oper/ IP/ WDM EMS</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td>WIN_FM_004</td>
<td>R1</td>
<td>Req</td>
<td>The WINMAN solution shall report any faults sending alarms to the operator and propagating these alarms to the SMS through the corresponding interfaces</td>
<td>SMS/Oper</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td>WIN_FM_005</td>
<td>R1</td>
<td>Req</td>
<td>All alarms shall contain common information plus extended data specific to the particular alarm.</td>
<td>All</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
</tbody>
</table>

Des

Every alarm shall contain at least the following fields, according to ITU-T recommendation X.733:
- Event type
- Start/End time
- Network element
- Transport Entities Affected
- Perceived Severity
- Probable cause
- Acknowledgement flag
- Alarm status

This is minimum set of fields WINMAN should support

Des

The status of an alarm shall take one of the following values:
- New
- Acknowledged
- Cleared+D18

Des

Every alarm shall have a field to include additional information relevant to each type of alarm.

Des

Every alarm should have an editable field to include information that is relevant to the Operator

WIN_FM_006  | R1  | Req  | The WINMAN solution shall be able to apply alarm filtering to eliminate redundant or superfluous messages according to specific criteria that are user configurable | Oper/MS | M    | Fin  | Fun   |
<p>| WIN_FM_007 | R1 | Req | The WINMAN solution shall be able to modify the filtering criteria for alarm reporting | Oper/S MS | M | Fin | Fun |
| Des | The WINMAN solution shall be able to apply filtering criteria for alarm reporting to the operator |
| Des | The WINMAN solution shall be able to apply filtering criteria for alarm propagation to the SMS |
| Des | Filtering shall be based on the actual alarm fields values or on persistency analysis |
| Des | Persistency analysis shall allow the filtering of: |
| Des | - Threshold Alarms. This filter suppresses alarms that re-occur less than a specific number of times within a user configurable time window. |
| Des | - Repeated Alarms. Multiple occurrences of alarms within a user configurable time window are suppressed. Only one alarm of a burst will survive this filter. |
| Des | - Transient Alarms. Temporary faults that have very short life span can be suppressed according to user configurable rules |
| WIN_FM_008 | R1 | Req | The WINMAN solution shall be able to modify the filtering criteria for alarm reporting to the operator and alarm propagation to the SMS | Oper/S MS | M | Fin | Fun |
| WIN_FM_009 | R1 | Req | The WINMAN solution shall be able to enable/disable alarm reporting to the operator and alarm propagation to the SMS on a given connectivity service or group of connectivity services | SMS/O per | D | Fin | Fun |
| WIN_FM_009 | R1 | Req | The WINMAN solution shall be able to send filtering criteria for alarm reporting to the EMSs | Oper/IP, WDM EMS | M | Fin | Fun |
| Des | If the EMSs support alarm filtering, WINMAN shall send them alarm filtering information through its southbound interface in order that only alarms fulfilling with the filtering criteria are propagated to WINMAN |
| WIN_FM_010 | R1 | Req | The WINMAN solution shall be able to receive filtering criteria for alarm reporting from the SMS | SMS/O per | M | Fin | Fun |
| WIN_FM_011 | R1 | Req | Des | WINMAN northbound interface shall allow the reception of filtering criteria. Only alarms fulfilling these criteria shall be propagated to the SMS | All | D | Fin | Fun |
| WIN_FM_012 | R1 | Req | Des | The WINMAN solution shall find the root cause of a set of alarms. The system will only deliver the root cause alarm, suppressing the rest of the alarms that have the same cause. This will result in the triggering of new alarms (root alarms). | SMS/O | M | Fin | Fun |
| WIN_FM_013 | R1 | Req | Des | Information about the root cause shall include at least: - NE identifier - Identifiers of impacted connections | Oper | D | Fin | Fun |
| WIN_FM_014 | R1 | Req | Des | All events (network events, system events, alarms) shall be stored in event logs | Oper | M | Fin | Fun |
| WIN_FM_015 | R1 | Req | Des | Active events shall be stored in active events logs | Oper | M | Fin | Fun |
| WIN_FM_016 | R1 | Req | Des | Past events shall be stored in historic events logs | Oper | M | Fin | Fun |
| WIN_FM_017 | R1 | Req | Des | The user should be able to control the size of event logs by means of filtering criteria, size and time limits | Oper | M | Fin | Fun |
| WIN_FM_018 | R1 | Req | Des | The user should be able to backup, restore and export event logs | Oper | M | Fin | Fun |
| WIN_FM_019 | R1 | Req | Des | Alarms shall be displayed in a network topology map and in a graphical display of the alarmed objects | Oper | M | Fin | Fun |
| WIN_FM_020 | R1 | Req | Des | An alarm browser shall enable the display and manipulation of alarms | Oper | M | Fin | Fun |
| WIN_FM_021 | R1 | Req | Des | An event browser should be provided allowing the application of filtering and ordering criteria to its view | Oper | M | Fin | Fun |
| WIN_FM_022 | R1 | Req | Des | The WINMAN solution should contain graphical reporting tools to display alarm information in graphs and tables | Oper | D | Fin | Fun |</p>
<table>
<thead>
<tr>
<th>WIN_FM_017</th>
<th>Req</th>
<th>The user shall be able to acknowledge alarms</th>
<th>Oper</th>
<th>M</th>
<th>Fin</th>
<th>Fun</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Des</td>
<td>Alarms shall be acknowledged by means of an alarm browser or another suitable interface</td>
<td>Des</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Des</td>
<td>The alarm status shall change from 'new' to 'acknowledged'</td>
<td>Des</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WIN_FM_018</td>
<td>Req</td>
<td>The user shall be able to manually clear alarms</td>
<td>Oper</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td></td>
<td>Des</td>
<td>The alarm status shall change from 'new' to 'cleared'</td>
<td>Des</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WIN_FM_019</td>
<td>Req</td>
<td>The WINMAN solution shall perform continuity checks and tests of provided connectivity services</td>
<td>Oper</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td>WIN_FM_020</td>
<td>Req</td>
<td>The WINMAN solution shall support after communication is restored or upon user request, the synchronisation of alarm information (alarm up and alarm clear) with actual state of devices</td>
<td>Oper/IP, WDM EMS</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td></td>
<td>Des</td>
<td>Communication problems can exist among WINMAN and the EMS's or even among WINMAN internal components. In all cases, when communications are restored, the synchronisation process must be triggered</td>
<td>Des</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WIN_FM_021</td>
<td>Req</td>
<td>The WINMAN solution shall support sending a test command to the EMS and view the results from the alarm display</td>
<td>Oper</td>
<td>D</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td>WIN_FM_022</td>
<td>Req</td>
<td>The WINMAN solution shall support sending a correction command to the NEMS, and view the results</td>
<td>Oper</td>
<td>D</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td>WIN_FM_023</td>
<td>Req</td>
<td>The WINMAN solution shall be able to manage network protection mechanisms</td>
<td>Oper</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td></td>
<td>Des</td>
<td>The WINMAN solution should isolate the rest of the network from failure so that it can continue to function without interference.</td>
<td>Des</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Des</td>
<td>WINMAN shall support MPLS protection mechanisms.</td>
<td>Des</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Des</td>
<td>WINMAN shall support WDM protection mechanisms.</td>
<td>Des</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Des</td>
<td>WINMAN's northbound interface should support the definition of a level of protection.</td>
<td>Des</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**3.4.3.1.2.5 Network Data Management**

The aim of the Network Data Management Process is to ensure that network performance goals are tracked, and notification is provided when they are not met, including threshold and specific requirements for usage collection or recording. The process must provide sufficient and relevant information to verify compliance to SLAs and QoS levels.

<table>
<thead>
<tr>
<th>ID</th>
<th>Req</th>
<th>Des</th>
<th>Type</th>
<th>Requirement</th>
<th>Actor</th>
<th>Need</th>
<th>Stab</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>WIN_FM_024</td>
<td>R1</td>
<td>WINMAN shall support a sequential strategy for multi-layer (IP, MPLS, WDM) protection.</td>
<td>Des</td>
<td>WINMAN shall support a sequential strategy for multilayer (IP, MPLS, WDM) protection.</td>
<td>M</td>
<td></td>
<td>M</td>
<td></td>
</tr>
<tr>
<td>WIN_FM_024</td>
<td>R1</td>
<td>The WINMAN solution shall perform network restoration solutions in case of failure</td>
<td>Req</td>
<td>The WINMAN solution shall perform network restoration solutions in case of failure</td>
<td>SMS/O per</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td>WIN_FM_024</td>
<td>R1</td>
<td>WINMAN's northbound interface should support the restoration time parameter</td>
<td>Des</td>
<td>WINMAN's northbound interface should support the restoration time parameter</td>
<td>M</td>
<td></td>
<td>M</td>
<td></td>
</tr>
<tr>
<td>WIN_FM_024</td>
<td>R1</td>
<td>WINMAN shall perform IP restoration (rerouting) upon IP link failures.</td>
<td>Des</td>
<td>WINMAN shall perform IP restoration (rerouting) upon IP link failures.</td>
<td>M</td>
<td></td>
<td>M</td>
<td></td>
</tr>
<tr>
<td>WIN_FM_025</td>
<td>R1</td>
<td>WINMAN shall support MPLS restoration mechanisms.</td>
<td>Des</td>
<td>WINMAN shall support MPLS restoration mechanisms.</td>
<td>M</td>
<td></td>
<td>M</td>
<td></td>
</tr>
<tr>
<td>WIN_FM_026</td>
<td>R1</td>
<td>WINMAN shall support WDM restoration mechanisms</td>
<td>Des</td>
<td>WINMAN shall support WDM restoration mechanisms</td>
<td>M</td>
<td></td>
<td>M</td>
<td></td>
</tr>
<tr>
<td>WIN_FM_026</td>
<td>R1</td>
<td>WINMAN shall support a sequential strategy for multi-layer (IP, MPLS, WDM) restoration</td>
<td>Des</td>
<td>WINMAN shall support a sequential strategy for multi-layer (IP, MPLS, WDM) restoration</td>
<td>M</td>
<td></td>
<td>M</td>
<td></td>
</tr>
<tr>
<td>WIN_FM_027</td>
<td>R1</td>
<td>WINMAN (INMS) should support a co-ordination strategy for multilayer (IP, MPLS, WDM) restoration</td>
<td>Des</td>
<td>WINMAN (INMS) should support a co-ordination strategy for multilayer (IP, MPLS, WDM) restoration</td>
<td>D</td>
<td></td>
<td>D</td>
<td></td>
</tr>
<tr>
<td>WIN_FM_028</td>
<td>R1</td>
<td>The WINMAN solution shall monitor the network status and the results of restoration processes, in order to check the efficiency of the performed restorations</td>
<td>Req</td>
<td>The WINMAN solution shall monitor the network status and the results of restoration processes, in order to check the efficiency of the performed restorations</td>
<td>SMS/O per</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td>WIN_FM_028</td>
<td>R1</td>
<td>The WINMAN solution should generate auto-maintenance alarms when the system itself is faulty such as Data base overflow</td>
<td>Req</td>
<td>The WINMAN solution should generate auto-maintenance alarms when the system itself is faulty such as Data base overflow</td>
<td>SMS/O per</td>
<td>D</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td>WIN_FM_028</td>
<td>R1</td>
<td>The Winman system shall have a single, centralised scheduler for all fault management activities that require scheduling (correlation time windows)</td>
<td>Req</td>
<td>The Winman system shall have a single, centralised scheduler for all fault management activities that require scheduling (correlation time windows)</td>
<td>All</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td>WIN_FM_028</td>
<td>R1</td>
<td>The Winman System Scheduler shall trigger alarms, if activation of activity fails (i.e. defined as having acknowledge that all retries failed)</td>
<td>Req</td>
<td>The Winman System Scheduler shall trigger alarms, if activation of activity fails (i.e. defined as having acknowledge that all retries failed)</td>
<td>All</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
</tr>
<tr>
<td>Req</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WIN_PM_001</td>
<td>R1</td>
<td>Req</td>
<td>The WINMAN Solution shall monitor the QoS of the connectivity services supported.</td>
<td>SMS/Oper</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
<td></td>
</tr>
<tr>
<td>WIN_PM_002</td>
<td>R1</td>
<td>Req</td>
<td>The WINMAN Solution shall allow analysis of critical performance criteria such as delay, lost packets and availability.</td>
<td>SMS/Oper</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
<td></td>
</tr>
<tr>
<td>WIN_PM_003</td>
<td>R1</td>
<td>Req</td>
<td>The WINMAN Solution shall allow the setting of the filtering criteria for performance reporting.</td>
<td>SMS/Oper</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
<td></td>
</tr>
<tr>
<td>WIN_PM_004</td>
<td>R1</td>
<td>Req</td>
<td>The WINMAN Solution should support the start/stop, suspend/resume of performance monitoring for a given connectivity service, or group of services.</td>
<td>SMS/Oper</td>
<td>D</td>
<td>Fin</td>
<td>Fun</td>
<td></td>
</tr>
<tr>
<td>WIN_PM_005</td>
<td>R1</td>
<td>Req</td>
<td>The WINMAN Solution shall enable/disable performance reporting on a given connectivity service or group of services.</td>
<td>SMS/Oper</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
<td></td>
</tr>
<tr>
<td>WIN_PM_006</td>
<td>R1</td>
<td>Req</td>
<td>The WINMAN solution shall be able to store the Performance measurements in a PM log.</td>
<td>All</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
<td></td>
</tr>
<tr>
<td>WIN_PM_007</td>
<td>R1</td>
<td>Req</td>
<td>A browser shall enable the display of Performance measurements.</td>
<td>Oper</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
<td></td>
</tr>
<tr>
<td>WIN_PM_008</td>
<td>R1</td>
<td>Req</td>
<td>The WINMAN solution should contain graphical reporting tools to display performance information in graphs and tables.</td>
<td>Oper</td>
<td>D</td>
<td>Fin</td>
<td>Fun</td>
<td></td>
</tr>
<tr>
<td>WIN_PM_009</td>
<td>R1</td>
<td>Req</td>
<td>The WINMAN Solution shall allow the setting of thresholds in order to generate notifications when gathered data reach or cross such thresholds.</td>
<td>SMS/Oper</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
<td></td>
</tr>
<tr>
<td>WIN_PM_010</td>
<td>R1</td>
<td>Req</td>
<td>The WINMAN solution shall generate performance alarms when performance thresholds are crossed.</td>
<td>All</td>
<td>M</td>
<td>Fin</td>
<td>Fun</td>
<td></td>
</tr>
</tbody>
</table>

Des

The WINMAN Solution should collect variable values and compare to see if they exceed any of the 3 thresholds (warning, error, critical- these thresholds are user configurable).
3.4.3.2 Test Requirements

The purpose of this section is to define the service layer requirements to test the WINMAN solution. This is performed in order to test the WINMAN system. It is considered out of the scope of WINMAN, since WINMAN deals with Network Management Layer functionality.

3.4.3.2.1 Service Configuration

The aim of the Service Configuration Process is to correctly provide service configuration or re-configuration, including connection management activities, within the timeframe required to meet the ever-increasing service intervals in support of on-time delivery to customers. The process covers the installation and/or configuration of service for specific customers. The re-configuration of service after the initial service installation is also supported.

3.4.3.2.2 Service Quality Management

The aim of the Service Quality Management Process is to provide effective service specific monitoring, to provide meaningful and timely performance information and to ensure service performance meets or exceeds commitments. The process supports monitoring service of the service quality and cost on a service basis in order to determine whether service levels and costs are met consistently, or if there are problems with or improvements that can be made to a product, and whether the sale and use of the service is tracking to forecasts.
3.4.3.2.3 Network Development and Planning

The aim of the Network Development and Planning Process is to design, develop and deploy a low cost network and information technology infrastructure that meets the requirements of services provided on that network. The process encompasses the development and acceptance of network and information technology infrastructure strategies, and describes the standard network configurations primarily for operational use and defines the rules for networks, e.g. planning, installation, etc. It carries out the design of the network capabilities to meet a specific service need at the desired cost and the design, deployment and introduction of new technologies to support new services, features or enhancements. Proper installation, monitoring, controlling, billing and meeting the forecasted demand of the network must be ensured. The process also supports cases of un-forecasted demands.

### 3.4.4. Non-functional Requirements

In this section non-functional requirements are listed.

<table>
<thead>
<tr>
<th>ID</th>
<th>Rel</th>
<th>Type</th>
<th>Requirement</th>
<th>Actor</th>
<th>Need</th>
<th>Stab</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>WIN_NF_001</td>
<td>All</td>
<td>Req</td>
<td>The WinMan Data Communications Network should not be depending on the transport network being managed</td>
<td>All</td>
<td>D</td>
<td>Fin</td>
<td>Int</td>
</tr>
<tr>
<td>WIN_NF_002</td>
<td>All</td>
<td>Req</td>
<td>The WINMAN Solution shall have a mechanism for internal and external clock synchronisation.</td>
<td>All</td>
<td>M</td>
<td>Fin</td>
<td>Adm</td>
</tr>
<tr>
<td>WIN_NF_003</td>
<td>All</td>
<td>Req</td>
<td>The WINMAN solution shall be based on specific methodologies providing the necessary guidance for specifying, designing, implementing and testing efficient management services and systems</td>
<td>Oper</td>
<td>M</td>
<td>Fin</td>
<td>Int</td>
</tr>
<tr>
<td>WIN_NF_004</td>
<td>All</td>
<td>Req</td>
<td>The WINMAN design and implementation shall be based on specific standards, so as to be able to inter-operate with other systems over well-defined interfaces</td>
<td>All</td>
<td>M</td>
<td>Fin</td>
<td>Por/Int</td>
</tr>
<tr>
<td>Requirement</td>
<td>Type</td>
<td>Req</td>
<td>Description</td>
<td>Status</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------------</td>
<td>------</td>
<td>-----</td>
<td>-------------</td>
<td>--------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WIN_NF_005</td>
<td>All</td>
<td>Req</td>
<td>The WINMAN solution implementation should be tested thoroughly to identify misbehaviour and other defects. Testing conditions shall be simulated closest to the real ones.</td>
<td>All</td>
<td>D</td>
<td>Fin</td>
<td>Mai/Ver</td>
</tr>
<tr>
<td>WIN_NF_006</td>
<td>All</td>
<td>Req</td>
<td>The WINMAN solution should be integrated with a specified methodology covering both the internal system integration, but also the integration and adaptation with external systems.</td>
<td>Oper</td>
<td>D</td>
<td>Fin</td>
<td>Ver</td>
</tr>
<tr>
<td>WIN_NF_007</td>
<td>All</td>
<td>Req</td>
<td>The WINMAN solution should be tested in real conditions before put into operation</td>
<td>All</td>
<td>D</td>
<td>Fin</td>
<td>Ver</td>
</tr>
<tr>
<td>WIN_NF_008</td>
<td>All</td>
<td>Req</td>
<td>The performance of the WINMAN solution should be tested while in operation to fine tune its performance parameters</td>
<td>All</td>
<td>D</td>
<td>Fin</td>
<td>Per</td>
</tr>
<tr>
<td>WIN_NF_009</td>
<td>All</td>
<td>Req</td>
<td>The operational parameters of the WINMAN solution should be fine tuned while in operation</td>
<td>All</td>
<td>D</td>
<td>Fin</td>
<td>Ver</td>
</tr>
<tr>
<td>WIN_NF_010</td>
<td>All</td>
<td>Req</td>
<td>The WINMAN solution software should be upgraded if corresponding needs appear (bugs, new network elements, etc.)</td>
<td>All</td>
<td>D</td>
<td>Fin</td>
<td>Upg</td>
</tr>
<tr>
<td>WIN_NF_011</td>
<td>All</td>
<td>Req</td>
<td>WINMAN shall be well documented</td>
<td>All</td>
<td>M</td>
<td>Fin</td>
<td>Doc</td>
</tr>
<tr>
<td>WIN_NF_012</td>
<td>All</td>
<td>Req</td>
<td>All Management Functions shall be designed as distributed processes not necessarily on one and the same processing platform</td>
<td>All</td>
<td>M</td>
<td>Int</td>
<td>Por</td>
</tr>
<tr>
<td>WIN_NF_013</td>
<td>All</td>
<td>Req</td>
<td>The distributed MFs shall be platform independent</td>
<td>All</td>
<td>M</td>
<td>Fin</td>
<td>Por</td>
</tr>
<tr>
<td>WIN_NF_014</td>
<td>All</td>
<td>Req</td>
<td>WINMAN shall have a mechanism to control the activation/deactivation of its functions</td>
<td>Oper</td>
<td>M</td>
<td>Fin</td>
<td>Adm</td>
</tr>
<tr>
<td>WIN_NF_015</td>
<td>All</td>
<td>Req</td>
<td>Interactions between WINMAN components shall be minimised in order to minimise the bandwidth required by the system</td>
<td>Oper</td>
<td>M</td>
<td>Fin</td>
<td>Per</td>
</tr>
<tr>
<td>WIN_NF_016</td>
<td>All</td>
<td>Req</td>
<td>WINMAN solution shall avoid replication of information</td>
<td>All</td>
<td>M</td>
<td>Fin</td>
<td>Per</td>
</tr>
</tbody>
</table>
4. WINMAN USE CASES (UCs)

4.1. Introduction

The identification and description of appropriate Use Cases that illustrate the basic functionality of the targeted system, constitute the next step in defining the system’s architecture after the requirements capture. The architecture can only be fully defined when the message sequence charts (MSCs) and class diagrams are defined. In this document only textual use cases are specified.

4.1.1. UCs Actor’s Model

The following Figure 4.1 depicts the WINMAN Actors Model considered for the production of the Use Cases.

**Figure 4.1:** The WINMAN actors model

**ASSUMPTION#1:** It has been agreed at the current stage (T2.2 lifetime) not to identify internal WINMAN actors in UCs (IDFS, WDM-NMFS, IP-NMFS, etc), and refer to these entities as the “WINMAN Solution”. Moreover, ATM and SDH external actors will also not be considered as they lie outside the WINMAN implementation scope. Therefore the actual model considered for the production of the UCs is depicted in the following figure:
ASSUMPTION#2: Separate use cases were built for the SMS system and the Operator, where appropriate, even though they look the same at the highest level. At lower levels they may be different. The distinction between SMS and WO triggered UCs mainly reflects the need for automatic and manual procedures.

4.1.2. Transport entities terminology adopted

Appropriate terminology has been used for building the Use Cases (UC). Relevant references include (but are not limited to) [G.805], [G.852.2], [TMF509], [M.3120], [G.cls].

Further description about the terminology and the entities used in WINMAN can be found in chapter 5.3.

4.2. Identification & Classification of UCs

The following table depicts the names and allocation of UCs into appropriate categories.

The horizontal classification of UCs relates to the functional grouping of UCs and is compliant to [TMF-509] in terms of the terminology used. However, the Policies Provision and Scheduler groups of UCs are introduced by WINMAN.

The vertical classification of UCs is according to the level of abstraction and related broad/narrow context and modularity of the UCs. See UCs Template in Appendix B for more.

An identifier is allocated to each UC (in parenthesis). The scheme adopted is XYZ, where X denotes the horizontal classification (functional grouping), Y denotes the vertical classification (level) and Z is a sequential number.

<table>
<thead>
<tr>
<th>Summary level (S)</th>
<th>User level (U)</th>
<th>Atomic level (A)</th>
</tr>
</thead>
</table>

Figure 4.2: UCs actors model
<table>
<thead>
<tr>
<th>Policies Provision (P)</th>
<th>Provision Policies (PS1)</th>
<th>Add Policies (PU1)</th>
<th>Check Policy (PA1)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Provide Policies Report (PS2)</td>
<td>Modify Policies (PU2)</td>
<td>Find Policy (PA2)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Execute Policies (PU3)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Activate Policies (PU4)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Remove Policies (PU5)</td>
<td></td>
</tr>
<tr>
<td>Scheduler-related (S)</td>
<td>Schedule Task (SU1)</td>
<td>Delete Task (SU2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Modify Task (SU3)</td>
<td></td>
</tr>
<tr>
<td>Network Provisioning (N)</td>
<td>Provide ICS-SMS (NS1)</td>
<td>Verify CS Request (NU1)</td>
<td>Compute list of IP Routes (NA1)</td>
</tr>
<tr>
<td></td>
<td>Provide ICS-WO (NS2)</td>
<td>Create CS-SMS (NU2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Modify ICS-SMS (NS3)</td>
<td>Create CS-WO (NU3)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Modify ICS-WO (NS4)</td>
<td>Implement CS (NU4)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Delete ICS-SMS (NS5)</td>
<td>Activate CS (NU5)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Delete ICS-WO (NS6)</td>
<td>De-activate CS (NU6)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Release CS (NU7)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Remove CS (NU8)</td>
<td></td>
</tr>
<tr>
<td>Network Inventory Management (I)</td>
<td>Register EMS (IS1)</td>
<td>Configure Network (IU1)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Un-Register EMS (IS2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Notify Network Change (IS3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Add IP node (IA1)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Delete IP node (IA2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Add WDM node (IA3)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Delete WDM node (IA4)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Add WDM link (IA5)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Delete WDM link (IA6)</td>
<td></td>
</tr>
<tr>
<td>Network Maintenance &amp; Restoration (M)</td>
<td>Recover Network (MS1)</td>
<td>Identify Affected Connections (MU1)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Revert Network (MS2)</td>
<td>Identify Root Cause (MU2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Open Alarm (MS3)</td>
<td>Synchronize Alarms (MU3)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Clear Alarm (MS4)</td>
<td>Acknowledge Alarm (MU4)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Provide Alarm Reports (MS5)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Network Data Management (D)</td>
<td>Process Performance Measurements (DS1)</td>
<td>Calculate Aggregations &amp; Statistics (DU1)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Provide PM Report (DS2)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4.3. UCs Overview

This section presents an overview of all identified UCs. Critical aspects regarding their context are briefly discussed, while UCs interrelations are depicted in UML diagrams, according to the previous classification. Due to the nature of interactions, some UCs appear in multiple diagrams. A short description of the goal and context of each UC is also provided. A more detailed description of the UCs (one following the WINMAN UCs template) can be found in Appendix B.

4.3.1. Policies Provision (P)

4.3.1.1 Introduction to Policy Based Network Management systems

To aid understanding the operation of the PBNM system, the architecture of the IETF is provided, because it has been adopted as a reference model in WINMAN. This architecture is represented in Figure 4.3 and comprises the following entities:

Policy console
It is the user interface to construct policies, deploy policies, and monitor status of the policy-managed environment.

Policy decision point (PDP)
It is the entity that makes decisions (policy actions) based on the fulfilment of some conditions (policy conditions).

Policy enforcement point (PEP)
It is the entity that enforces a policy decision.

Policy repository
A directory and/or other storage service (e.g. relational database) where policies and related information are stored.

Policy communication protocols
A protocol to read/write data from the policy repository (e.g. LDAP), and a protocol to communicate between PDP and PEP (e.g. COPS).
This architecture will be used to add new policies, modify existing ones, activate policies when necessary, and to remove a policy when it is no longer needed, as summarised hereafter:

**Add Policy**
The WINMAN Operator is enabled by means of the appropriate GUI to edit a policy and introduce it in the WINMAN system. The policy will be checked against inconsistencies and conflicts prior to storing it in the policy repository.

**Modify Policies.**
The modification of a policy only affects the policy stored in the repository. The modification will take effect when the policy is retrieved to be instantiated.

**Activate Policies**
The policy is activated when it is retrieved from the repository and it is appropriately instantiated in the appropriate system components in the model of Figure 4.3. For instance, the instantiation of a policy might consist of downloading a routing table in the PEP of the IP management system. This phase is usually triggered by the scheduler, but it could be also triggered directly by the WINMAN Operator, or as a consequence of the Add Policies itself (imagine that we have only one routing table, in this case it is downloaded just after the storage).

**Remove Policies**
The policy is removed from the policy repository. In that case, the policy that had been previously instantiated is not affected.
Execute Policies
The execution of a policy is triggered by the arrival of events considered in UCs for Network Provisioning, Network Data Management, and Network Maintenance & Restoration. The execution of a policy consists in checking the relevant conditions (rules) and if true, carrying out the actions specified in the policy. For instance, assume that a request to set-up a connectivity service arrives from the SMS. This request might carry policy parameters that affect route selection, because a policy was previously instantiated for this purpose.

4.3.1.2 Management Policies overview

4.3.1.2.1 Authorisation Policies

Authorisation policies define what activities a member of the subject domain can perform on the set of objects in the target domain. These are essentially access control policies, to protect resources and services from unauthorised access. A positive authorisation policy defines the actions that subjects are permitted to perform on target objects. A negative authorisation policy specifies the actions that subjects are forbidden to perform on target objects. Authorisation policies are implemented on the target host by an access control component.

4.3.1.2.2 Information Filtering Policies

Filtering policies are needed to transform the information input or output parameters in an action. For example, a location service might only permit access to detailed location information, such as a person is in a specific room, to users within the department. External users can only determine whether a person is at work or not. Positive authorisation policies may include filters to transform input or output parameters associated with their actions, based on attributes of the subject, or target, or on system parameters (e.g. time). In many cases it is not practical to provide different operations as a means of selecting the information. Although these are a form of authorisation policy they differ from the ones described in 4.3.1.2.1, in that it is not possible for an external authorisation agent to make an access control decision based on whether or not an operation, specified at the interface to the target object, is permitted. Essentially the operation has to be performed and then a decision made on whether to allow results to be returned to the subject or whether the results need to be transformed. Filters can only be applied to positive authorisation actions.

4.3.1.2.3 Delegation Policies

Delegation is often used in access control systems to cater for the temporary transfer of access rights. However the ability of a user to delegate access rights to another must be tightly controlled by security policies. This requirement is critical in systems allowing cascaded delegation of access rights. A delegation policy permits subjects to grant privileges, which they possess (due to an existing authorisation policy, to grantees to perform an action on their behalf e.g. passing read rights to a printer spooler in order to print a file. A delegation policy is always associated with an authorisation policy, which specifies the access rights that can be delegated. Negative delegation policies forbid delegation. Note that delegation policies are not meant to be used for assignment of rights by security administrators.

4.3.1.2.4 Refrain Policies

Refrain policies define the actions that subjects must refrain from performing (must not perform) on target objects even though they may actually be allowed to perform the action. Refrain policies act as restraints on the actions that subjects perform and are implemented by subjects. Refrain policies have a similar syntax to negative authorisation policies, but are enforced by subjects rather than target access controllers. They are used for situations where negative authorisation policies are inappropriate – we do not trust the targets to enforce the policies (e.g. they may not wish to be protected from the subject).

4.3.1.2.5 Obligation Policies

Obligation policies specify the actions that must be performed by managers within the system when certain events occur and provide the ability to respond to changing circumstances. For example, security management
policies specify what actions must be specified when security violations occur and who must execute those actions; what auditing and logging activities must be performed, when and by whom. Management policies could relate to management of QoS, storage systems, software configuration etc. Obligation policies are event-triggered and define the activities subjects (human or automated manager components) must perform on objects in the target domain. Events can be simple, i.e. an internal timer event, or an external event notified by monitoring service components e.g. a temperature exceeding a threshold, or a component failure. Composite events can be specified using event composition operators.

4.3.1.3 Modes of Operation & Liaison with Policies

There are three possible modes of operation for WINMAN: manual, semiautomatic and automatic.

4.3.1.3.1 Manual

In this mode of operation the WINMAN operator does everything, i.e. manually sets up the WDM and IP paths as he/she wishes. There are no policy-related interactions in the system since the WO is doing everything. The following types of policies apply:

Authorisation Policies
Authorisation Policies will apply for both the WINMAN operator and the SMS. The WINMAN Operator and the SMS will have some complementing access rights.

Information Filtering Policies
SMS will not have any views of the system, whereas the WINMAN operator will have full view.

Obligation Policies
Obligation policies will apply for the WINMAN operator in the form of notifications (e.g. set up of an IP path over non-existing WDM paths)

4.3.1.3.2 Semiautomatic

In this mode of operation the WINMAN system does everything apart from decision making, which is carried out by the operator. For example, when new paths are being instantiated by the system, the WINMAN operator will get a list of possible paths from which one can be selected for used for servicing a specific connectivity request.

Authorisation Policies
Authorisation Policies will apply for both the WINMAN operator and the SMS. The WINMAN Operator and the SMS will have complementing access rights, up to a certain point.

Information Filtering Policies
SMS & WINMAN operator will have full view of the system

Obligation Policies
Obligation policies will apply for the SMS (e.g. set up of an IP path over non-existing WDM trails)

4.3.1.3.3 Automatic

In this mode of operation the WINMAN system is doing everything, including decision making. The WINMAN operator and the SMS will be able to use the WINMAN system. That is because all the decision-making is carried out by the policies running on the WINMAN system. Policies can be used to carry out constrained policy-based routing, different QoS levels, etc. without any interaction between the SMS and WINMAN. Only the WINMAN operator will be able to add, modify, remove policies to/from the system.

Authorisation Policies
Access control Policies will apply for both the WINMAN operator and the SMS. The WINMAN Operator and the SMS will have client guest user access profile, with the WINMAN operator having some extra functionalities (add, modify, remove policies to/from the system).

**Information Filtering Policies**
SMS & WINMAN operator will have full view of the system

**Obligation Policies**
Obligation policies will apply for the SMS (e.g. set up of an IP path over non-existing WDM paths)

### 4.3.1.4 Policy UCs Diagram

Figure 4.4 depicts the Policy provision UCs considered in WINMAN.

**Figure 4.4: Policy provision UCs**

#### 4.3.1.5 Provide Policies UC

The purpose of “Provide policies” at summary level, is to allow the WINMAN operator to add/modify/execute/delete a policy in/from the WINMAN system.
4.3.1.6 Provide Policies Report UC

This UC at User level creates reports regarding the policies that exist in the system. The nature of reports is quite generic, ranging from simple statistics to presentation of more complex policy related data at summary level.

4.3.1.7 Add Policies UC

The purpose of this UC at User level is to install a new policy in the WINMAN system leaving it ready to be activated.

4.3.1.8 Modify Policies UC

The WINMAN Operator modifies an existing policy, at User level.

4.3.1.9 Activate Policies UC

The purpose of “Activate Policies” (at User level) is to instantiate a certain (set of) policies in the appropriate system components, leaving it ready to be used by the Execute Policies UC.

4.3.1.10 Execute Policies UC

The purpose of “Execute Policies” (at User level) is to carry out actions according to an instantiated policy. The policy conditions are checked and the relevant actions are executed. This UC may be called from any of the NP, NIM, NM&R and NDM UCs.

4.3.1.11 Remove Policies UC

This use case (User level) allows the WINMAN operator to remove a policy because it will be no longer used. The policy to be removed is in the repository. The system must have a policy validation module.

4.3.1.12 Check Policy UC

The Check Policy (Atomic level) is intended to determinate the policy validity in itself and also if it is in conflict with all other existing policies. The principal mechanisms driving this check will be dictated by rules specifically designed for such purpose. These rules are called meta-policies.

4.3.1.13 Find Policy UC

The purpose of this use case at Atomic level is to find in the repository the policy required. WINMAN has a repository and a location service.

4.3.2. Scheduling (S)

4.3.2.1 WINMAN Scheduler Functionality

The scheduler is a part of functionality in the WINMAN system that can perform a variety of tasks in the system, on areas as (but not limited to):
- Connection management
  (Implementing /Activating a network connection etc)
- Configuration management
  (Periodically making changes in the network)
- Fault Management
  (Providing FM summary reports once or on a certain time base, or changing correlation rules)
Performance Management
(Providing Threshold reports or changing, thresholds)
System Administration
(Performing backups etc)
Policy Management
(Executing policies etc)
All of these tasks can be scheduled and performed by the scheduler in the WINMAN system on one condition:
the type of task must somehow be defined.

4.3.2.2 Tasks to be scheduled

The scheduler keeps a list of tasks to be scheduled, these tasks are in the first place performed in a
chronological fashion (of course), and the other parameter that determines execution order is the task priority.
However policies that apply to the scheduler can overrule the execution order.
Typical task request properties are:
Task type
Task priority
Date/Time or periodical schedule of task
Failure situation
Number of retries
Cause alarm when task activation fails
Task Owner
Output destination (prompt to Operator, etc)

The task to be scheduled can have a fixed date and time, but can also have a periodical schedule. Another
property of a task is the failure situation that determines how the scheduler should act on failure situations.
A task owner is considered the actor or use case entitled to request the scheduling of a task, two possible
owners are distinguished; either a task is owned by the WINMAN Operator or by connection management
(Provide ICS/Delete ICS/Modify ICS). The WINMAN Operator can only delete/modify a task when it had also
scheduled it. In case that connection management is the owner of a scheduled task, only a connection
management use case is able to modify/delete the task (not even the WINMAN Operator can perform such
action directly). For instance, a task is scheduled by Provide ICS for connection establishment and can be
modified/deleted by Modify ICS and Delete ICS. Of course, the WINMAN Operator is able to remove such a
task by using one of the latter connection management use cases.

4.3.2.3 Scheduler Issues

A responsibility of the scheduler that has not been captured in a use case description is the execution of a task
from the task list of the scheduler. The reasons why not to compose a UC description of the task execution are:
Methodology wise: No superordinate or actor of a task execution use case can be identified.
The relation between a task execution use case and the use case to be triggered is not covered by the
terminology.
Every use case must have a set of goals of which a subset might be the responsibility of a plural number of
subordinate use cases. The task execution use case has no goals in common with the use case to be
triggered.
The actions to be performed when a task is executed is an internal WINMAN matter and is not involved in
any external interaction.
From a practical perspective: How tasks must be executed is not entirely clear, possible approaches are
mentioned below.

The most important issue related to task execution is consistency regarding policies. When for instance a task is
scheduled, a policy may be created or modified such that at the time of task execution this will be in conflict
with the policies.

The following approaches can be taken:
It is not allowed to update policies that are active in the WINMAN system.
The WINMAN Operator is informed of the fact that a policy will be conflicting with the current state of the WINMAN system. WINMAN Operator remains to be responsible for introducing a conflicting policy. All future decisions will be aligned with the existing policies. Changes to policies will have impact on decisions future decisions only. System entities that have been subjected to policy decisions in the past will re-comply to a modified policy.

4.3.2.4 Scheduler UCs Diagram

Figure 4.5 depicts the UCs that implement the WINMAN scheduling functionality.

![UC Diagram](image)

**Figure 4.5:** Policy provision UCs

4.3.2.5 Schedule Task UC

A task is added to the task list of the scheduler.

4.3.2.6 Delete Task UC

A task is deleted from the task list of the scheduler.

4.3.2.7 Modify Task UC

A task is modified in the task list of the scheduler.

4.3.3. Network provisioning (NP)

4.3.3.1 Types of operations

In the use cases description we consider three types of operation: automatic, semi-automatic and manual. See also relevant discussion in the Policies section.

In **automatic mode** the user (SMS or WINMAN Operator) just ask for a connection between two or more end points with a set of constraints. The system calculates all IP paths that connect the end points and respect the constraints and selects the best one.

In **semi-automatic mode** the system works just like in automatic mode, but with one difference: instead of choosing the best path by itself it lets the WINMAN Operator choose the one he wants.
In the **manual mode** of operation the WINMAN Operator design the path. WINMAN then validates the path checking if it really connects the end points and if it can guarantee the bandwidth and QoS parameters requested.

### 4.3.3.2 WINMAN support for connectivity

WINMAN supports two types of connectivity services:

- **IP connectivity** relating to setting up and maintaining end-to-end IP paths. This type of connectivity service is available in all operational modes as specified above, and can be managed by both the WINMAN Operator and the Service Management System. IP connectivity is considered as the main focus of the WINMAN solution and includes the setup of connections in the WDM layer to accommodate overlying IP connections in a transparent manner.

- **Optical connectivity** relating to setting-up and maintaining optical trails by the WINMAN Operator in the manual mode only. This feature is valuable in cases where it is desirable to manually setup connections in the WDM layer for e.g. optimising network capacity / performance, or dealing with optical element malfunctions.

### 4.3.3.3 ICS and Network Resource states

Each step performed by the WINMAN system in the *Provide ICS, Delete ICS and Modify ICS* use cases brings the ICS and the network resources into another state.

#### 4.3.3.3.1 ICS states

Figure 4.6 provides an overview of the successive states of an ICS. The ovals represent the states and the labelled arrows represent the processes to be executed to get from one state to another. The following **ICS states** are distinguished:

- **Reserved**
  The ICS is created, which means that the network connection(s) is (are) designed and the required network resources are determined and provisioned (parameters have their values). The design of the ICS is recorded in the WINMAN system. The network resources are reserved in the interval(s) in which the ICS becomes operational. The WINMAN system is ready to implement the ICS.

- **Implemented**
  The ICS is implemented, which means that the network resources are allocated and thus can not be used by another ICS. The managed network is accessed for the allocation of the physical network resources and the testing of the network connection(s). The WINMAN system is ready to activate the ICS.

- **Operational**
  The ICS is in operational state, which means that the network connection(s) is (are) available to the end-users, and maintenance processes have started for this ICS.
4.3.3.3.2 Network resources usage states

Figure 4.7 depicts the network resource usage states and how the resource changes from one state to another in consequence of the ICS configuration processes as defined in Figure 4.6. The definition of the usage state is taken from [X.731]:

**Idle**: The resource is currently not in use

**Active**: The resource is in use, and has sufficient spare operating capacity to provide for additional users simultaneously

**Busy**: The resource is in use, but it has no spare operating capacity to provide for additional users at this instant

---

**Figure 4.7: Network resource state diagram**

1. Implement ICS (non-shared resource)
2. Release ICS (non-shared resource)
3. Implement (first) ICS (shared resource)
4. Release (last) ICS (shared resource)
5. Implement (new) ICS (shared resource)
6. Release ICS (shared resource)
7. Release ICS (shared resource)
8. Implement (new) ICS (shared resource)
4.3.3.3.3 Network resources reservation states

In this model the state of the resource indicates whether the state can be reserved by one ICS in a given moment or not. There are two possible states:

**Free**: the resource is not reserved by any ICS.

**Reserved**: the resource has been reserved for a ICS.

The description of a network resource contains a Scheduler attribute, which holds a list of time intervals for which the resource is reserved by one or more ICSs, as depicted in Figure 4.8.

![Figure 4.8: Reservation State](image)

It can be seen that while at the current moment the resource is not reserved, and thus it can be used to fulfil other ICS requests, from T1 to T2 the resource is expected to be in use by ICS S1, and therefore it is not available for reservation by other ICSs during that interval.

4.3.3.3.4 Multi-Session ICS

When considering the services stated in the business model (such as MoIP and VPN), it is necessary for WINMAN to be able to provide network connectivity during non-consecutive time-intervals. This will cause the ICS to consist of a number of sessions corresponding with these time-intervals. Figure 4.9 illustrates the case of an ICS that consists of two sessions, during these sessions the actual network connectivity is provided by WINMAN.

![Figure 4.9: Multi-session ICS](image)

When considering services that will stop after a certain period, the WINMAN Operator or the SMS will specify the time the ICS is no longer needed. In case of a continuous service, the ICS has no end time, therefore the number of sessions within an ICS can be anything between 1 and \( \infty \).

When the SMS tries to provision a network connection, a time is specified when network connectivity is required and a time when this network connection is no longer needed. When the latter time has been reached, SMS must issue a Delete or Modify ICS request.

4.3.3.4 Provide ICS UC

Figure 4.10 depicts the UCs involved in providing an ICS.
Figure 4.10: Provide ICS UCs

This summary level Use Case encompasses the functionality involved in providing a connection between endpoints, and uses several user-level use cases to fulfil its goals. These include all the steps from connection request verification up to the activation of the connection (bringing it into the operational state). Provide is capable of connecting IP endpoints, including the setting up of connections in the WDM layer in a transparent way.

Two separate UCs are defined for supporting the provision of IP connectivity services:

**Provide ICS by the SMS**: The Service Management System requests from the WINMAN system to provide an ICS.

**Provide ICS by the WO**: The WINMAN Operator requests the WINMAN system to provide an ICS.

**Parameters involved in Provide ICS**

The following set of parameters is defined for requests of the SMS/WINMAN Operator to Provide/Modify an ICS or operations involved in these steps:

**Connection ID**

**Connection end points**

**Scheduling parameters**

Parameters related to scheduling are for instance when an ICS must be activated

**Policies**

The policies involved in the request, like routing constraints

**Connection directionality**

If a connection is bi-directional or unidirectional

**NLA parameters**

Negotiable abstract network parameters, such as:

- Delay
- Delay variation
- Bandwidth
- Error rate in abstract terms (BER for instance)
Protection

Mode of Operation
Automatic
Semi-automatic
Manual
All parameters should be free from specific network technology terminology.

Steps involved in Provide ICS

The following processes (steps in the use cases) are distinguished when providing the ICS:

Verify ICS request
This step describes the WINMAN system responsibilities for the verification and validation of the ICS request to the WINMAN system. The WINMAN system verifies the correctness of all the attributes of the request and determines the validity of the request itself (if applicable) by applying policy rules to this request.

Create ICS
This step describes the WINMAN system responsibilities for the design of the network connection(s) that support(s) the ICS and the determination of the required network resources for these network connection(s). The design is filed in the WINMAN system. The required resources are reserved for the timeslots in which the ICS will be operational.
The ICS state becomes Reserved.

Implement ICS
This step describes the WINMAN system responsibilities for the implementation of the network connections. The implementation starts when the ICS is on the point of becoming operational (scheduling). The WINMAN system allocates the required network resources and changes their usage state from Idle to Active or Busy. So far, the managed network has not been involved in the implementation. After that the WINMAN system allocates the physical resources in the managed network through the IP and/or WDM EMSs. Subsequently, the parameters of the network resources are provisioned, and the network connection is set-up and tested.
The ICS state becomes Implemented.

Activate ICS
This step describes the WINMAN system responsibilities for the activation of the connectivity ICS. This step involves the interaction of the WINMAN system with the managed network, which leads to the operational state of the IP connectivity ICS.
The ICS state becomes Operational.

4.3.3.5 Modify ICS UC

Figure 4.11 depicts the UCs involved in modifying a connectivity service within WINMAN.
The "Modify IP connectivity service" use case is used when one or more service parameters need to be changed. The WINMAN operator and the SMS can request to change the service characteristics.

Two separate UCs are defined for supporting the modification of connectivity services:

**Modify ICS by the SMS**: The main flow described in this UC is a modification of the service in automatic mode. The modification is done after a single request from the SMS. A step by step modification, driven by several requests from the SMS, is described as an alternative flow.

**Modify ICS by the WO**: This UC describes how the WINMAN operator can modify an IP connectivity service.

The IP flexibility allows modifying a service without service outages. This means that with the IP connectivity service in active state WINMAN can implement the changes on the network.

The following modifications of a service have been identified:

---

**Figure 4.11: Modify ICS UC**
Change of the Network Level Agreements, such as, but not limited to, bandwidth, reliability (availability or protection level), delay, jitter or packet loss. This modification may not require changes in the managed network, if the existing IP path that supports the service meets the new QoS parameters.

Add or remove a leg to a point-to-multipoint or multipoint-to-multipoint service. This functionality will not be included if these services are not included in WINMAN.

Change the scheduling (initial or final time) of the service

The use case can be triggered by the SMS, or the WINMAN operator. In the first case, the process will be automatic, that is, the SMS request WINMAN to modify the service parameters and WINMAN carries out the needed actions. In the other case, the operator can have more control over the process.

Other considerations about the Modify ICS use case:

The Modify ICS request may or may not cause changes in the network connectivity

This depends whether the current IP path(s) meets the new NLA requirements, or whether the same network resources are available in the new timeslots when scheduling changes. If not, another IP path(s) must be designed and/or additional WDM resources may be needed to support certain links in the IP route.

The following table gives an overview of possible required changes in the network connections due to an ICS modification:

<table>
<thead>
<tr>
<th>ICS modification</th>
<th>IP path(s)</th>
<th>WDM connectivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>NLA</td>
<td>No change</td>
<td>No change</td>
</tr>
<tr>
<td></td>
<td>No change</td>
<td>Change</td>
</tr>
<tr>
<td></td>
<td>Change</td>
<td>No change</td>
</tr>
<tr>
<td></td>
<td>Change</td>
<td>Change</td>
</tr>
<tr>
<td>End points</td>
<td>Change</td>
<td>No change</td>
</tr>
<tr>
<td></td>
<td>Change</td>
<td>Change</td>
</tr>
<tr>
<td>Scheduling</td>
<td>No change</td>
<td>No change</td>
</tr>
</tbody>
</table>

The current IP path(s) can meet the new NLA requirements

The current IP path(s) can not meet the new NLA requirements, but has not to be changed. It requires at certain links in the IP path other/additional supporting resources from the WDM network.

The current IP path(s) can not meet the new NLA requirements. Certain links in the path or the complete path require re-routing. The WDM layer is able to support the new IP path(s), so, no changes are necessary in the WDM layer.

The current IP path(s) can not meet the new NLA requirements. Certain links in the path or the complete path require re-routing. In the WDM layer other/additional resources have to be created to support the new IP path.

If access points change (move and/or add) then the IP path(s) changes. In this case no changes are necessary in the WDM layer to support the modified IP path(s)

In this case additional resources in the WDM network are required

It is assumed that a new scheduling scheme will not require any changes in the IP and/or WDM network. But one could argue that in the new timeslots other network connections are required as the resources of the current connections are not available in these new timeslots.

The following figure shows the possible network modifications required due to the ICS modification:

- Full modify
- Partial modify
- Extend modify
- No modify
Figure 4.12: Possible route modifications of an ICS

The Modify can be Service Affecting and Non Service affecting with respect to the ICS

Service Affecting (SA)
In this case the ICS will be interrupted when making the changes in the network and data may get lost.
This allows that the current ICS is deactivated before making the new ICS operational.

Non-Service Affecting (NSA)
In this case it is not allowed to interrupt the ICS. This means that the new ICS must be implemented and made operational before the current ICS is deactivated. This has implications for the steps in the Modify use case.

4.3.3.6 Delete ICS UC

Figure 4.13 depicts the UCs involved in deleting an ICS.
This use case describes the opposite of the Provide ICS use case. This is the summary use case described in the most general terms.

Two separate UCs are defined for supporting the deletion of IP connectivity services:

**Delete ICS by the SMS**: The Service Management System requests from the WINMAN system the deletion of an ICS. It is assumed that an ICS is de-activated and released at the end of a session.

**Delete ICS by the WO**: The Operator requests from the WINMAN system the deletion of an ICS. It is assumed that an ICS is de-activated and released at the end of a session.

The following processes (steps in the use cases) are distinguished when deleting the ICS:

**Verify ICS request**
This step describes the WINMAN system responsibilities for the verification and validation of the ICS request to the WINMAN system. The WINMAN system verifies the correctness of all the attributes of the request and determines the validity of the request itself (if applicable) by applying policy rules to this request.

**De-activate ICS**
This step describes the WINMAN system responsibilities for the de-activation of the ICS. This step involves accessing the managed network through the IP and/or WDM EMS(s) for the de-activation of the ICS. After that, the WINMAN system updates the state of the ICS. The physical network resources are still allocated and the connections are still set up but are not available to the end-user. The ICS state becomes **Implemented**.

**Release ICS**
This step describes the WINMAN system responsibilities for the release of the ICS, and it involves the interaction of the WINMAN system with the managed network through the IP and/or WDM NMS(s) for the release of the physical resources in the network. The WINMAN system also releases the logical network resources, which makes them available for other ICSs. The state of the involved network resources becomes Idle (if no other ICS currently makes use of the resource) or Active (if another ICS currently makes use of the resource). The design of the ICS remains filed in the WINMAN system; it may become operational in another time interval. The ICS state becomes **Reserved**.
Remove ICS
This step describes the WINMAN system responsibilities for the removal of the ICS from the list of ICSs. The record of the ICS is removed and the ICS is not existing anymore.

4.3.3.7 Verify CS Request UC

The Verify CS Request is considering the verification of connectivity service (CS) requests from both the WINMAN Operator (WO) and the SMS. This 2-step method (verify semantics-validate against policies) could be practically applied for any type of request verification as long as there is an appropriate algorithm (policy) to handle different sets of attributes.

Verify CS Request should in principal be able to evaluate requests for setting-up end-to-end connections between any pair (set) of endpoints within the WINMAN scope.

4.3.3.8 Create CS UC

Create CS by the SMS: This use case is called from a Summary level use case for the design of a CS between two or more end points. The system has to find the best path that fulfils the requirements proposed and make the necessary reservations of resources on the data storage facilities. No changes on the network are made at this time. The use case is prepared to support the modes of operation automatic and semi-automatic. In the automatic and semi-automatic mode the best path found is returned automatically. In the manual mode the WINMAN Operator is prompted to select one of the paths found.

Create CS by the WO: This use case is used directly from the WINMAN operator to create connectivity between two or more end points either at the IP or the WDM layer (you cannot have interlayer connections). The WINMAN Operator gives a list of the links he wants to use to connect the endpoints and the system has to check if the path fulfils the requirements proposed and make the necessary reservations of resources. No changes on the network are made at this time.

Example of reservation of resources

Let’s suppose that some client requests a connection between router 1 and router 3.
We already have IP connectivity between those two end points, therefore the system will try to reserve resources on the IP links that already exist. A connection would be established between Router 1 and Router 3, passing by Router 2.
Now let’s imagine that the link between Router 2 and Router 3 doesn’t have enough free resources to support the connection. The IP layer has no other route between the desired endpoints so it is unable to establish the service.

Then the system will try to solve the problem by changing the configuration of the optical network.
The optical network may come up with three different solutions:
1 - An optical path is established directly between Router 1 and Router 3 through OXC 1, OXC 2 and OXC 4.
2 - An optical path is established directly between Router 1 and Router 3 through OXC 1, OXC 3 and OXC 4.
An optical path can be established between Router 1 and Router 2 and another one between Router 2 and Router 3. But if the system knows that the IP layer was unable to solve the problem because there were no free resources on the link between Router 2 and Router 3, it knows that the problem can be solved just by reinforcing that link with one more optical path (lambda). There is no need to use another lambda between Router 1 and Router 2.

The two possible optical paths between Router 1 and Router 3 (OXC 1,2,4 and OXC 1,3,4) may seem very similar. But we see that one of the paths uses optical links that already support traffic. If, when calculating the IP link cost, we take into account the occupation of the links we will use, then we can conclude that the best one of this two possibilities is OXC 1,3,4. This way we can distribute the traffic between all optical links.

### 4.3.3.9 Implement CS UC

This use case describes the process of implementing a CS, and can be considered as the counterpart of use case "release CS". The CS under consideration (if in the IP domain) is expected to be in the reserved state and will change to the implemented state, if this use case is successful. The scheduler invokes the "implement CS"-use case when the time for implementing the CS is reached. WINMAN requests the involved EMSs to establish and test the connectivity.

### 4.3.3.10 Activate CS UC

This UC incorporates the functionality for activating an already implemented CS, thus changing its state to Operational. The WINMAN Operator or the SMS does not invoke Activate CS directly, but via the Provision ICS or Modify ICS request. The scheduler triggers indirectly, via the relevant summary level use cases, the Activate CS use case when the time for activating the CS is reached.

### 4.3.3.11 De-Activate CS UC

This UC is the counterpart of the previous UC. The WINMAN Operator, or the SMS does not invoke De-activate CS directly, but via the Delete ICS or Modify ICS. The scheduler triggers indirectly, via the relevant summary level use cases, the De-activate CS use case when the time for deactivating the CS is reached.

### 4.3.3.12 Release CS UC

This use case describes the process of releasing a service. Releasing a service is the opposite of the implementation of a service. The service goes from the implemented to the reserved state. Note that in the reserved state the design of the service is kept. Resources for a future activation of the service are still reserved, but until then they can be used by other services.

### 4.3.3.13 Remove CS UC

This use case is used by the Delete ICS use case, or the WINMAN Operator to eliminate all the reservations of resources for the CS on the storage facilities. The removal of reservations is made only on the WINMAN storage facilities.

### 4.3.3.14 Compute list of IP Routes UC

This use case is constructed to be called from other use cases (Create CS for example). It accepts a list of two end points and calculates all the IP routes (combinations of IP links) that connect one end point to the other. In order to calculate all the possible IP routes between two end points we will use an algorithm similar to the Dijkstra Algorithm (used in OSPF). The major difference is that the Dijkstra Algorithm returns only the less expensive route. In this case we want to find all the possible routes, in order to provide the calling UCs with the means to choose the one that can fulfill their requirements (QoS and bandwidth for example).
This algorithm searches on the IP NEs database for one of the end points (routers) of the connection. That end point will be considered the root. Starting at the root the algorithm will have to analyse all the branches, and discover which combinations of branches can lead to the other end point.

### 4.3.3.15 Compute IP Paths under Constraints UC

This use case is constructed to be called from others use cases (Create CS for example). This use case accepts a list of IP routes and a list of constraints. By comparing all the free resources on the IP links used by all the routes with the list of constraints, it discovers which routes can fulfil the constraints.

An IP route is composed of IP links. All the IP links of each route will be analysed and compared with the parameters asked. The links that can’t fulfil the requirements are marked and all the routes that use them are rejected.

In the end should be returned a list of all IP routes that use only accepted IP links. These IP Routes that conform to the given constraints, are called *IP Paths* in the sequel. A list of rejected IP links should also be returned. The list of rejected links is returned because it can help to decide what readjustments should be made on the network, if necessary.

In this use case we use the definitions:

- **Rejected IP link**: An IP link that can’t fulfil the requirements of the connection.
- **Accepted IP link**: An IP link that can fulfil the requirements of the connection.
- **Rejected IP path**: An IP path made of one or more rejected IP links.
- **Accepted IP path**: An IP path made only of accepted IP links.

At the beginning of the use case all the IP paths are accepted IP paths, and all the IP links are accepted IP links.

### 4.3.3.16 Compute list of Optical Trails

This use case is constructed to be called from others use cases (Create CS for example). This use case accepts a list of two end points and calculates all possible Optical trails that connect one end point to the other. The process is very similar to the one used on the Compute list of IP routes use case.

This algorithm searches on the WDM NEs database for one of the end points of the trail. That end point will be considered as the root. Starting at the root the algorithm will have to analyse all the branches, and discover which combinations of branches can lead to the other end point.

In the end of the computation all the routes must be checked to verify if it is possible to implement them. The process of implementing optical trails is not as flexible as implementing IP paths. We have to consider that not all optical equipment support lambda conversion or can’t even switch all the lambdas that pass through it. Because of that probably not all the calculated Optical trails are possible to implement.

### 4.3.4. Network Inventory Management (NIM)

Figure 4.14 depicts the UCs related to Network Inventory Management.
Figure 4.14: Network Inventory Management UCs

NIM UCs incorporate all the functionality necessary for managing the WINMAN Inventory, which constitutes the configuration of the network managed by WINMAN.

### 4.3.4.1 Register EMS UC

This UC registers a new EMS to the WINMAN system, thus configuring and activating a communication link towards this EMS within the WINMAN Southbound I/F. Therefore it constitutes an “auto-discovery” of the new EMS.

### 4.3.4.2 Un-Register EMS UC

This UC un-registers an EMS to the WINMAN system, thus dropping a communication link towards this EMS within the WINMAN Southbound I/F. Therefore it constitutes a counterpart of the Register EMS UC.
4.3.4.3 Notify Network Change UC

This UC handles notifications regarding changes in network elements, received from the WINMAN Southbound interface, i.e. the EMSs managed by WINMAN. It requires that a communication link towards the EMS is active, for exchange of messages at the WINMAN Southbound I/F.

4.3.4.4 Configure Network UC

The "Configure Network" use case is used when one or more network inventory parameters need to be changed (including addition and/or removal of network elements) manually, i.e. by the WINMAN Operator. This interaction is effected via appropriate GUI representations of the current network configuration map. Moreover, Configure Network may be used to effect changes to the network inventory that are automatically reported to WINMAN (e.g. through Register EMS). The WINMAN solution allows modifying the topology without service outages. This means that the system should verify that changes do not affect service by means of policy-based validation.

The following modifications of a network have been identified:
- Changes in the IP domain.
- Changes in the WDM domain.
Both types of changes should be handled by the Configure Network UC.

4.3.4.5 Add IP node UC

This use case is used by the Configure Network use case when new IP node-s are going to be added to the WINMAN system.

4.3.4.6 Delete IP node UC

This use case is used by the Configure Network use case when existing IP node-s are going to be deleted from the WINMAN system

4.3.4.7 Add WDM node UC

This UC creates a WDM node and adds it to the WINMAN Inventory.

4.3.4.8 Delete WDM node UC

This UC removes a WDM node from the WINMAN Inventory.

4.3.4.9 Add WDM link UC

Creates a WDM link entity in the WINMAN Inventory.

4.3.4.10 Delete WDM link UC

Deletes a WDM link entity in the WINMAN Inventory.

4.3.4.11 Modify WDM link UC

Modifies the characteristics of a WDM link in the WINMAN Inventory.

4.3.4.12 Add EMS UC

This UC modifies the network configuration (updates WINMAN Inventory) by adding new EMS information, through an established communication link at the WINMAN Southbound I/F.
4.3.4.13 Delete EMS UC

This UC modifies the network configuration (updates WINMAN Inventory) by deleting EMS information, thus acting as counterpart of the Add EMS UC.

4.3.5. Network Maintenance & Restoration (NM&R)

Figure 4.15 depicts the NM&R UCs considered by WINMAN:

![Diagram of Network Maintenance & Restoration UCs](image)

**Figure 4.15:** Network Maintenance & Restoration UCs

NM&R UCs incorporate the WINMAN functionality for Fault Management. This refers to operations that are done and reflected in the alarms. These are operations on a single alarm object. The operator acknowledges an alarm or clears it manually. The EMS or some internal server in WINMAN can issue a new alarm or an event that the state of some network element has changed.
This section additionally depicts problem analysis and recovery actions that are also relevant to NIM operations. The system can recover the network automatically or by means of an operator request. An operator can restore the system to the state it was before the recovery.

4.3.5.1 Recover Network UC

This UC is used for re-routing all connections by-passing faulty parts of the network.

4.3.5.2 Revert Network UC

Returns to the original routing, as was before the Recover Network UC invocation.

4.3.5.3 Open Alarm UC

This UC handles alarms issued by an EMS, or by an internal WINMAN application like the Performance Management, or the topology manager when the NE status is changed.

4.3.5.4 Clear Alarm UC

Cleans an alarm and removes it from the list of active alarms.

4.3.5.5 Provide Alarm Reports UC

Creates reports on the different attributes of alarms.

4.3.5.6 Identify Affected Connections UC

This UC finds all active connections influenced by a given alarm.

4.3.5.7 Identify Root Cause UC

This UC identifies the root causes in an alarm list.

4.3.5.8 Synchronize Alarms UC

This UC queries the EMSs for their open alarms for re-synchronisation of the active alarms in the WINMAN system.

4.3.5.9 Acknowledge Alarm UC

Moves an alarm to an 'acknowledged' status.

---

1 The terms recovery, restoration and protection are used in the literature to denote actions taken in order to heal network connection from failures in the network (elements, fibres). Within this document the following definitions are applied:

**Recovery**: is a general term, which is used to refer to the mechanisms that can be used to recover a network from failures. Two main categories or recovery mechanisms can be identified: protection and restoration

**Protection**: Protection reserves in advance backup paths, which can be used in the case of a failure. Protection is faster and simpler than restoration. However resources are waste to protect the working paths. Protection is usually the first line recovery

**Restoration**: Restoration dynamically searches for new routes in order to re-route the affected resources. Restoration is slower and complex compared to protection. Restoration is the second line recovery mechanism. Finally the term recovery policies will be used to refer to the rules and recovery schemas (protection, restoration) that will be used to recover the different classes of CS.
4.3.6. Network Data Management (NDM)

Figure 4.16 depicts the NDM UCs considered by WINMAN:

NDM UCs incorporate the functionality for Performance Management (PM) of the system managed by WINMAN. These are the use cases providing means to collect and analyse performance data. The operator can basically produce reports and configure the way WINMAN collects performance information. The configuration can be on a counter-by-counter manner or by an ICS. WINMAN intends to support a set of pre-configured data collection policies to be used by operators that don’t want to get into the glory details.

4.3.6.1 Process Performance Measurements UC

Collects the counters according to a defined policy. If the thresholds are crossed, an alarm is triggered.

4.3.6.2 Provide PM Report UC

Produces reports with an analysis of quality of the supported CS.

4.3.6.3 Calculate Aggregations & Statistics UC

Calculates aggregation of the counters for different periods of time; moreover calculates computed counters and statistics (compared to average and standard deviations) according to the customer’s definitions. Calculates QoS counters for ICSs

If some thresholds were crossed by the calculated values, alarms are generated and sent to the FM.
5. System Analysis and Architecture

5.1. Introduction

This chapter describes the management architecture the WINMAN solution will adopt for the integrated management of IP over WDM networks. It determines the boundaries of the system, and proposes both the high level and the internal system architecture designed to meet the requirements identified for the system in the areas of configuration management, fault management and performance management. Only the informational and logical architectures are addressed in this document. Standard UML diagrams are used for the description of the static (i.e. domain model) and dynamic (i.e. scenarios) functionality of WINMAN. The overall WINMAN aim is to offer an integrated network management solution, the WINMAN solution, which is capable to provide end-to-end IP connectivity services derived from Service Level Agreements (SLAs). It will define and specify an open, distributed, and scalable management architecture for IP connectivity services on hybrid transport networks (ATM, SDH and WDM), with its focus on the management of IP directly over WDM networks.

A two-layer management architecture has been conceived for the system that is intended for implementation with the purpose of demonstrating the feasibility of the WINMAN solution. The first layer contains the technology dependent managers, that is one for the IP network and another manager for the WDM optical network. These two management systems will interact with existing element management systems or even subnetwork management systems properly adapted through what is called the southbound WINMAN interface. Other network technologies, such as SDH or ATM, although not addressed explicitly in this document, have also been taken into account during the definition of the architecture to make it generic enough to allow its simple extensibility to any technology.

On top of this layer there is a second one, the so-called INMS, whose purpose is the integration of the above mentioned management systems. The idea behind the concept of integration is to make the best possible use of the underlying transport technologies to efficiently offer the intended end-to-end transport services. At this layer, a technology agnostic management is considered. This means that any network management system, no matter what technology it manages, can be integrated in the architecture and managed by the INMS provided that it supports the specified interface at that point.

While defining WINMAN architecture, the following goals have been considered:

- Open architecture, with published APIs to the relevant actors, giving third parties the possibility to implement their own management application. Wherever possible, interfaces specified by the relevant standardisation fora (ITU, TeleManagement Forum, IETF, OIF, ODSI) should be adopted
- Flexible architecture, i.e. being able of evolving to and accommodating new systems. In particular, seamless accommodation of SDH and ATM network management systems should be enabled
- Modular architecture, i.e. giving capability to add or remove management blocks in a plug and play fashion without requiring major changes to existing components. This requires a clear separation of responsibilities between blocks
- Scalable architecture, i.e. it should be able to cope with network evolution scenarios towards IP over WDM and growth of the network in terms of number of network elements and type of network elements
- Distributed architecture, i.e. operating on a platform in which distribution of the system is transparent to the components of which that system is composed
- Platform and persistency independence of the architecture
- Generality of the solution, i.e. it should be able to support the different kinds of business models and operational processes that are likely to be found among potential users of WINMAN
- Minimisation of the number of interfaces between blocks
5.2. High level architecture

WINMAN will follow a 3-tier architecture, namely a presentation layer (web-based interface), a business/logical layer and a persistence layer, as it is shown in Figure 5.1.

![Figure 5.1: 3-tier architecture view of the WINMAN solution](image)

As it can be seen in Figure 5.2, the high level architecture of the WINMAN comprises three different network management systems, that is, the IP-NMS and the WDM-NMS at the technology dependent layer and the INMS above them. The user interface for the complete solution is also considered in the Figure 5.2.

![Figure 5.2: Overview of WINMAN high level architecture](image)

Only the simplest option is depicted in Figure 5.2, with a single system per management layer. Note that the architecture proposed hereafter can support multiple SMS over the Northbound Interfaces, multiple NMSs at the technology specific layer, and multiple EMSs under the Southbound Interfaces.
5.2.1. Systems Description

5.2.1.1 Integrated Network Management System (INMS)

The Integrated Network Management System is devoted to carry out the integration of the management functions of the technology specific management systems. The INMS is technology agnostic in the sense that its representation of the network is not tied to any specific technology. Through a properly abstracted information model, the INMS is able to use the domain NMSs for the provisioning and monitoring of end to end connections without any knowledge about network details except for the layered structure. This means that the INMS does not make any assumptions about the type of network managed by the domain NMSs; it only requires them to support a common interface for the establishment and releasing of connections and for the notification of fault or performance related events. That common interface is devised for connection oriented networks following the main focus of WINMAN. The adaptation of the connectionless nature of IP to that interface is a subject of study under the project.

The configuration management functions enable single point access to provisioning tasks and to end-to-end views of connections and their underlying infrastructure and facilities, independent of domain. The main functions that the INMS performs in the area of configuration management are:

- Serving of requests from the Service Management Systems for the provisioning, modification and deletion of IP connectivity services, supporting different classes of service like real-time or VPN services.
- Configuration of the end-to-end connections and services across the different technology domains, designing the end-to-end route as an ordered set of subnetwork connections.
- Presentation of an end-to-end view of the services to the operator showing the logical hierarchy of all network resources (nodes and links) constituting the end-to-end connection.
- Maintenance of an inventory of all the network resources relevant at this layer with their status and their hierarchical relationship. This system only has knowledge about subnetworks at each of the domains and about points of interconnection at the edge of those subnetworks.
- Co-ordination of the restoration process when services become faulty or fall under acceptable quality of service levels.
- Providing of updates to the fault and performance management functions reflecting the changes in the network configuration and new services.

In the area of fault management, the functions carried out by the INMS are:

- Inter-domain alarm correlation via user defined correlation rules
- Notification to the service layer systems of faulty services
- Triggering of the restoration process

In the area of performance management, the INMS is responsible of the following functions:

- Collection of performance data from the underlying management systems
- Analysis of data and assessment of the quality of service delivered
- Generation of service performance degradation alarms
- Aggregation of information for the purposes of sending to the SMSs SLA related values

5.2.1.2 WDM-NMS

The WDM Network Management System carries out the management of the WDM transport network layer and offers a technology independent management view of the WDM network. It relies on one or several WDM-EMSs that provide a first level of abstraction of the network and deal with vendor-specific aspects of network elements. The WDM-NMS incorporates full functionality in the areas of configuration management, fault management and performance management so that it can be deployed as a stand-alone management system capable of managing a WDM network by its own.
The configuration management functionality the WDM-NMS supports is:

- Provision of optical paths, including protection
- Wavelength routing, including the capability of wavelength transition; implementation of the designed optical path in the network elements
- Discovery of network resources in terms of network elements, ports per network element, fibre connections between nodes, wavelengths per fibre, transport capacity per wavelength and protection mechanisms. The change notifications are forwarded to the INMS only if the modification can affect the view of the network at this higher level
- Maintenance of an inventory with the WDM resources and connections. Also maintains information about policies in order to apply the rules in the routing process
- Co-ordination of the actions needed for the restoration of paths at the optical network

In the area of fault management, the WDM-NMS undertakes the following functions:

- Collection and distribution of alarms affecting the WDM network
- Identification of root cause of alarms, making use of user defined correlation rules
- Notification of faulty optical paths to the INMS
- Triggering of the optical network restoration process

In the area of performance management the WDM-NMS is in charge of:

- Gathering of data about the performance levels in the WDM network, in terms of signal to noise ratio, estimated BER, power, etc.
- Analysis and aggregation of the collected performance data
- Generation of performance degradation alarms
- Sending of relevant performance information to the INMS

5.2.1.3 IP-NMS

The IP policy-based network management system carries out the management of the IP network layer and offers a technology independent management view of the IP network, adapted to the connection oriented approach on which WINMAN focuses. The IP-NMS relies on one or several IP-EMS for the management of those aspects of the network elements not relevant at the network management layer. Like in the WDM case, the IP-NMS is able to work as an stand-alone system for the end-to-end management of the IP network.

The configuration management functionality is:

- Design of connections inside its own subnetworks
- Establishment and maintenance of end-to-end QoS connectivity services, including the setting up of MPLS LSPs
- Discovery of network topology. Just as in WDM-NMS, the change notifications received from IP-EMS are transmitted to the INMS level only if the modification can affect the view of the network at this higher level
- Maintenance of an inventory with the IP resources and connections
- Co-ordination of the restoration process when it only affects the IP network

With respect to fault management, the functionality carried out by the IP-NMS is:

- Reception of alarms about faulty nodes and links
- Correlation of IP alarms to determine the faulty entity
- Sending of the correlated alarms to the INMS

In the area of performance management, the IP-NMS performs the following functions:

- Gathering of performance information about the network, in the terms of link load, network congestion, effective throughput, etc
- Analysis of performance data to identify hotspots and bottlenecks in the network
- Generation of performance related alarms
- Sending to the INMS of appropriate information to allow the assessment of the delivered quality of service
5.2.1.4 GUI

WINMAN has a single graphical user interface that interacts with the three previously described systems. It supports the access to information and functions of the INMS, the WDM-NMS and the IP-NMS.

The GUI should be as thin as possible in order to be WEB enabled. It will have a middle tier, the View Manager, that carries the GUI logic while the client itself will be responsible for the Graphical User Interface alone.

5.2.1.5 Interfaces

5.2.1.5.1 Northbound interface

It communicates the Service Management Layer and the Integrated Network Management System. It will be based on the CaSMIM interface specified by the TeleManagement Forum. The information exchanged through this interface is related to:

- Requests for the provisioning of an IP connectivity service with Network Level Agreement (NLA; supporting the requested SLA) parameters and scheduling data. The provisioning requests shall support enough information to fully characterise the type of service.
- Requests to modify service. The identified modifications are: change of the NLAs, such as, but not limited to, bandwidth, reliability (availability or protection level), delay, jitter or packet loss; change of scheduling (initial or final time)
- Requests for the deletion of IP connectivity services
- Change of IP connectivity service provisioning status notifications (creation, implementation, activation, modification, de-activation, release and removal) sent from the INMS to the service layer systems
- Alarm/event notifications about faults affecting the service sent from the INMS to the service layer systems
- Alarm status of IP connectivity services sent from the INMS upwards
- Quality of service reports requested by the service layer and provided by the INMS
- Clock synchronisation messages

The most important interactions that this interface supports are illustrated in the system scenarios that form part of section 5.4.

5.2.1.5.2 GUI to INMS interface

The following information is exchanged across this interface:

- Requests for the provisioning of an IP connectivity services constrained by Network Level Agreements sent from the GUI, including manual configuration information
- Requests to modify service parameters (NLA, scheduling, ...) sent from the GUI
- Requests for the deletion of IP connectivity services sent from the GUI
- Change of IP connectivity service provisioning status notifications (creation, implementation, activation, modification, de-activation, release and removal) sent from the INMS to the GUI
- Notification of new entities (subnetworks, termination points, etc.) added in the network generated by the INMS
- Information about subnetworks and subnetwork connections managed by the INMS
- Policy rules created by the operator and received by the INMS
- Notification of alarms and information about faults sent by the INMS to the GUI system
- Quality of service information about the IP connectivity service delivered in the network
The most important interactions that this interface supports are illustrated in the system scenarios that form part of section 5.4.

### 5.2.1.5.3 GUI to WDM-NMS interface

The following information is exchanged across this interface:

- Requests for the setting up, rerouting and tearing down of optical subnetwork connections issued from the GUI
- Notifications to the GUI about the degree of success of the requests for the setting up, rerouting and tearing down of optical subnetwork connections
- Requests for the addition/deletion of specific WDM nodes or links issued by the GUI
- Notification of new entities (nodes, links, ports, subnetworks, etc) incorporated to the network generated by the WDM-NMS
- Information about all network entities (from inventory information to end-to-end connections) managed by the WDM-NMS
- Performance related information about the WDM network sent from the WDM-NMS to the GUI

The most important interactions that this interface supports are illustrated in the system scenarios that form part of section 5.4.

### 5.2.1.5.4 GUI to IP–NMS interface

The following information is exchanged across this interface:

- Requests for the setting up, rerouting and tearing down of subnetwork connections issued from the GUI
- Notifications to the GUI about the degree of success of the requests for the setting up, rerouting and tearing down of the connections
- Requests for the addition/deletion of specific IP nodes or links issued by the GUI
- Notification of new entities (nodes, links, ports, subnetworks, etc) incorporated to the network generated by the INMS
- Information about all network entities (from inventory information to end-to-end connections) managed by the INMS
- Performance related information about the WDM network sent from the WDM-NMS to the GUI

The most important interactions that this interface supports are illustrated in the system scenarios that form part of section 5.4.

### 5.2.1.5.5 Interface between INMS and WDM-NMS

This interface will be based on a technology neutral one that represents the different network technologies by means of generic concepts. WDM specific features are hidden at this interface. The following information is exchanged between the INMS and the WDM-NMS:

- Connection requests (create/remove, implement/release, activate/deactivate and modify) from the INMS to the WDM-NMS. These requests include performance goals and level of reliability desired
- Responses to connection requests conveying the degree of success
- Notifications from the WDM-NMS about changes of network resources that can affect the view of the managed network that the INMS has
- Alarm/event notifications about faults affecting the established connections
- Performance data (BER, availability, etc.) about optical connections generated by the WDM-NMS
- Policies affecting the WDM network management
- Clock synchronisation information
The most important interactions that this interface supports are illustrated in the system scenarios that form part of section 5.4.

5.2.1.5.6 Interface between INMS and IP-NMS

This interface will be based on a technology neutral one that represents the different network technologies by means of generic concepts. It will be based on the CaSMIM interface specified by the TeleManagement Forum. IP specific features are hidden at this interface. The IP network is modelled as a connection oriented network to allow a common treatment of all technologies at the INMS level.

The following information is exchanged between the INMS and the IP-NMS:
- Connection requests (create/remove, implement/release, activate/deactivate and modify) from the INMS to the IP-NMS. These requests include NLA information
- Responses to connection requests conveying the degree of success
- Notifications from the IP-NMS about changes of network resources that can affect the view of the managed network that the INMS has
- Alarm/event notifications about faults affecting the established connections
- Performance data (effective throughput, delay, etc.) about connections provided by the IP-NMS
- Policies affecting the IP network management
- Clock synchronisation information

The most important interactions that this interface supports are illustrated in the system scenarios that form part of section 5.4.

5.2.1.5.7 WDM-NMS Southbound interface

It communicates the WDM-NMS and the WDM element management layer. It will be based on the MTMN interface specified by the TeleManagement Forum. The information exchanged across this interface is:
- Connection and disconnection requests issued by the WDM-NMS
- Up-to-date information about the network resources provided by the element management layer.
- Alarms affecting the WDM network
- Filtering rules placed by the WDM-NMS at the element management layer
- Performance measures (signal power, signal to noise ratio, …) made in the WDM network

The most important interactions that this interface supports are illustrated in the system scenarios that form part of section 5.4.

5.2.1.5.8 IP-NMS Southbound interface

It communicates the IP network management system and the IP element management layer. It will be based on the MTMN interface specified by the TeleManagement Forum. The information exchanged across this interface is:
- Connection and disconnection requests issued by the IP-NMS
- Up-to-date information about the network resources provided by the element management layer.
- Alarms affecting the IP nodes and links
- Filtering rules placed by the IP-NMS at the element management layer
- Performance measures (traffic load, effective throughput, …) made in the IP network
The most important interactions that this interface supports are illustrated in the system scenarios that form part of section 5.4.

5.3. Domain Model and Network Views

The purpose of this chapter is to describe the domain model entities relevant to the WINMAN management system interfaces and internal structure and to present the different views in these interfaces. The domain model will be the basic source for the external WINMAN interfaces information models, as well as for the internal representation and adaptation among the different layers (IP, MPLS, WDM). In addition, the views in the corresponding external interfaces or internally in the system and to the WINMAN operator are described based on the domain model entities.

5.3.1. Description

WINMAN will focus on a connection-oriented IP domain model, corresponding to MPLS technology. The use of IP connectionless technology in co-operation with MPLS is left for further study. In any case and based on the G.805 layered architecture we identify 2 layers being MPLS and WDM, while in case we will use connectionless IP we will have 3 layers. In the latter case, MPLS will be serving as a server layer for IP. In the following domain model, we have considered only MPLS entities and for clarity reasons, we replaced everything related to IP with MPLS, although the terms used are not successful in all cases. For this reason, the term IP or more important the term IP Connectivity Service (ICS) mentioned elsewhere in this deliverable, is described as MPLS or MPLS Connectivity Service or MPLS CS.

According to the above hypothesis, we suppose that the MPLS domain starts at the Customer Edge IP equipment and thus when traffic enters the WINMAN responsibility (Provider Edge IP equipment), we assume that it is already marked as MPLS. However, the exact boundaries of the MPLS domain, will be further investigated in the lifetime of the project and provided that there is a justified need for change, the document will be updated.

5.3.1.1 Common Entities

Connectivity Service (CS)
A CS is the basic service delivered by the MPLS or WDM network and managed by WINMAN. The service is delivered either by the MPLS network meeting some QoS goals resembling a connection-oriented circuit or by the WDM network.

Element Management System (EMS)
The Element Management System represents the abstraction of the sub-network(s) managed by the EMS (EMS management domain) and the element management system itself.

- Termination Point (TP)
  A termination point (TP) shall be a logical abstraction of an endpoint (actual or potential) of:
  - A topological (physical) link, or
  - A subnetwork connection
  A TP is contained within a managed element

Physical Termination Point (PTP)
A termination point that is an actual or potential endpoint of a topological (physical) link shall be abstracted as a physical termination point (PTP). Essentially, this is a representation of a physical port.

Connection Termination Point (CTP)
A Connection Termination Point is an actual or potential end point of a subnetwork connection.

Topological Link
A Topological Link is a physical link between two PTPs. A Topological Link has a name and references to the two PTPs. A Topological Link reported by an EMS to NMS will be between two managed elements (MEs) managed by the same EMS, and depending on the capabilities of the EMS and the MEs, a link may or may not be autodiscovered by the EMS.

Link connection
A link connection represents the transparent capacity of transfer information characterised by a given signal identification between two fixed points.

Subnetwork Connection (SNC)
A Subnetwork Connection relates Connection Termination Points. A Network Connection provides a transparent end-to-end connection through or within a subnetwork. The Subnetwork Connection may be created/deleted/modified by the INMS or Domain specific NMS and is implemented by the EMS. The Subnetwork Connection is contained in a Subnetwork.

Route
The route of a subnetwork connection shall be represented as an ordered series of CTPs names through which the subnetwork connection traverses, including the working and protect route. The protecting route is optional.

Managed Element (ME)
A Managed Element is an abstract class used to represent Network Elements visible across the interfaces.

Policies
Set of rules that define the system behaviour

5.3.1.2 MPLS Entities

MPLS Connectivity Service Connection (MPLS-CS)
An MPLS CS is the basic service delivered by the MPLS over WDM network and managed by WINMAN. The service is delivered over the MPLS network meeting some QoS goals resembling a connection-oriented circuit.

IP Element Management System (IP-EMS)
The IP/MPLS Element Management System represents the abstraction of the sub-network(s) managed by the IP-EMS (EMS management domain) and the element management system itself. This system manages both IP and MPLS protocols.

MPLS Connection Termination Point (CTP)
A Connection Termination Point is an actual or potential end point of a MPLS sub-network connection. At the MPLS layer it corresponds to a specific label and port.

MPLS Subnetwork
An MPLS Subnetwork is an abstraction provided by the IP-EMS to the IP-NMS, and by the IP-NMS to the INMS. The Subnetwork is the unit of work of an EMS system. The main services provided within a Subnetwork are the creation and edition of MPLS cross-connect tables. Managed Elements belong to Subnetworks. Subnetworks in the MPLS domain of WINMAN shall represent groups of co-located routers (Label Switch Routers), connected via LAN technologies, or single routers. Subnetworks (as represented over an interface) are created/deleted/modified by the system performing the role of agent in that interface.

MPLS Topological Link
An MPLS Topological Link represents capacity between two MPLS subnetworks. and is a physical link between two MPLS PTPs.

MPLS Link Connection (MPLS-LC)
A MPLS Link Connection is terminated by two and only two extremities, the CTPs. It is a “transport entity” that transfers information between “ports” across a link.
MPLS Cross-Connect table
A MPLS Cross-Connect table is a set of associations between {input label, ingress port} and {output label, egress port}

MPLS Subnetwork Connection (SNC)
A MPLS Subnetwork Connection relates MPLS Connection Termination Points. A Network Connection provides a transparent end-to-end connection through or within a subnetwork, equivalent to a portion of one MPLS path. The Subnetwork Connection may be created/deleted/modified by the INMS or WDM-NMS and is implemented by the EMS. The Subnetwork Connection is contained in a Subnetwork.

MPLS Trail or Label Switched Path
A MPLS trail represents a transport entity, which is responsible for the transfer of information between two MPLS trail-termination-points. A trail is supported by a set of SNCs. Due to the recursive nature of the LSPs, a MPLS Trail can support a MPLS Topological Link or a MPLS Link Connection in the client layer. In addition the MPLS trail can support IP connectionless service if this is the case for WINMAN.

MPLS Trail Termination Point
A Trail Termination Point represents an extremity of a trail. It is the combination of part of the adaptation function, the access point and the trail termination function, as defined in recommendation G.805.

MPLS Physical Termination Point
A MPLS PTP that is an actual or potential endpoint of a topological (physical) link shall be abstracted as a physical termination point (PTP). Essentially, this is a representation of a physical port, belonging to a Label Switch Router (LSR).

MPLS Managed Element (MPLS-ME)
An MPLS Managed Element is an abstract class used to represent Label Switched Routers visible across the interfaces. The complete list of types of MPLS managed elements considered in WINMAN is for further study.

MPLS policies
MPLS policies are sets of rules that specify the treatment to be applied to MPLS traffic and provisioning requests.

5.3.1.3 WDM Entities

Optical Connectivity Service Connection (OCS)
An OCS is the basic service delivered by the WDM network and managed by WINMAN. The service is delivered over the WDM network providing OCh SNCs or trails.

WDM Element Management System (WDM-EMS)
The WDM Element Management System represents the abstraction of the subnetwork(s) managed by the WDM-EMS (EMS management domain) and the element management system itself.

Optical Connection Termination Point (CTP)
An Optical Connection Termination Point is an actual or potential end point of an optical Channel subnetwork connection. In an optical network it is a representation of a specific wavelength of an optical port.

Optical Subnetwork
An Optical Subnetwork is an abstraction provided by the EMS to the WDM-NMS, and by the WDM-NMS to the INMS. The Subnetwork is the unit of work of an EMS system. The main services provided within a Subnetwork are the set-up and tear-down of Subnetwork Connections. Managed Elements belong to Subnetworks. Initially, subnetworks shall represent relatively simple collections of Managed Elements, e.g., only single rings, linear segments, or stand-alone managed elements. Subnetworks (as represented over an interface) are created/deleted/modified by the system performing the role of agent in that interface.
Optical Channel Subnetwork Connection (OChSNC)
An Optical Channel Subnetwork Connection relates Optical Connection Termination Points. A Network Connection provides a transparent end-to-end connection through or within a subnetwork. The Subnetwork Connection may be created/deleted/modified by the INMS or WDM-NMS and is implemented by the EMS. The Subnetwork Connection is contained in a Subnetwork.

Optical Channel Trail
An optical channel trail represents a transport entity which is responsible for the transfer of information between two optical channel trail termination points. A trail is supported by a set of SNC.

Optical Channel Trail Termination Point (OChTTP)
A Trail Termination Point represents an extremity of a trail. It is the combination of part of the adaptation function, the access point and the trail termination function, as defined in recommendation G.805.

Optical Topological Link
An Optical Topological Link represents capacity between two optical subnetworks and is a physical link between two WDM PTPs.

Optical Link Connection (OLC)
An Optical Link Connection is terminated by two and only two extremities, the Optical CTPs. It is a “transport entity” that transfers information between “ports” across a link.

WDM Physical Termination Point (WDM PTP)
A WDM PTP that is an actual or potential endpoint of a topological (physical) link shall be abstracted as a physical termination point (PTP). Essentially, this is a representation of a physical port, belonging to an Optical Managed Element

Optical Access group
An optical access group is a set of co-located trail termination points that are bound to one subnetwork or link.

Optical Managed Element (OME)
An Optical Managed Element is an abstract class used to represent Network Elements visible across the interfaces. The complete list of types of optical managed elements considered in WINMAN is for further study, but the following should at least be included:
- OADM
- OXC
- Optical Amplifier
- Terminal Multiplexer

Optical policies
Optical policies are sets of rules that indicate constraints or priorities that need to be applied during the processing of provisioning requests.

5.3.2. Class diagrams
The following diagrams are presented below:
The inheritance hierarchy, showing the inheritance relationships of the domain model entities described above
A transport technology independent generic domain model, that will be applied for the two technologies under consideration, i.e. MPLS and WDM providing correspondingly the
MPLS domain model and the
WDM domain model
Figure 5.3: Inheritance hierarchy

Figure 5.4: Generic domain model
Deliverable (D2.1) - Integrated network management requirements and architecture for IP transport services

Figure 5.5: MPLS Domain Model

Figure 5.6: WDM Domain Model
5.3.3. **Internal and External Network Views**

As already mentioned, the proposed WINMAN solution focuses in the Network Management Layer of the TMN pyramid. The interfaces for interactions with the outer world is the northbound interface towards the Third Parties Service Management Functional Systems, such as a VPN or VoIP, MoIP functionality systems and southbound to the Element Management Functional Systems. Within our scope are the IP and WDM technologies, since ATM and SDH functional systems will not be implemented. A presentation functional system or Workstation function in TMN terms providing a workstation reference point is provided to the operator, as a means of controlling and monitoring the system and underlying network. Below, the views of the network providing different level of detail according to the positioning in the TMN pyramid and depending on the interaction with the corresponding entity will be given. Note that not only the network views on external interfaces are given, but also the internal WINMAN system views provided for the adaptation among multiple layers and for better visualisation for the WINMAN operator.

The views are presented for the following internal or external interfaces:
- INMS to SMS
- IP NMS to INMS and WDM NMS to INMS
- IP EMS to IP NMS
- WDM EMS to WDM NMS

Also different multiple and/or integrated views containing multiple layers will be presented.

Note, that the terms and entities of the above-defined domain model will be used as much as possible, since not all entities can be used and moreover visualised in the figures.

5.3.3.1 **INMS to SMS Network View**

The view in this external interface should be as abstract as possible. In this sense, the INMS need to show to SMS only the edge Physical Termination Points, which are the ones interconnecting the Provider edge equipment with the Customer Edge equipment. Such PTPs are always MPLS PTPs and not WDM PTPs. In the following figure there are 3 MPLS Connectivity Service connections (MPLS CS). The CTP are presented as small (red) circles inside PTPs, which are depicted as bigger (grey) ellipses.

![Figure 5.7: INMS to SMS network view](image)

5.3.3.2 **IP NMS to INMS Network View**

The IP NMS will provide to the INMS the view that is depicted in the following figure. The PTPs shown are the edge MPLS-PTPs, which were also shown towards the SMS and the MPLS over WDM PTPs, which interconnect to the WDM network. Between the corresponding CTPs exist the MPLS Subnetwork connections (MPLS-SNCs).

Note that the MPLS CTP interconnecting to the WDM network are marked with a single apostrophe (’), while the corresponding CTPs in the WDM will be later marked by a double apostrophe ("-quotation mark).
5.3.3.3 WDM NMS to INMS Network View

The WDM NMS will provide to the INMS the view that is depicted in the following figure. The PTPs shown are the WDM PTPs interconnecting to the MPLS equipment, which were also shown in the IP NMS view and the pure WDM PTPs, which interconnect the WDM network. Between the corresponding optical CTPs exist the Optical Channel Subnetwork connections (Och-SNCs). Note, that for simplicity reasons, the WDM PTP are not labelled, although depicted in the figure.

5.3.3.4 IP EMS to IP NMS Network View

Assuming that IP EMS do exist in our network paradigm, the view that the IP (MPLS-capable) EMS will provide towards the IP (MPLS-capable) NMS is depicted in the following figure. In other words, we assume that there are 2 IP EMSs, each one responsible for its own MPLS sub-network. The line between the 2 MPLS sub-networks is an MPLS topological link, which is supported by an Och-SNC, as will be shown in another view. In general, 2 MPLS sub-networks can also be connected via a pure MPLS topological link, which is a fibre connecting the 2 ports of two neighbour Label Switch Routers, but this in not the case in this view. This is also indicated by the apostrophes in each CTP, indicating that the MPLS topological link is supported by the
WDM layer. Note, that the end-to-end MPLS-CS-2 is supported by the corresponding MPLS-SCNs (MPLS-SNC-3 and 4) plus a MPLS Link Connection supported by the MPLS Topological Link.

![Diagram](image)

**Figure 5.10: IP EMS to IP NMS network view**

### 5.3.3.5 WDM EMS to WDM NMS Network View

Similarly, the view that the WDM EMS will provide towards the WDM NMS is depicted in the following figure, assuming that there are 2 WDM EMS, each one responsible for its own WDM sub-network. The line between the optical CTP-C2 and optical CTP-C1 is an optical topological link, therefore a fibre connecting the 2 ports of the optical network elements. Note, that the Och SNC-3 is supported by the corresponding Och-SNCs (Och-SNC-3a and 3b) plus an Optical Link Connection supported by the Optical Topological Link (fibre interconnecting 2 WDM elements).

![Diagram](image)

**Figure 5.11: IP EMS to IP NMS network view**

### 5.3.3.6 Multiple Views in a single picture

The following figure depicts the multiple views between SMS-INMS, WDM NMS-INMS, IP NMS-INMS and IP EMS-IP NMS, WDM EMS-WDM NMS in a single picture. The following concatenations of MPLS CS are depicted reaching up to the NMS level (not EMS):

- MPLS CS-1=MPLS-SNC-1+LC+Och-SNC-1+LC+MPLS-SNC-2
- MPLS CS-2=MPLS-SNC-3+LC+Och-SNC-3+LC+MPLS-SNC-4
- MPLS CS-3=MPLS-SNC-5 (pure MPLS)
Analysing for example the MPLS-CS-1, it is composed by:
- the MPLS-SNC1 between MPLS CTP-A and MPLS CTP-A’,
- the optical link connection between MPLS CTP-A’ and Optical CTP-A”,
- the OCh-SNC1 between Optical CTP-A” and Optical CTP-B”,
- the optical link connection between Optical-CTP-B” and MPLS-CTP-B’ and finally
- the MPLS-SNC2 between the MPLS-CTP-B’ and MPLS-CTP-B.

5.3.3.7 Integrated or INMS internal View

The integrated internal view is depicted in the following figure, where the MPLS sub-network ellipse includes not only the pure MPLS PTPs and MPLS-SNCs, but also the MPLS to WDM PTPs. The MPLS CSs are shown as concatenated MPLS and Optical SNCs, together with the corresponding optical link connections. Such a view can be considered as an internal view, necessary for the adaptation between the 2 layers and might be provided towards the WINMAN operator. For clarity, also the INMS to SMS view is depicted, to show the interdependencies between the two ellipses.
5.3.3.8 IP EMS to Element layer View

In order to interconnect the above views with a real example and test-bed configuration the following view has been added. It shows the Element Management Layer of the MPLS and Optical Sub-networks together with the corresponding technology equipment. The above are shown in the following figure:
The network that is being displayed is composed by the 2 Optical Sub-networks and 2 MPLS sub-networks. The first optical sub-network is a ring of Optical Add-Drop-Multiplexers and the second one is an Optical Cross-Connect. In the first case, OChSNC are a concatenation of optical link connections (inside fibres) and cross-connect inside the OADM, while in the second case the OChSNC is degenerated only in one cross-connect inside the OXC. On the other hand the MPLS network is composed by multiple Label Switch Routers, either playing the role of Ingress, Core or Egress LSRs.

Note that we have selected 3 MPLS CS connections, 2 of them spanning both domains, while 1 of them is a pure MPLS one (between MPLS-CTP-A and MPLS-CTP-D). We suppose that the MPLS equipment on the left is managed by one NMS, and that the two MPLS LSRs on the far down-left corner are using the WDM ring to interconnect to the rest of the IP network.

5.3.3.9 Overall Network Model

An example of the previous defined domain model entities providing the views analysed above is depicted in the following figure. In addition, some adaptation entities such as the trail and Trail Termination Points (TTPs) appear between the 2 layers, showing the corresponding adaptation functionality. In the following figures not all entities appear in an ellipsis to streamline them in a way that are easily understandable. For example, PTPs and CTPs, do not appear always in the same figure. In addition, Topological links and link connections appear separately.
5.4. System scenarios

This section contains the most relevant scenarios of utilisation of WINMAN. It is intended to show the responsibilities of each system and the main interactions among them, and not to make an exhaustive specification of all possible cases. The scenarios are categorised in five major areas:

- Provisioning and configuration scenarios
- Policies scenarios
- Network inventory scenarios
- Network maintenance and restoration scenarios
- Network data management scenarios

5.4.1. Provisioning and configuration scenarios

5.4.1.1 Provision of ICS involving creation of new lightpath

5.4.1.1.1 Context

This scenarios corresponds with the requests received from the Service Management System for the provisioning of an IP connectivity service with Network Level Agreement (NLA; supporting the requested SLA) parameters and scheduling data. Includes the following steps:
5.4.1.1.2 Sequence diagram
Figure 5.16: Provision of ICS (I)
5.4.1.1.3 Description

- SMS sends a provisioning request to the INMS, specifying ICS identifier and new parameters for the service.
- INMS verifies the request and determines the validity of the requested service by applying policy rules. In the current step only INMS policy rules are verified.
- INMS makes the design of the network connections at INMS level, cause this system have enough information for that sort of decisions.
- Once INMS have made the design, it sends a request to the WDM NMS to design the WDM path in the WDM domain.
- WDM NMS proceeds to verify WDM policy rules and then to design and reserve the WDM path.
- WDM NMS sends response to INMS pointing success.
- INMS sends a request to the IP NMS to design the IP route.
- IP NMS verifies the request against policies and proceeds to the creation of the IP route.
- IP NMS sends a service created confirmation to the INMS.
- The implementation task is scheduled
- A notification is sent after the schedule.
- The activation task is scheduled
- A notification is sent after the schedule.
- At the scheduling time, INMS send a request to the WDM for the implementation and activation of the designed optical path.
- WDM NMS forwards that request to the correspondent EMSs, indicating the optical link to be implemented and activated.
- The EMSs responds to the WDM NMS indicating success in the operation.
- The WDM NMS forwards that confirmation to the INMS.
- INMS notifies to the IP NMS that a new link has been created, thus the IP topology has changed.
- The creation of a new link in the optical domain can be detected by the network elements, and notified upwards. In that case the EMS sends a notification to the WDM NMS informing about the new link created. Only the first notification of the new link is processed by the IP-NMS. If this notification from the EMS were not received, the IP NMS would send to the appropriate EMSs a new link notification.
- IP NMS sends notification to the INMS indicating that IP NMS has been informed about the new link.
- INMS sends an implementation request to the IP NMS, indicating the service identifier.
- IP NMS forwards the request to the appropriate EMSs.
- EMS sends a response to the IP NMS.
- IP NMS stores the route in the database.
- IP NMS sends a confirmation to the INMS about the service implementation.
- INMS notifies to the SMS that the service has been implemented.
- At the scheduled time the INMS sends to IP NMS an order to activate the given connection.
- IP NMS sends to EMS the order to activate the new route and receives its confirmation.
- IP NMS notifies the activation confirmation to the INMS (for WDM).
- IP NMS notifies the activation confirmation to the INMS (for IP).

5.4.1.2 Modification of ICS involving change of IP route

5.4.1.2.1 Context

Different modifications of a service can be made: change of the Network Level Agreements, change the scheduling, etc. This scenario describes a case in which the modification requires the changing of the route and the setting up of a new optical path.
5.4.1.2.2 Sequence diagram

- **Modification Request** (id, endPoints, scheduling, qos)
- **Verify Policies**
- **Create Service**
- **Create Service Request** (id, endPoints, scheduling, qos)
- **Verify Policies**
- **Routing** (ip)
- **Reserve WDM**
- **Create Service Confirmation** (OK, id)
- **Create Service** (OK)
- **Implementation & Activation Service Request** (id)
- **Implementation & Activation Service Request** (opticalLink)
- **Implementation & Activation Service Confirmation** (OK, opticalLink)
- **Implementation & Activation Service Confirmation** (OK, id)
- **Link Created** (link)
- **Link Acknowledged** (link)
- **Link Created** (link)
- **Link Created** (link)
Figure 5.18: Modification of ICS (I)
Deliverable (D2.1) - Integrated network management requirements and architecture for IP transport services

WINMAN-WP2-PTI-020a-D2-1-b1 © WINMAN Consortium
5.4.1.2.3 Description

- SMS sends a modification request to the INMS, specifying ICS identifier and new parameters for the service.
- INMS verifies the request and determines the validity of the requested service by applying policy rules. In the current step only INMS policy rules are verified.
- INMS makes the design of the network connections at INMS level, cause this system have enough information for that sort of decisions.
- Once INMS have made the design, it sends a request to the WDM NMS to design the WDM path in the WDM domain.
- WDM NMS proceeds to verify WDM policy rules and then to design and reserve the WDM path.
- WDM NMS sends response to INMS pointing success.
- INMS sends a request to the IP NMS to design the IP route.
- IP NMS verify the request against policies and proceeds to the creation of the IP route.
- IP NMS sends a service created confirmation to the INMS.
- At the scheduling time, INMS send a request to the WDM for the implementation and activation of the designed optical path.
- WDM NMS forwards that request to the correspondent EMSs, indicating the optical link to be implemented and activated.
- The EMSs responds to the WDM NMS indicating success in the operation.
- The WDM NMS forwards that confirmation to the INMS.
- INMS notifies to the IP NMS that a new link has been created, thus the IP topology has changed.
- The creation of a new link in the optical domain can be detected by the network elements, and notified upwards. In that case the EMS sends a notification to the WDM NMS informing about the new link created. Only the first notification of the new link is processed by the IP-NMS. If this notification from the EMS were not received, the IP NMS would send to the appropriate EMSs a new link notification.
- IP NMS sends notification to the INMS indicating that IP NMS has been informed about the new link.
- INMS sends an implementation request to the IP NMS, indicating the service identifier.
- IP NMS forwards the request to the appropriate EMSs.
- EMS sends a response to the IP NMS.
- IP NMS stores the route in the database.
- IP NMS sends a confirmation to the INMS about the service implementation.
- INMS notifies to the SMS that the service has been implemented.
- INMS sends to IP NMS order to proceed with the service modification.
- IP NMS sends to EMSs the order to de-active and release the old connection.
- EMSs respond to IP NMS.
- IP NMS sends to EMS the order to activate the new connection and receives its confirmation.
- IP NMS sends the modify confirmation to the INMS.
- INMS sends to IP NMS the notification that the link is going to be deleted.
- IP NMS notifies the deletion to the EMS, which responds to the IP NMS.
- IP NMS sends approval back to INMS.
- INMS sends a request to deactivate and release the optical link to the WDM NMS.
- WDM NMS forwards this request to the appropriate EMS.
- EMS responses to WDM NMS.
- WDM NMS sends to INMS the result of the latter operation.
- INMS sends confirmation to SMS about the successful service modification.
5.4.1.3 Deletion of ICS involving deletion of optical lightpath

5.4.1.3.1 Context

It is the opposite of the provision service operation. An optical path is release after the ICS is deleted.
5.4.1.3.2 Sequence diagram
5.4.1.3.3 Description

- INMS receives the request from SMS and validates it.
- INMS sends the response to SMS approving the request.
- INMS sends to IP NMS order to deactivate the connection.
- IP NMS forwards the request to the EMS.
- EMS confirms the deactivation to the IP NMS.
- IP NMS sends to INMS response about the deactivation request.
- INMS notifies to SMS that the service has been deactivated.
- INMS sends to IP NMS order to release the connection.
- IP NMS sends to EMS the order to release the old connection.
- EMS responds to IP NMS.
- IP NMS sends to INMS response about the connection release.
- INMS sends to IP NMS the notification that the link is going to be deleted.
- IP NMS notifies the deletion to the EMS, which responds to the request.
- IP NMS sends approval back to INMS.
- INMS sends a request to deactivate and release the optical link to the WDM NMS.
- WDM NMS forwards this request to the appropriate EMS.
- EMS responds to WDM NMS.
- WDM NMS sends to INMS the result of the latter operation.
- INMS schedules task for removing the ICS
- INMS notifies to SMS that the connectivity service has been released.
- INMS updates the connections (logically) data removing the one treated.
- INMS notifies to SMS that the connectivity service has been deleted.

5.4.1.4 Setting up of an optical connection

5.4.1.4.1 Context

The WINMAN operator defines the pair of ingress and egress nodes that he wants to connect with the new lightpath. The WDMNMS after identifying the appropriate WDM subnetwork forwards the request to the corresponding WDM-EMS. If this procedure is successful the INMS is updated with the information about the new lightpath.
5.4.1.4.2 Sequence diagram

5.4.1.4.3 Description

- The operator requests the creation of a new optical path through the GUI
- The WDM-NMS validates the requests
- The WDM-NMS calculates the list of possible routes
- The WDM-NMS selects the optimal route from the route list
- The WDM-NMS issues a request to one (or several) WDM-EMS to implement the optical path
- The WDM-EMS implements the path
- The WDM-EMS notifies to the WDM-NMS about the successful creation
- The WDM-NMS notifies the INMS about the updating of the state of involved resources
- The WDM-NMS notifies the operator about the successful creation

5.4.1.5 Setting up of a LSP

5.4.1.5.1 Context

The WINMAN operator creates a LSP path through the functionality provided by the network map. Basically there are two alternatives. The operator can define just the source and destination nodes or define all the intermediate nodes of the path. In the first case the IP NMS uses a routing algorithm to calculate the path while in the second case it just validates the that the List of Nodes submitted by the operator form a valid path that to the rest of the Service Level Agreement parameters.
If the required LSP spans across different IP-EMS then each intermediate IP-EMS forwards the request until it reaches the destination EMS. (The other alternative would be for the IP NMS to contact each intermediate IP-NMS)

5.4.1.5.2 Sequence diagram (I)

![Sequence Diagram](image)

**Figure 5.22:** Establish LSP (using standard routing methods)

5.4.1.5.3 Description (I)

- The operator requests the creation of a new LSP through the GUI
- The IP-NMS validates the requests
- The IP-NMS calculates the best path to satisfy the request
- The IP-NMS locks the resources that will support the path
- The IP-NMS issues a request to an intermediate IP-EMS to implement the LSP
- The intermediate IP-EMS updates the state of the network under its control and forwards the creation request
- The destination IP-EMS updates the state of the network under its control, commits the changes and informs the intermediate IP-EMS
- The intermediate IP-EMS commits the changes and informs the IP-NMS
5.4.1.5.4 Sequence diagram (II)

In the second case the WINMAN operator explicitly defines the preferred route by submitting along with the source and destination nodes, the addresses of all intermediate nodes (The only difference from the previous diagram is step 3).

![Sequence diagram](image)

**Figure 5.23**: Establish LSP (By explicitly defining all intermediate Nodes)

5.4.1.5.5 Description (II)

- The operator requests the creation of a new LSP through the GUI, indicating its route
- The IP-NMS validates the requests
- The IP-NMS calculates the best path to satisfy the request
- The IP-NMS locks the resources that will support the path
- The IP-NMS issues a request to an intermediate IP-EMS to implement the LSP
- The intermediate IP-EMS updates the state of the network under its control and forwards the creation request
• The destination IP-EMS updates the state of the network under its control, commits the changes and informs the intermediate IP-EMS
• The intermediate IP-EMS commits the changes and informs the IP-NMS
• The IP-NMS commits the changes and informs the operator
• The IP-NMS notifies to the INMS about the updating of the state of the involved resources

5.4.2. Policies scenarios

5.4.2.1 Setup Policy affecting both IP and WDM networks

5.4.2.1.1 Context

The operator of WINMAN creates a policy in the system that affect both IP and WDM networks or their management (for instance, “Filter out all non serving affecting alarms”)
5.4.2.1.2 Sequence diagram
5.4.2.1.3 Description

- The WINMAN operator using the GUI interface defines the entities on which the policies will apply.
- The INMS returns the selected entities (in detail to the GUI)
- The WINMAN Operator using the GUI, defines the conditions and the actions for the new policy.
- The INMS detects the appropriate policy manager to handle the new policy and forwards it (WDM policy section first)
- The WDM NMS verifies the policy’s syntax and semantics
- The WDM NMS will then check if there are any conflicts between the policy to be installed onto the system and the previously installed policies.
- Then if conflicts exist between the new policy and the old ones, the WDM NMS will solve them.
- Otherwise if no conflict exists between the new policy and the old ones, the WDM NMS will locate the policy destination point.
- Then the WDM NMS will store the new policy in the policy database and will notify the INMS of the status of deployment of the policy,
- The INMS will notify the WINMAN Operator of the completion of the operation.
- Then INMS will send the IP policy to the appropriate NMS (IP second)
- The IP NMS verifies the policy’s syntax and semantics
- The IP NMS will then check if there are any conflicts between the policy to be installed onto the system and the previously installed policies.
- Then if conflicts exist between the new policy and the old ones, the IP NMS will solve them.
- Otherwise if no conflict exists between the new policy and the old ones, the IP NMS will locate the policy destination point.
- Then the IP NMS will store the new policy in the policy database and will notify the INMS of the status of deployment of the policy,
- The INMS will notify the WINMAN Operator of the completion of the operation.

5.4.2.2 Activate IP policy

5.4.2.2.1 Context

A certain IP policy is deployed after the condition for its activation is met.
5.4.2.2.2 Sequence diagram

![Sequence diagram]

**Figure 5.25:** Activation of a policy

5.4.2.2.3 Description

- When a certain condition occurs, the trigger/timer will notify the IP NMS
- The IP NMS will detect the appropriate policy from the policy database/repository
- Then, the IP NMS will retrieve that policy from the policy database/repository
- The IP NMS will set the policy to the appropriate element
- The IP NMS will notify the INMS of the policy’s deployment status

5.4.2.3 Execute IP policy

5.4.2.3.1 Context

A certain condition, related to the IP network management, occurs and is detected at the INMS level. Then the INMS requests the IP-NMS to execute the policy associated with the condition.
5.4.2.3.2 Sequence diagram

![Sequence Diagram]

**Figure 5.26**: Execution of a policy

5.4.2.3.3 Description

- The IP NMS receives an external event from the EMS
- The IP NMS checks the policy conditions
- If any of the conditions are met, the IP NMS execute the actions defined by the specific policy

5.4.3. Network inventory scenarios

5.4.3.1 Notification from an IP-EMS of a new edge IP node

5.4.3.1.1 Context

One IP-EMS notifies the addition of a new IP node to the network. The node is a network element located at the edge of the IP network, and is therefore relevant for the INMS.
5.4.3.1.2 Sequence diagram

![Sequence diagram](image)

**Figure 5.27**: Notification of new IP node

5.4.3.1.3 Description

- The relevant IP EMS sends a notification to the to the IP NMS containing initial information about a new edge IP node (A previous successful registration of the relevant EMS is prerequisite).
- The IP NMS queries the IP EMS for the new edge IP node in order to verify the notification and obtain all the information that it is needed.
- The edge IP node (through the relevant IP EMS) responses with the requested information.
• The IP NMS validates the information about the new IP edge node (is this a new IP node? Is it an edge node? Etc.).
• The IP-NMS notifies the INMS about the creation of new PTPs in the IP subnetwork to which the IP node belongs.
• The INMS validates the information about the new PTPs (take appropriate actions if it deems it necessary).
• Notify the WINMAN operator about the new data.

5.4.3.2 Notification from an WDM-EMS of a new edge WDM node

5.4.3.2.1 Context

One WDM-EMS notifies the addition of a new WDM node to the WDM network. The node is a network element located at the edge of the optical network, and is therefore relevant for the INMS.
5.4.3.2.2 Diagram

![Diagram of WDM node notification process]

**Figure 5.28:** Notification of new WDM node

5.4.3.2.3 Description

- The relevant WDM EMS sends a notification to the WDM NMS containing initial information about a new edge WDM node (A previous successful registration of the relevant EMS is prerequisite).
- The WDM NMS queries the WDM EMS for the new edge WDM node in order to verify the notification and obtain all the information that it is needed.
- The edge WDM node (through the relevant WDM EMS) responds with the requested information.
- The WDM NMS validates the information about the new WDM edge node (is this a new WDM node? Is it an edge node? Etc.).
- The WDM-NMS notifies the INMS the creation of new PTPs in the WDM subnetwork to which the WDM node belongs.
• The INMS validates the information about the new PTPs (take appropriate actions if it deems it necessary).
• Notify the WINMAN operator about the new data.

5.4.3.3 Notification from an WDM-EMS of a new optical link

5.4.3.3.1 Context

One WDM-EMS notifies the addition of an optical link to the WDM network.

5.4.3.3.2 Sequence diagram

![Sequence Diagram](image)

Figure 5.29: Notification of new optical link

5.4.3.3.3 Description

- The relevant WDM EMS sends a notification to the WDM NMS containing initial information about a new optical link (A previous successful registration of the relevant EMS is prerequisite).
- The WDM NMS queries the WDM EMS for the new optical link in order to verify the notification and obtain all the information that it is needed.
- The WDM EMS responds with the requested information.
- The WDM NMS validates the information about the new optical link (is this a new link? Etc.).
- The WDM NMS sends a notification WINMAN operator about the new data.

5.4.3.4 Deletion of an IP node

5.4.3.4.1 Context
The deletion of a network node is notified from the appropriate EMS

5.4.3.4.2 Sequence diagram

5.4.3.4.3 Description
- The IP EMS sends a notification to the corresponding NMS regarding the “deletion” (unable to manage) of a node (A previous successful registration of the relevant EMS is prerequisite)
- The notification is forwarded to the INMS.
- The INMS validates the “deletion” (unable to manage) of the relevant node. The validation includes the calculation – execution of all necessary actions that permit the continuation of all established services – connections.
- The INMS notifies the WINMAN operator about the performed actions and their outcome.
5.4.3.5 Notification from an WDM-EMS of a new optical connection via control plane

5.4.3.5.1 Context

A new optical connection, not created by WINMAN, is set up in the network.

5.4.3.5.2 Sequence diagram

Figure 5.31: Notification of connection set up via control plane

5.4.3.5.3 Description

- The relevant WDM EMS sends a notification to the WDM NMS containing initial information about a new lightpath established via control plane (A previous successful registration of the relevant EMS is prerequisite).
- The WDM NMS queries the WDM EMS for the new lightpath in order to verify the notification and obtain all the information that it is needed.
- The WDM EMS responses with the requested information.
- The WDM NMS validates the information about the new lightpath (is this a new lightpath? Etc.).
- The WDM NMS sends a notification WINMAN operator about the new data.
5.4.3.6 Register WDM-EMS

5.4.3.6.1 Context
EMS sends a register to WINMAN, telling which type of Element Management System it is and also indicating that it is operational to receive commands from WINMAN.

5.4.3.6.2 Sequence diagram

![Sequence Diagram]

Figure 5.32: Register EMS

5.4.3.6.3 Description
- EMS sends registration request to the NMS
- The registration request is validated for validity (parameters and context).
- EMS data is stored in a location that is available to WINMAN
- A positive reply is sent back with a reference to the WINMAN interface.
• A notification is sent to the INMS stating that a network change has occurred
• EMS changes its state in such a way that it is able to receive requests from WINMAN

5.4.3.7 Unregister WDM-EMS

5.4.3.7.1 Context
EMS wishes to remove the connection with WINMAN.

5.4.3.7.2 Sequence diagram

![Sequence diagram](image)

**Figure 5.33: Unregister EMS**

5.4.3.7.3 Description

5.4.3.8 Description

• EMS informs WINMAN that it wishes to delete the communication.
• The EMS data is removed from the location that is available to WINMAN
• A notification is sent to the INMS stating that a network change has occurred
• A positive reply is sent back to the EMS
• EMS changes its state in such a way to be unable to receive requests from WINMAN
5.4.4. Network maintenance and restoration scenarios

5.4.4.1 Failure in the WDM network

5.4.4.1.1 Context

A failure occurs in the WDM network. This failure can be a fibre cut, a network element not working properly or any other one. Both optical and IP network equipment detect a failure in the network and raise alarms, that reach WINMAN NMSs through the EMSs. The affected ICS are not restored.

5.4.4.1.2 Sequence diagram

![Sequence diagram](image)

Figure 5.34: Failure in the WDM network

5.4.4.1.3 Description

- The WDM-EMS sends one or several alarms to the WDM-NMS.
- The WDM-NMS correlates the received alarms (if several) and finds the root cause for the alarm.
- The IP-EMS sends several alarms to the IP-NMS indicating the links that are not working.
- The IP-NMS correlates the receive alarms, and it is not able to find a root cause inside its domain.
- The WDM-NMS sends the correlated alarm to the INMS indicating the list of affected connections (Och Trails in G.872 terminology)
The IP-NMS forwards the IP alarms to the INMS.
The INMS correlates the received alarms and finds the root cause, located in the WDM network. The restoration process could start here, but that is described in another scenario. IP alarms are not cleared after the root cause has been found in the WDM network.
The INMS reports the SMS the list of ICS that have gone out of service.
The INMS sends to the GUI the correlated alarm, with an Id that will allow the GUI to ask the WDM-NMS for more data about the alarm.

5.4.4.2 IP node failure

5.4.4.2.1 Context
An IP node goes out of service. Adjacent routers detect it does not respond an issue alarms, that reach the IP-NMS through the IP-EMS. WDM equipment may detect the absence of signal caused by this IP node failure, but that possibility is not considered in the scenario.

5.4.4.2.2 Sequence diagram

![Figure 5.35: IP node failure](image-url)
5.4.4.2.3 Description

- The IP-EMS sends one or several alarms to the IP-NMS.
- The IP-NMS correlates the received alarms and locates the root cause in the failing IP node.
- The IP-NMS sends the correlated alarm to the INMS, indicating the root cause.
- The INMS performs the correlation of the received alarm. It does not find a different root cause.
- The INMS reports the SMS the list of ICS that have gone out of service
- The INMS sends to the GUI the correlated alarm, with an Id that will allow the GUI to ask the IP-NMS for more data about the alarm

5.4.4.3 Alarm clearing after WDM failure

5.4.4.3.1 Context

The failure in the optical network disappears. Network elements detect this situation and issue alarms, that reach WINMAN NMSs through EMSs.

5.4.4.3.2 Sequence diagram

[Sequence diagram image]

**Figure 5.36:** Alarm clearing after WDM failure
5.4.4.3.3 Description

- The WDM-EMS sends alarm messages to the WDM-NMS indicating the clearing of previously raised alarms. The perceived severity conveyed in the messages is ‘Cleared’.
- The WDM-NMS correlates the received alarm messages (if several) and decides that root cause alarm should also be cleared.
- The IP-EMS sends several alarms to the IP-NMS indicating the clearing of alarms.
- The IP-NMS correlates the receive alarms, and it is not able to find a root cause inside its domain.
- The WDM-NMS sends a clearing message for the correlated alarm.
- The IP-NMS forwards the IP clearing alarms to the INMS
- The INMS correlates the received alarm messages and decides that the correlated alarm should also be cleared.
- The INMS reports the SMS the list of ICS that have gone back into service.
- The INMS sends to the GUI a message indicating the clearing of the correlated alarm

5.4.4.4 Alarm clearing upon operator request

5.4.4.4.1 Context

An operator wants to clear an alarm affecting the WDM network. He requests the clearing through the GUI.
5.4.4.4.2 Sequence diagram

![Sequence Diagram]

**Figure 5.37:** Alarm clearing upon operator request

5.4.4.4.3 Description

- WINMAN Operator sends a request to the GUI for clearing an alarm in the WDM NMS. Alarm ID is provided to the GUI.
- GUI system validates the request from the Operator.
- GUI forwards the clearing alarm request to the INMS, and updates its database and the information presented to the user.
- INMS clears the alarm, updates alarm status in its alarm database and sends a request to the WDM-NMS for clearing the alarm in this system, in which the alarm was originated.
- WDM-NMS clears the alarm and updates alarm status in its alarm database.
- The INMS notifies the user, through the GUI, about the successful clearing
5.4.4.5 IP Node Problematic Status Changed

5.4.4.5.1 Context

A change of status in an IP node creates problems in the network. The IP-EMS notifies it and WINMAN detects the problem.

5.4.4.5.2 Sequence diagram

![Sequence diagram for IP node problematic status changed]

Figure 5.38: IP node problematic status changed

5.4.4.5.3 Description

- The EMS sends an status change message to the IP NMS
- The IP NMS validates the request
• If the request is invalid, the IP NMS updates the network inventory database. The WDM NMS then notifies the INMS, who will notify the WINMAN Operator using the GUI.
• The GUI will do a read on the database in order to update its displays.
• Otherwise the IP NMS will look if the operator requested the status from the specific element to be suppressed.
• The IP NMS will update the status of the specific element in the network inventory.
• The INMS will be notified
• The GUI will be notified
• The GUI will then do a Read on the database in order to update its displays.

5.4.4.6 Acknowledge of an alarm

5.4.4.6.1 Context
The WINMAN operator acknowledges an alarm.

5.4.4.6.2 Sequence diagram

Figure 5.39: Acknowledge of an alarm
5.4.4.6.3 Description

- The WINMAN Operator uses the GUI to acknowledge an alarm
- The INMS will identify the appropriate alarm handler and forward the alarm acknowledgement to it (in this case the IP NMS)
- The IP NMS will then validate the alarm (if request was valid)
- The IP NMS will check if the alarm status has been acknowledged
- The IP NMS will then set the alarm status to acknowledged and will notify the INMS.
- The INMS will notify the GUI of the completion of the acknowledgement of the IP alarm process
- The GUI will update all the relevant views.

5.4.4.7 Specify alarm filtering policy at a WDM-EMS

5.4.4.7.1 Context

The operator requests the creation of an alarm filter at the WDM-EMS.

5.4.4.7.2 Sequence diagram

Figure 5.40: Specify alarm filtering policy
5.4.4.7.3 Description

- WINMAN Operator sends a request to the GUI system for specifying a filtering policy in a specific WDM-EMS. In the request it is indicated the filter rules and the system (systemAffected) where they have to be applied.
- GUI system forwards the request to the INMS (policies have to be distributed from the INMS), indicating the same parameters the in the previous case.
- INMS sends the request to the WDM-NMS.
- WDM-NMS sends the request to the WDM-EMS, indicating only the alarm filters to be applied in the EMS.
- WDM-EMS stores the filters in its DataBase.
- WDM-EMS send a confirmation to the WDM-NMS, which is forwarded to the INMS, GUI and finally to the Operator, indicating that the Filtering Policy has been stored.

5.4.4.8 SMS requests connection status

5.4.4.8.1 Context

The SMS queries the INMS about the status of a certain ICS.

5.4.4.8.2 Sequence diagram

![Sequence diagram](image)

**Figure 5.41:** SMS requests connection status

5.4.4.8.3 Description

- SMS requests the status of a connection identified with its connection identifier. This request is sent to the INMS because is the system which manages the end-to-end connection.
- INMS responses to the SMS with the status data of the indicated connection.

5.4.4.9 Successful Recovery of ICS upon failure in the optical network

In this section a failure in the optical network is assumed and the ICS are recovered according to the proposed multi-layer survivability schemes in the project. It is assumed that the backup resources are available, therefore
this scenario is successfully accomplished. For readability purposes, the scenario has been decomposed into the 3 sub-sections, according to the three recovery classes of the ICS.

5.4.4.9.1 Successful Recovery in the optical of High Priority ICS upon failure in the optical network

5.4.4.9.1.1 Context

A failure in the optical network triggers the recovery in the optical of high priority ICS. The outcome of the recovery process is successful.

5.4.4.9.1.2 Sequence diagram

Figure 5.42: Successful Recovery in the optical of High Priority ICS upon failure in the optical network

5.4.4.9.1.3 Description

- Received alarms, indicating a root failure in the optical domain trigger the recovery process. The INMS searches to find the affected ICS. The affected ICS are sorted based on the recovery class that they belong. High priority ICS are protected by backup resources in the optical domain and therefore protection in the optical domain is triggered (protect( ))
- The WDM-NMS sorts the affected optical paths according to some criteria (e.g. bandwidth, etc) and then check if there are available resources in the backup route.
• The WDM-EMS is triggered to switch the traffic from the working to the protected route. This is performed for all the affected optical paths that serve high priority ICS.
• All the successful recovery actions are recorded to the DB.

5.4.4.9.2 Successful Recovery in the optical of Medium Priority ICS upon failure in the optical network

5.4.4.9.2.1 Context

A failure in the optical network triggers the recovery in the optical of medium priority ICS. The outcome of the recovery process is successful.

5.4.4.9.2.2 Sequence diagram

![Sequence diagram](image)

**Figure 5.43:** Successful Recovery in the optical of Medium Priority ICS upon failure in the optical network
5.4.4.9.2.3 Description

- Received alarms, indicating a root failure in the optical domain trigger the recovery process. The INMS searches to find the affected ICS. The affected ICS are sorted based on the recovery class that they belong. Medium priority ICS are re-routed in the optical domain.
- The WDM-NMS sorts the affected optical paths according to some criteria (e.g. bandwidth, etc) and then search to find an alternative route in the optical domain to re-route the affected optical paths.
- If there are free resources, the WDM-EMS is triggered to re-route the traffic to the new route. This is performed for all the affected optical paths that serve medium priority ICS.
- All the successful recovery actions are recorded to the DB.

5.4.4.9.3 Successful Recovery in the optical of Low Priority ICS upon failure in the optical network

5.4.4.9.3.1 Context

A failure in the optical network triggers the recovery in the optical of low priority ICS. The outcome of the recovery process is successful.
5.4.4.9.3.2 Sequence diagram

![Sequence diagram]

**Figure 5.44:** Successful Recovery in the optical of Low Priority ICS upon failure in the optical network

5.4.4.9.3.3 Description

- Received alarms, indicating a root failure in the optical domain trigger the recovery process. The INMS searches to find the affected ICS. The affected ICS are sorted based on the recovery class that they belong. Low priority ICS are re-routed in the electrical domain.
- The IP-NMS sorts the affected paths according to some criteria (e.g. bandwidth, etc) and then search to find an alternative route to re-route in the electrical domain the affected paths.
- If there are free resources, the IP-EMS is triggered to re-route the traffic to the new route. This is performed for all the affected paths that serve low priority ICS.
- All the successful recovery actions are recorded to the database.
5.4.4.10 Unsuccessful Recovery in the optical of ICS upon failure in the optical network

This section presents the scenario where ICS cannot be all recovered in the optical network and recovery actions should be also taken in the electrical domain. For readability purposes, the scenario has been decomposed into the 3 following sub-section, according to the three recovery classes of the ICS considered in WINMAN. This scenario is also successful in the sense that recovery is successful in the electrical layer and unsuccessful in the optical.

5.4.4.10.1 Unsuccessful Recovery in the optical of High Priority ICS upon failure in the optical network

5.4.4.10.1.1 Context

A failure in the optical network triggers the recovery in the optical of high priority ICS. The outcome of the recovery process in the optical domain is unsuccessful and recovery is attempt in the MPLS (electrical) layer.
5.4.4.10.1.2 Sequence diagram

![Sequence diagram]

**Figure 5.45:** Unsuccessful Recovery in the optical of High Priority ICS in the optical domain upon failure

5.4.4.10.1.3 Description

- Received alarms, indicating a root failure in the optical domain trigger the recovery process. The INMS searches to find the affected ICS. The affected ICS are sorted based on the recovery class that they belong. High priority ICS are protected by backup resources in the optical domain however, it is assumed that there are no available resources in the backup route and therefore protection in the electrical domain is triggered (protect( ))
- The IP-NMS sorts the affected paths according to some criteria (e.g. bandwidth, etc) and then check if there are available resources in the backup route in the electrical domain this time.
- The IP-EMS is triggered to switch the traffic from the working to the protected route. This is performed for all the affected paths that serve high priority ICS.
- All the successful recovery actions are recorded to the DB.
5.4.4.10.2 **Unsuccessful Recovery in the optical of Medium Priority ICS upon failure in the optical network**

5.4.4.10.2.1 **Context**

A failure in the optical network triggers the recovery in the optical of medium priority ICS. The outcome of the recovery process in the optical domain is unsuccessful and recovery is attempt in the MPLS (electrical) layer.

5.4.4.10.2.2 **Sequence diagram**

![Sequence diagram](image)

**Figure 5.46:** Unsuccessful Recovery in the optical of Medium Priority ICS in the optical domain upon failure
5.4.4.10.2.3 Description

- Received alarms, indicating a root failure in the optical domain trigger the recovery process. The INMS searches to find the affected ICS. The affected ICS are sorted based on the recovery class that they belong. Medium priority ICS are re-routed in the optical domain, however it is assumed that there are no available resources in the optical domain to re-route the affected paths and therefore re-routing in the electrical domain is triggered.
- The IP-NMS sorts the affected optical paths according to some criteria (e.g. bandwidth, etc) and then search to find an alternative route in the electrical domain this time to re-route the affected paths.
- If there are free resources, the IP-EMS is triggered to re-route the traffic to the new route. This is performed for all the affected paths that serve medium priority ICS.
- All the successful recovery actions are recorded to the DB.

5.4.4.11 Successful Recovery in the electrical of ICS upon failure in the electrical domain

5.4.4.11.1 Successful Recovery in the electrical of High Priority ICS upon failure in the electrical domain

5.4.4.11.1.1 Context
A failure in the electrical domain triggers the recovery in the electrical of high priority ICS. The outcome of the recovery process in the electrical domain is successful.

5.4.4.11.1.2 Sequence diagram
5.4.4.11.1.3 Description

- Received alarms, indicating a root failure in the electrical domain trigger the recovery process. The INMS searches to find the affected ICS. The affected ICS are sorted based on the recovery class that they belong. High priority ICS are protected by backup resources in the electrical domain and therefore protection in the electrical domain is triggered (protect() ).
- The IP-NMS sorts the affected paths according to some criteria (e.g. bandwidth, etc) and then check if there are available resources in the backup route.
- The IP-EMS is triggered to switch the traffic from the working to the protected route. This is performed for all the affected optical paths that serve high priority ICS.
- All the successful recovery actions are recorded to the DB.

5.4.4.12 Reversion of network to its state previous to recovery

5.4.4.12.1 Context

The purpose of this scenario is to reverse the network to the state before the recovery actions were taken. All the successful protection and re-routing actions are stored in the database, and this scenarios use these records in order to reverse the network to its previous state. The changeRoute() operation is used to reverse the path to its initial route.
5.4.4.12.2 Sequence diagram

WINMAN operator

INMS

IP NMS

WDM NMS

IP EMS

WDM EMS

revertNetwork()

retrieveRecoverRecords()

validateRecoveryRecords()

revertWdmNetwork(HIGH_PRIORITY, affectedOpticalPaths)

changeRoute()

result()

revertIpNetwork(HIGH_PRIORITY, affectedPaths)

changeRoute()

result()

revertWdmNetwork(MEDIUM_PRIORITY, affectedOpticalPaths)

changeRoute()

result()

revertIpNetwork(MEDIUM_PRIORITY, affectedPaths)

changeRoute()

result()

revertIpNetwork(LOW_PRIORITY, affectedPaths)

changeRoute()

result()
5.4.4.12.3 Description

- The WINMAN operator using the GUI wants to reverse the network to the state before the recovery actions were taken.
- The recovery actions are retrieved from the database and validated.
- Firstly, the optical paths serving high priority ICS are reversed. The WDM-NMS triggers the WDM-EMS to change the route of the traffic to the initial route \( \text{changeRoute()} \)
- Then, optical paths serving medium priority ICS are reversed. The WDM-NMS triggers the WDM-EMS to change the route of the traffic to the initial route \( \text{changeRoute()} \)
- Finally, the paths serving low priority traffic are reversed. The IP-NMS triggers the IP-EMS to change the route of the traffic to the initial route \( \text{changeRoute()} \)
- The WINMAN operator through the GUI is notified about the outcome of the whole reverse process.

5.4.4.13 Synchronize alarms between the INMS and the WDM-NMS

5.4.4.13.1 Context

The INMS wants to share the same information about active alarms with the WDM-NMS. It requests the WDM-NMS to send its list of current alarms.

5.4.4.13.2 Diagram

![Diagram of alarm synchronisation]

**Figure 5.49**: Synchronisation of alarms

5.4.4.13.3 Description

- INMS request the alarm list to the WDM-NMS
- WDM-NMS sends to the INMS the list of active alarms which exists in the WDM-NMS, with the severity and the connections affected by these alarms.
5.4.5. Network data management scenarios

5.4.5.1 Threshold crossing event sent from IP-EMS

5.4.5.1.1 Context

This scenario covers steps 1 to 5 of the “Process Performance Measurements” use case. One IP-EMS issues an event to the IP-NMS notifying some performance-related parameter has gone over or under a predefined value.

5.4.5.1.2 Sequence diagram

![Sequence diagram of IP threshold crossing event]

Figure 5.50: IP threshold crossing event

5.4.5.1.3 Description

- Event containing the variables corresponding to several counters is sent to the IP-NMS
- The IP-NMS determines, according for instance to the performance monitoring policies established in the IP-NMS which counters are logged, which ones have to be checked against threshold crossings and the values of these thresholds.
- The data selected in the previous action is logged.
- The data logged in the previous action is monitored in order to detect any crossing threshold event
• In case that a threshold is detected to be crossed, the Open Alarm use case is executed.

5.4.5.2 Retrieval of performance measures of the IP network

5.4.5.2.1 Context
The INMS retrieves performance information from an IP-NMS

5.4.5.2.2 Sequence diagram

![Sequence diagram](image)

Figure 5.51: Retrieval of performance measures of the IP network

5.4.5.2.3 Description
• The INMS issues a query for performance information to the IP-NMS
• The IP-NMS finds the data needed
• The IP-NMS sends the requested information to the INMS

5.4.5.3 Provide performance report

5.4.5.3.1 Context
This scenario covers part of the use case “Provide PM Report”. One SMS or one operator requests the INMS to produce and send a report about the performance delivered by the network.
5.4.5.3.2 Sequence diagram

Figure 5.52: Provide performance report

5.4.5.3.3 Description

- A ProvideICSPerformance request is sent from SMS
- The list of attributes of the ICS requested are mapped to IP variable domain
- The list of attributes of the requested ICS is mapped to a list of variables of the WDM domain
- The list of identified variables is requested to the IP-NMS system log
- Reply IP data.
- The list of identified variables is requested to the WDM-NMS system log
- Reply WDM data.
- The system executes the method Calculate Aggregation and Statistics in order to serve the request originated by the WO or by the SMS
• Respond to ProviderICSPerformance Request to SMS.

5.4.5.4 Start performance monitoring

5.4.5.4.1 Context

This scenario deals with the start of a monitoring process initiated by the WINMAN operator. The monitoring process is assumed to refer to parameters of the end-to-end connectivity service. This might entail for sure the request of parameters in one or in the two technological domains.

This scenario activates the performance log mechanism that is mentioned in all the Performance use cases

5.4.5.4.2 Sequence diagram

![Sequence diagram](image)

**Figure 5.53:** Start performance monitoring
5.4.5.4.3 Description

- PMP stands for Performance Monitoring Policy.
- A PMP may consist in a set of parameters related to the ICS which is going to be monitored, as well as the scheduling of the monitoring process. Among these parameters, we can specify service variables, thresholds relative to these variables and events that must be generated whether these thresholds are crossed. The events are sent to the WO. The following is an example:

```plaintext
MEASURE "jitter" AT "2s" INTERVALS
IF "jitter" GREATER THAN "1ms" ISSUE "eventA"
   eventA:: notification to the WINMAN Operator containing the message "jitter threshold has been exceeded by A%"
```

- An instance of PMP is sent to be installed in the IP-NMS
- The IP monitoring service is programmed
- SET IP-EMS Monitoring data.
- Notify from IP-EMS.
- The WDM monitoring service is programmed
- SET WDM-EMS Monitoring data.
- Notify from EDM-EMS.
- The IP-NMS notifies about the success or failure of the setup (programming) of the monitoring service. In case of success, data relevant to the IP network is started to be logged
- The WDM-NMS notifies about the success or failure of the setup (programming) of the monitoring service. In case of success, data relevant to the WDM network is started to be logged
- The INMS notifies about the success or failure of the Start of the Performance Monitoring. In case of success, data related to the performance of the selected ICS is started to be logged

5.4.5.5 Stop performance monitoring

5.4.5.5.1 Context

This scenario closes the monitoring process that was initiated as described in the scenario previous scenarios. It also deactivates the performance log mechanism that is mentioned in the PM use cases.
5.4.5.5.2 Sequence diagram

![Sequence Diagram]

**Figure 5.54:** Stop performance monitoring

5.4.5.5.3 Description

- This order removes the Policy Performance Monitor that it is assumed is operative. The removal process is covered in the Remove Policy method and detailed in the appropriate scenario.
- The monitor process associated to the IP domain is asked to be stopped.
- This notification informs that the monitoring process in the IP domain has been stopped or the reason why it was not. In case of success, the data logging is also stopped.
- The monitoring process associated to the WDM domain is asked to be stopped.
- This notification informs that the monitoring process in the WDM domain has been stopped or the reason why it was not. In case of success, the WDM data logging is also stopped.
- Notification of success or failure of the stop process. In case of success, data logging is immediately stopped.
5.5. Systems architecture

5.5.1. Reference architecture for NMS

The present section describes the architecture of a common framework for all systems that form part of WINMAN. The particularities that belong to each of the WINMAN systems (INMS, WDM-NMS and IP-NMS) are described in subsequent sections.

This generic NMS is based on a 3-tier architecture, as illustrated in Figure 5.55:

![3-tier architecture of the generic NMS](image)

**Figure 5.55**: 3-tier architecture of the generic NMS

For the purpose of carrying out the functionality of the generic NMS system the subsystems shown in Figure 5.56 and described next have been identified. Some arrows (those related to the Policy Manager and the Supporting Subsystems) have not been included in order to simplify the figure. The complete set of relationships between subsystems is shown in Table 5.1:
Table 5.1: Interfaces between subsystems. Each row indicates the subsystems that provide data or perform operations for the subsystem named in the first column

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Provisioning manager</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. End-to-end routing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Network inventory manager</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Logical tree manager</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Network restoration manager</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. Policy manager</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7. Alarm manager</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8. Alarm correlator</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9. QoS manager</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10. Threshold manager</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11. Performance collector</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5.5.1.1 Application subsystems

5.5.1.1.1 Provisioning Manager

Type: module

Functionality: this block is in charge for provisioning the ICS that users – the SMS or WINMAN operators
request through the Northbound Interface or the GUI, respectively. It manages the provisioning process, including scheduling.

Internal data: everything related to connectivity service.

Dependencies with:
- End-to-end Routing – designs connections upon request
- Policy Manager – provides provisioning policies
- Logical Tree Manager – stores information about connectivity
- Network Inventory Manager – provides information about network topology

### 5.5.1.1.2 End-to-end Routing

Type: module.

Functionality: performs the design of the end-to-end connections inside its own network, taking into account QoS constraints and routing policies.

Internal data: routing policies.

Dependencies with:
- Policy Manager – provides routing policies
- Network Inventory Manager – provides physical data about the network
- Logical Tree Manager – provides logical data about the network
- Alarm Manager – provides information about the fault status of the network
- QoS Manager – provides information about network performance

### 5.5.1.1.3 Network Inventory Manager

Type: module.

Functionality: The Network Inventory Manager is responsible to store, update, maintain and provide information about the data that WINMAN uses related to network physical resources, according to the information received from the network element layer and the GUI system. The other blocks will use these data.

Internal data: everything related to physical network topology

Dependencies with: (no dependencies)

### 5.5.1.1.4 Logical Tree Manager

Type: module

Functionality: updates, maintains and provides logical information of the network, like connectivity or capacity, according to the information received from the Provisioning Manager and from lower level applications.

Internal data: everything related to logical and connectivity data.

Dependencies with:
- Network Inventory Manager – provides physical topology information
5.5.1.1.5 Policy Manager

This component is the heart of the policy-based management system. It is constituted by the Policy Engine and the Policy Decision Point.

Type: module.
Functionality: this component is responsible of managing and providing policies, necessary to make decisions in a variety of actions. For instance, it checks a provision request against the correspondent policies. Alarm and performance mechanisms can be policy oriented. The routing and the restoration mechanisms can also be controlled by policies.
Internal data: policies data
Dependencies with:
- Network Inventory Manager – provides information about the physical network
- Logical Tree Manager – provides information about the logical network

5.5.1.1.5.1 Policy Engine

The Policy Engine receives policies to be installed. When this happens, it notifies to the Policy Decision Point that a new policy has been stored and therefore it can be retrieved to be executed. The policy repository is also managed by this component. Also, all the other interactions with the user take place at this component level. The policy conflict check is also executed in this component although it might entail the calling of specific services of the platform.

5.5.1.1.5.2 Policy Decision Point

Is the logical unit that takes the decisions concerning a given policy. These decisions can be to register a condition in the monitoring system or to send a command to a policy enforcement point as a consequence of having retrieved a policy from the database.

5.5.1.1.6 Alarm Manager

Type: module.
Functionality: receives alarms from the Lower Level Application, triggers alarm correlation, stores alarm data and distributes alarms to other systems and subsystems.
Internal data: alarms, connectivity data, logical and physical network information.
Dependencies with:
- Threshold Manager – generates performance related alarms
- Alarm Correlator – correlates alarms to find root causes

5.5.1.1.7 Alarm Correlator

Type: module.
Functionality: filters, correlates and evaluates the alarms to find out their root cause and generate new alarms, sending the results to the alarm manager.
Internal data: connectivity data, logical and physical network information.
Internal subsystems
Dependencies with:
- Network Inventory Manager – provides information about the physical network
Logical Tree Manager – provides information about the logical network
Policy Manager – provides correlation rules

5.5.1.1.8 QoS Manager

Type: module.
- Functionality: monitors and analyses the QoS data of the paths provisioned in the network and sent by the Lower Level Applications. It also provides performance data to the GUI and the Higher Level Applications.
- Internal data: performance and quality of service information about connections
- Dependencies with:
  - Performance Collector – retrieves data from lower level applications
  - Threshold Manager – checks data against configurable thresholds

5.5.1.1.9 Threshold Manager

Type: module.
- Functionality: check counters against the defined thresholds in order to generate alarms and reports if the thresholds are passed.
- Internal data: user configurable thresholds, threshold crossing alarms.
- Dependencies with:
  - Policy Manager – provides policies for the setting up of thresholds
  - Network Inventory Manager – provides information about the physical network
  - Logical Tree Manager – provides information about the logical network

5.5.1.1.10 Performance Collector

Type: module.
- Functionality: collects performance data from the Lower Level Applications.
- Internal data: policies for performance collection
- Dependencies with:
  - Network Inventory Manager – provides information about the physical network
  - Logical Tree Manager – provides information about the logical network
  - Policy Manager – provides collection policies

5.5.1.1.11 Network Restoration Manager

Type: module
- Functionality: It is responsible for the network restoration actions taken in order to prevent the disruption of the provided connectivity services. These actions are taken when alarms from the Alarm Manager arrive to the Network Restoration Manager subsystem.
- Internal data: restoration policies.
- Dependencies with:
  - Provisioning Manager – controls the provision of changes required to restore a connection.
  - Alarm Manager – generates service affecting alarms
  - Policy Manager – provides restoration policies
5.5.1.2 Supporting subsystems

5.5.1.2.1 Northbound Interface Manager

Type: interface.
Functionality: This subsystem is in charge of managing the interface between specific subsystems and the higher level applications.
Internal data:
Dependencies with: no dependencies

5.5.1.2.2 Southbound Interface Manager

Type: interface.
Functionality: This subsystem is in charge of managing the interface between specific subsystems and the lower level applications.
Internal data:
Dependencies with: no dependencies

5.5.1.2.3 Database Access

Type: interface.
Functionality: This service provides a single interface for storing data in a database. The database will contain policies, alarms and connections
Internal data:
Dependencies with: no dependencies

5.5.1.2.4 View Manager

Type: interface.
Functionality: Presents NMS related information in a formatted manner to be sent to the GUI.
Internal data:
Dependencies with: no dependencies

5.5.1.2.5 Directory Service

Type: module
Functionality: makes management information like policies or resources accessible to all elements of the system
Internal data: common management information
Dependencies with: no dependencies

5.5.1.2.6 Subsystem Administrator

Type: module.
Functionality: This service allows subsystems to be installed, registered and connected among them, thus enabling specific subsystems to locate other subsystems by name. It also performs administration functions such as monitor subsystem status, start/terminate subsystems, etc.
5.5.1.2.7 Timer Service

Type: module.
Functionality: provides interfaces for synchronising time in a distributed object environment by supplying and controlling generic timing information. It also provides operations for defining and managing time-triggered events.
Internal data:
Dependencies with: no dependencies

5.5.1.2.8 Metapolicy Service

Type: module.
Functionality: This service is an integrand part of the policy-based management functionality. Among its roles we can highlight the authorisation of the source and the policy conflict resolution at the policy deployment phase.
Internal data:
Dependencies with: no dependencies

5.5.1.2.9 Transaction Service

Type: module.
Functionality: This service is considered in order to support transactions. This service can be useful at the provisioning stages where many variables must be set and can not be left in an inconsistent state.
Internal data:
Dependencies with: no dependencies

5.5.1.2.10 Event and Notification Service

Type: module.
Functionality: This service allows subsystems to dynamically register or unregister their interest in specific events. Thus, information can be exchanged between different subsystems.
Internal data:
Dependencies with: no dependencies

5.5.2. INMS architecture

5.5.2.1 Introduction

The following figure shows the subsystems architecture for the INMS. It has been derived from the framework previously described.
5.5.2.2 Provisioning Manager

- **Functionality:** this subsystem is in charge for provisioning an ICS between two or more end-points. It includes an scheduling function which triggers the appropriate create/implement/release requests to the NMS Provisioning Manager.

Once the End-to-end Routing subsystem has computed the optimum path through the subnetworks, it sends the information to the Provisioning Manager, which stores that information and communicates with the NMSs in order to complete the design of the requested path.

- **Internal data:**
  - NLA parameters of the path
  - End-to-end points for the ICS.
  - Logical resources and entities constituting the computed path (see Logical Tree Manager).
  - State of the resources and of the path.
  - Policies specifying acceptable requests

5.5.2.3 End-to-end Routing

- **Functionality:** performs the design of the end-to-end path, taking into account QoS constraints and routing policies. The End-to-end Routing subsystem finds the optimum path, and sends the logical information to the Provisioning Manager. The End-to-end Routing modules know only about
subnetworks and links between subnetworks. The end-to-end routing inside a subnetwork is delegated via the Provisioning Manager to the NMS End-to-end Routing manager.

- Internal data:
  Routing policies

### 5.5.2.4 Network Inventory Manager

- Functionality: Stores, updates, maintains and provides physical data in the INMS domain. Physical data are received from the NMSs or the WINMAN Operator, and are related, for instance, to the information about points of connection between subnetworks.
- Internal data: the inventory information of interest at this level is subnetworks, access points and access point groups and topological links.

### 5.5.2.5 Logical Tree Manager

- Functionality: updates, maintains and provides logical information of the network, mainly connectivity information. This information comes from the Provisioning Manager (if connections are made by the management plane) and from NMSs (when connections are made by the control plane)...
- Internal data: everything related to logical and connectivity data:
  - Subnetworks
  - Subnetwork Connections
  - Connection Termination Points (CTP)
  - Link Connections (the topology)
  - State of all these managed entities
  - Capacity (used and available)

### 5.5.2.6 Policy Manager

- Functionality: this component is responsible of managing and providing policies in the INMS, necessary to make decisions in a variety of actions.
- Internal data: policies data:
  - Policies for provision/modify/delete service requests
  - Policies for selecting subnetworks
  - Policies for selecting link connections
  - Policies for fault management (like correlation rules)
  - Policies for performance management (threshold policies, collector policies)

### 5.5.2.7 Alarm Manager

- Functionality: receives alarms from the NMSs and from the Threshold Manager, triggers alarm correlation, stores alarm data and disseminates it. Received alarms are caused by faults in subnetworks and subnetwork connections and link connections or with problems in the performance of a subnetwork connection detected by the Threshold Manager. These alarms are correlated in order to find a root cause in the lower layer domain, if it exists. In this case, lower level paths affected are determined, and included in a new alarm that the Alarm Manager sends to Network Restoration Manager and to the SMS and GUI systems.
  - If the root cause cannot be found, then original fault alarms are sent to the SMS and GUI systems.
- Internal data:
  - Alarms (Subnetwork connection fault, Subnetwork fault, link connection fault, root cause alarm)
Connectivity and logical data (see Logical Tree Manager)  
network information (see Network Inventory Manager)

5.5.2.8 Alarm Correlator

- Functionality: filters, correlates and evaluates the alarms received from the Alarm Manager to find out their root cause and generate new alarms, sending the results back to the Alarm Manager. These new alarms include the possible root cause and location of the reported faults.
- Internal data:
  - Alarms
  - Physical network information (see Network Inventory Manager)
  - Correlation policies

5.5.2.9 QoS Manager

- Functionality: monitors and analyses the QoS data sent by the Performance Collector about ICS’s. It is responsible to manage the QoS thresholds (set up, modify, release) via the Threshold Manager and it sends QoS information (threshold alarms, events, etc.) to the GUI and the SMS.
- Internal data:
  - QoS parameters (bit error rate, delay, delay-jitter, etc)
  - Performance policies

5.5.2.10 Threshold Manager

- Functionality: check the current performance measures against the defined threshold values in order to generate alarms (to the Alarm Manager) when the threshold values are crossed.
- Internal data:
  - QoS parameters on ICS level of abstraction
  - Performance alarms
  - Threshold policies

5.5.2.11 Performance Collector

- Functionality: collects the performance information about ICS by combining the independent performance information received by the NMS’s. All useful data is made available to the QoS Manager
- Internal data:
  - QoS parameters (bit error rate, delay, delay-jitter, etc)
  - Collection policies

5.5.2.12 Network Restoration Manager

- Functionality: co-ordination of the protection/restoration of the underlying subnetworks based on the defined survivability strategy. Supports actions for preventing the QoS degradation of the existing connectivity services.
- Internal data:
  - Restoration policies
5.5.3. **WDM NMS architecture**

5.5.3.1 **Introduction**

The following figure shows the subsystems architecture for the WDM-NMS. It has been derived from the framework previously described.

![WDM NMS architecture](image)

**Figure 5.58:** WDM-NMS architecture

5.5.3.2 **Provisioning Manager**

Functionality: this subsystem is in charge for provisioning an optical path between two or more end-points. The scheduling function is located at the INMS level, which triggers the appropriate create/implement/release requests to the WDM-NMS Provisioning Manager.

Once the End-to-end Routing subsystem has computed the optimum optical path, it sends the information to the Provisioning Manager, which stores that information and communicates, when appropriate, with the EMSs in order to implement and activate the designed optical path.

Internal data:

- NLA parameters of the path
- End-to-end points for the optical path.
- Logical resources and entities constituting the computed path (see Logical Tree Manager).
State of the resources and of the optical path.
Policies specifying acceptable requests

5.5.3.3 End-to-end Routing

Functionality: performs the design of the end-to-end connections inside the WDM domain, taking into account QoS constraints and routing policies. The End-to-end Routing subsystem finds the optimum optical path, and sends the logical information to the Provisioning Manager.

Internal data:
Routing policies

5.5.3.4 Network Inventory Manager

Functionality: Stores, updates, maintains and provides physical data in the WDM-NMS domain. Physical data are received from the EMSs or the WINMAN Operator, and are related, for instance, to the information about a WDM node (NE).

The WDM-NMS Network Inventory Manager also sends an abstracted view of that inventory information to the Network Inventory Manager in the INMS. The inventory information of interest at the INMS is only subnetworks, access points and access point groups.

Internal data: everything related to physical data:
- Optical Managed Elements (and its parameters, such as ports, buffers, bandwidth of each port, etc.). For instance, Optical Cross-Connects, Optical Amplifiers and Optical Multiplexers are OME.
- Optical Links (and its parameters, such as bandwidth, Managed Elements that it connects, etc.)
- Shared Risk Link Group (SRLG)
- WDM-EMS data
- Physical Termination Points (PTP)
- Optical Subnetworks, to which Managed Elements belong

5.5.3.5 Logical Tree Manager

Functionality: updates, maintains and provides logical information of the network, mainly connectivity information. This information comes from the Provisioning Manager (if connections are made by the management plane) and from EMSs (when connections are made by the control plane)…

Internal data: everything related to logical and connectivity data:
- Optical Managed Elements (specifying logical data that contains, such as CTPs)
- Optical Channel Trails (or Path)
- Connection Termination Points (CTP)
- Link Connections
- Optical Channel Subnetwork Connections (OChSNC)
- Optical Access groups
- State of all this managed entities
5.5.3.6 Policy Manager

Functionality: this component is responsible of managing and providing policies in the WDM-NMS, necessary to make decisions in a variety of actions.

Internal data: policies data:
- Policies for provision/modify/delete service requests
- Policies for optical routing
- Policies for optical restoration
- Policies for fault management
- Policies for performance management

5.5.3.7 Alarm Manager

Functionality: receives alarms from the EMSs through the Southbound Interface and from the Threshold Manager, triggers alarm correlation, stores alarm data and disseminates it.

Received alarms are caused by faults in Optical Managed Elements or in Optical Links, or with problems in the performance of an element detected by the Threshold Manager. These alarms are correlated in order to find a root cause in the optical domain, if it exists. In this case, optical paths affected are determined, and included in a new alarm that the Alarm Manager sends to Network Restoration Manager and to the INMS and GUI systems.

If the root cause cannot be found, then original fault alarms are sent to the INMS and GUI systems.

Internal data:
- Alarms (Optical Link fault, Optical Managed Element fault, root cause alarms)

5.5.3.8 Alarm Correlator

Functionality: filters, correlates and evaluates the alarms to find out their root cause and generate new alarms, sending the results to the Alarm Manager. This new alarm includes the possible root cause of the reported faults.

Internal data:
- Alarms
- Correlation policies

5.5.3.9 QoS Manager

Functionality: monitors and analyses the QoS data sent by the EMSs through the Southbound Interface about Optical Manage Elements and Links. It also provides performance data to the GUI and the INMS, in a formatted and more abstract way. For instance, to the INMS it only sends QoS information about an optical trail, and not about all the links and subnetworks this trail traverses inside the WDM domain.

Internal data:
- QoS parameters (bit error rate, delay, delay-jitter, etc)
- Performance policies

5.5.3.10 Threshold Manager

Functionality: check counters against the defined thresholds in order to generate alarms when the thresholds are crossed.

Internal data:
- QoS parameters (bit error rate, delay, delay-jitter, etc)
5.5.3.11 **Performance Collector**

Functionality: collects performance data from the EMSs about Optical Managed Elements and Links, which arrives to the Performance Collector through the Southbound Interface. All useful data is made available to the QoS Manager.

Internal data:
- QoS parameters (bit error rate, delay, delay-jitter, etc)
- Collector policies

5.5.3.12 **Network Restoration Manager**

Functionality: It is responsible for the network restoration actions taken in order to prevent the disruption of the provided connectivity services. These actions are taken when alarms from the Alarm Manager arrive to the Network Restoration Manager subsystem.

If a root cause is found in the WDM domain, a restoration process can start, based on policies and NLA. Otherwise, if the root cause is found in the INMS, and it is located in the WDM domain, the INMS sends a restoration request to the WDM-NMS Network Restoration Manager.

When the Restoration Manager triggers a restoration, the Provisioning Manager performs the changes that are needed in the WDM domain, independently of a possible restoration in the IP domain.

Internal data:
- Restoration policies

5.5.4. **IP NMS architecture**

5.5.4.1 **Introduction**

The architecture of the IP NMS described hereafter is aligned with the generic architecture that has been selected to drive the architecture of all the WINMAN NMSs. This means that we made use of the same functional blocks and we adopt their basic interrelationships.

In order to present a readable model we provide three views, each in a different picture, namely the configuration view, the performance view and the fault view. Each view contains the subsystems relevant to the corresponding management area. We can say that in case that we were interested to implement only the configuration management functionality it would be needed only the elements appearing in the configuration management view and so on. The three views are attached in this introduction to be referenced in the description of the subsystems.

This architecture is also compliant with the IETF policy-based management model. It identifies a Policy Manager which is functionally equivalent for the three views and three Policy Enforcement Points (PEPs). The PEPs are in fact the aggregation of other functional blocks. More specifically we identified:
- Provisioning Manager and Network Restoration Manager as PEP-1 in the configuration management view
- QoS Manager, Threshold Manager and Performance Collector as PEP-2 in the performance management view
- Alarm Manager and Alarm Correlator as PEP-3 in the fault management view

This points out that in WINMAN we understand the Policy Based Management concept not as an additional management function but as the same management functions (or part of them) of conventional systems under the control of policies.
Deliverable (D2.1) - Integrated network management requirements and architecture for IP transport services

Figure 5.59: The IP-NMS configuration management view

Figure 5.60: The IP-NMS performance management view
5.5.4.2 Provisioning Manager

- Functionality: this block is in charge of provisioning the end to end connectivity that users the INMS or WINMAN operators request through the Northbound interface or the GUI. It manages the provisioning process, including scheduling.

  Internal data: everything related to connectivity service.

  Internal subsystems: LSP Functional Block, Network Functional Block

Figure 5.62 presents the detailed architecture of the MPLS manager. Round squares represent active components and the other squares represent the data entities used by these components. The data entities can be stored in a simple database.
Two major functional blocks, displayed in dark shadow, were identified: the Network and the LSP. The Network Functional Block manages network resources and decides about path routes. The LSP Functional Block manages the creation/deletion, implementation/release and activation/de-activation of end-to-end Label Switched Paths under QoS constraints.

### 5.5.4.2.1 Network Functional Block

The Network functional Block is responsible for an IP/MPLS domain management and consists of five components and three data entities. The components are: 1) Network Manager; 2) IP Domain Manager; 3) IP Router Manager; 4) MPLS Domain Manager; 5) LSR Manager. The data entities are: 1) Network Map; 2) IP Router; 3) LSR.

#### 5.5.4.2.1.1 Network Manager

The Network Manager performs the inter domain IP/MPLS management and path selection under constraints (QoS, schedule, policies). This component is aware of the topology of both domains, IP and MPLS so it is used to make mappings between them and to select the best routes.

It has a role similar to INMS whereas, instead of mapping IP to WDM, it is in charge of mapping IP to MPLS domains. When a LSP is implemented the path consists of a list of hops that need to be translated into LSRs and interfaces, so that the segments in each LSR can be allocated.

The computation of paths under constraints is carried out in this component, since it is the only component that has access to all the information required. This component receives a list of paths from the IP Domain Manager and a list of scheduled LSPs that share the time slot with the LSP being computed. By knowing the hops associated to each scheduled LSP and their resources, the Network Manager, helped by a set of policy rules, selects a path satisfying both the QoS requirements and the network resources available.
5.5.4.2.1.2 IP Domain Manager

The IP Domain Manager is responsible for the management of an IP domain. This component is responsible for the computation of IP paths between two end-points without QoS. OSPF is considered here, so that, a set of routes for each type of service can be offered. OSPF is supposed to be emulated and applied offline based on information available from the Network Map (NTW_MAP).

5.5.4.2.1.3 MPLS Domain Manager

The MPLS Domain Manager manages the MPLS resources. It provides the MPLS domain topology to the Network Manager to map MPLS to IP domains. This component is used during the implementation of an LSP, for the allocation of the required LSR resources.

5.5.4.2.2 LSP Functional Block

The LSP functional block is responsible for the global and individual management and scheduling of end-to-end Label Switched Paths under constraints. This functional block consists of three components and one data entity. The components are: 1) Path Manager; 2) LSP Manager; 3) Scheduler. The data entity is LSP.

5.5.4.2.2.1 Path Manager

The path manager is responsible for the scheduling of new LSPs. This component creates and deletes the instances of the LSP Manager component. When a create ICS event arrives to this component, a LSP Manager instance is created (to compute the paths under QoS constraints). In case failure this instance is deleted. If the create succeeds the Path Manager sends a message to the Scheduler to schedule the implementation, activation, de-activation and release of this LSP. It also manages the pre-emption of LSPs, so that arriving higher priority LSPs can be satisfied. The end-to-end connectivity under constraints is offered by this component to the North Bound interface Manager.

5.5.4.2.2.2 LSP Manager

The LSP Manager will be in charge of managing a particular Label Switched Path and transforming the path management requests into orders that are understandable by the Network Manager so that it can allocate resources in the LSRs. An instance is created when the Path Manager receives a Create ICS request, after the policies are verified. In case of failure this instance is deleted. The main function is to trigger a computation of the path under constrains and then manages the path created (implementation, activation, de-activation and release).

5.5.4.2.2.3 Scheduler

The Scheduler main function is to time-schedule the implementation, activation, de-activation and release of LSPs. Figure 5.63 represents the scheduling of a new LSP implementation and release (for activation the figure is similar). When requested by a particular LSP Manager, the scheduler shall provide it with a list of its concurrent LSPs. With this information, the LSP Manager can request the Network Manager for computing the best MPLS path that, in turn, will be scheduled for future computations.
5.5.4.2.2.4 LSP

The LSP data entity models a Label Switched Path and contains information that is similar to the traffic engineered tunnel object, which can be found in an MPLS MIB [MPLS-TE-MIB]. It contains also additional information on the implementation/release and activation/de-activation times. Among other attributes, the LSP data entity is also characterised by its route through the MPLS domain, described by hops, and its traffic/quality characteristics.

The MIB objects from which this data entity was derived consist of the following tables, as shown in Figure 5.64:

- Tunnel table (TUNNEL) for setting up MPLS tunnels;
- Resource table (RESOURCES) for setting up the tunnel associated resources (e.g., ResourceMaxRate, resourceMeanRate, resourceMaxBurstSize);
- Tunnel hop table (HOPS) for configuring strict and loose source routed MPLS tunnels hops;

![Figure 5.64: LSP data entities relationship](image)

5.5.4.3 End-to-end Routing

Due to the MPLS specific mechanism, this module can be considered as embedded in the Provisioning Manager module.

5.5.4.4 Network Inventory Manager

Functionality: The Network Inventory Manager is responsible to store, update, maintain and provide information about the data that IP-NMS uses related to network physical resources, according to the information received from the network element layer and the GUI system. The other blocks will use these data.

Internal data: everything related to physical data.

5.5.4.5 Logical Tree Manager

Functionality: updates, maintains and provides logical information of the network, like connectivity or capacity, according to the information received from the Provisioning Manager and from lower level
applications.

Internal data: everything related to logical and connectivity data.

5.5.4.6 Network Restoration Manager

Functionality: It is responsible for the network restoration actions taken in order to prevent the disruption of the provided connectivity services. These actions are taken when alarms from the Alarm Manager arrive to the Network Restoration Manager subsystem.

Internal data: restoration policies

5.5.4.7 Policy Manager

Functionality: this component is responsible of managing and providing policies, necessary to make decisions in a variety of actions. For instance, it checks a provision request against the correspondent policies. Alarm and performance mechanisms can be policy oriented. The routing and the restoration mechanism can also be established by policies.

Internal data: policies data

Internal subsystems: Policy Engine, Policy Decision Point

5.5.4.8 Alarm Manager

Functionality: receives alarms from the Lower Level Application, triggers alarm correlation, stores alarm data and distributes alarms to other systems and subsystems.

Internal data: IP alarms

5.5.4.9 Alarm Correlator

Functionality: filters, correlates and evaluates the alarms to find out their root cause and generate new alarms, sending the results to the alarm manager

Internal data: connectivity data, logical and physical network information.

5.5.4.10 QoS Manager

- Functionality: monitors and analyses the QoS data of the paths provisioned in the network and sent by the Lower Level Applications. It also provides performance data to the GUI and the Higher Level Applications.

Internal data: QoS information (delay, throughput, traffic load) about connections in the IP domain.

5.5.4.11 Threshold Manager

- Functionality: check counters against the defined thresholds in order to generate alarms and reports if the thresholds are passed.

- Internal data: user configured thresholds; performance measures; threshold policies.

5.5.4.12 Performance Collector

- Functionality: collects performance data from the Lower Level Applications.

- Internal data: collection policies
5.6. GUI architecture

5.6.1. Overview

The WINMAN integrated Graphical User Interface (GUI) should be a general-purpose tool that could be used to:

- Convey the management functions to the user/operator in an abstract and easily perceptible way
- Test WINMAN external interfaces
- Test and verify the function of WINMAN internal components

The proposal for the WINMAN integrated GUI is a thin browser-based client (for instance a Web browser), that could test all WINMAN functionality. The WINMAN GUI should be a platform independent application for operators of underlying management systems. It should provide a common look and feel for managing the underlying subsystems, whose technologies, interfaces and protocols may be very different from each other.

The GUI should interface with only one subsystem – the view manager –, which will act as a middle tier. In this way the view manager can be responsible of presenting to the GUI only the elements that are needed for displaying the view that the user asked which will saves time and computer resources.

In order to have with a common look and feel as well as a common starting point for a human manager we need to define a common style guide for the all the actions that have a similar presentation mode such as form, charts, tree views etc

5.6.2. Functional Architecture

In a 3-tier functional architecture the main functional components are the following:

A “thin” client
A business logic middleware layer
The back end server or repository

A «thin» client holds minimal functionality, such as repository and business logic intelligence. However, the client runs mainly presentation and communications functionalities, which allow one to access services over the network and display the results to the user.

The majority of data management and business logic intelligence is executed in the server layers, which are the middleware layer and the repository functional components. The business logic middleware layer interacts with both the thin client and the back end functional server. In other words client functionality never accesses repository data directly, but only through the middleware layer. In this way the maintenance is concentrated on middleware and repository side.

The WINMAN framework utilises the three-tier approach as shown in Figure 5.65. In the three-tier approach with thin client concept, the main idea is to distinguish the presentation layer from the application or business logic (middleware layer with access and control) and service specific functionality (base layer). The service specific functionality in this case is the service components (so called back-end servers), which may be traditional NMS managers or agents or some other management applications. The Controller component’s main functionality is to provide the validate rules of data input.
An example of mapping this functional architecture to the physical one using the Web technology is the following:

- The Web browser, with Java enabled «applet» downloading capability), (for presentation),
- The Web application server (for business logic or application layer),
- The database (for data repository).

A possible technology for the mapping of functional to physical architecture is that the middleware gateway can be CORBA based (e.g. ORBIX ORBWeb), capable of communicating with the underlying management systems. The gateway should have two main functions, redirect the traffic between the parties and make translations between protocols.

In the WINMAN architecture diagram we identified the Application logic as the View Manager component, which interfaces with all the other components in order to get its data. The view manager is not connected directly to the Data repository but is using other components such as the Alarm manager and the Persistency manager. The Controller can be any system that saves the GUI metadata.

### 5.6.2.1 Data Caching

One of the main concerns of designing software systems is that users are very sensitive to the response time of the system. Therefore, high variability in the response time of the user interface should be prevented. So, building a GUI that displays large amounts of information stored and managed by a repository system can be very challenging. Client data caching appears as the best approach to deal with the performance problems of the user interface. Repository information cached in the client’s functional system can be directly used for user interface manipulations. This can reduce secondary storage accesses and communication overhead. However, data caching as has been designed and implemented in current systems does not completely address the user interface requirements.

### 5.6.2.2 Functional Properties

The main idea for the functional architecture of the GUI system is that a thin client should be adopted for presentation purposes, and the rest of the functionality is built in the other server layers. The information is accessed through the network. In this way the following functional properties are evident:

- It can be started securely from anywhere in the network using any client/software supporting communication over the network.
It collects all the managements GUIs in one place where they can easily be found and launched. It shows the status of different functional subsystems, and the user can get more detailed information of the situation in separate visualisation instances (e.g. windows). It is easy to add new tasks to the GUI framework and also update the characteristics of the existing tasks dynamically.

5.6.2.2.1 Advantages and Disadvantages

In detail the advantages of the GUI are:
- Operators have a single integrated visualisation console for all operations
- The client functional layer is easy to handle and cheap
- Remote access is easy and cheap
- Mobility of the operator is possible
- Distributed development work of management GUIs between different developers is also easy using the integrated WINMAN GUI concept because individual developers can independently implement GUIs, which can be integrated with the WINMAN integrated GUI framework transparently.
- Due to its flexibility and ease of use the integrated WINMAN GUI can be utilised as a testing and demonstrating tool for various purposes by just configuring it case by case.

The disadvantages are mostly based on the communication over the network and the network itself:
- Security may be a problem
- Communication latency
- Functionality of user interfaces may be limited
- Although these problems are real at the moment, they have been noticed and there will be improvements in all problem areas.

5.6.2.3 Model-View-Controller Architecture

The Model-View-Controller (MVC) pattern plays a fundamental role in deploying GUI objects. It is closely interwoven with object-oriented design (OOD). The three core abstractions of MVC are described in short (see Figure 5.66):

The **Model** holds all data relevant to an entity or process, and performs processing on that data. Data is independent of the component’s visual representation. The Model notifies the View that its state has changed and the view needs to be redrawn.

The **View** accesses and displays data contained in the Model. It knows about the Model only to the degree allowed by the Model’s “public property interface”.

The **Controller** is the glue between View and Model. It reacts to gestures and events in the View enacted by the user, which may result in update of the data in the Model. The Controller knows about the Model only to the degree allowed by the Model’s “public behaviour interface”.

![Figure 5.66: The MVC paradigm](image)
The MVC pattern can be contemplated in combination to the pre-mentioned 3-tier architecture. The View is on the top layer, and the Model is located at the two lower layers.

**5.6.3. GUI Functional Components – Graphical view**

The GUI is responsible of getting the Data model and the presentation views and displaying them to the user. In order to do so the GUI has 2 components for retrieving the data and many widget components that handles all types of GUI widgets as described in the Figure 5.67.

![Figure 5.67 GUI Functional components](image)

**5.6.3.1 External interfaces component**

**5.6.3.1.1 Model Manager**

Gets the object model and data that needs to be presented from the view manager in the middleware. For example list of NE or list of alarms.

**5.6.3.1.2 Viewer**

Gets the presentation data from the XML files or DB regarding the labels colours and other.

**5.6.3.2 Display components**

**5.6.3.2.1 Window manager**

This is the controller that manages the window display according to the presentation data, the session attributes, the requested widgets and the object model. For example the window will get the type of widgets that are
needed from the presentation data and their format, will get the session attributes such as user and actions from the session manager and the data that needs to be displayed from the model manager.

### 5.6.3.2.2 Widgets

The widgets are responsible for all the controllers that are displayed. As noted elsewhere in this document, widgets should enable easy programming in a plug & play environment according to definitions done in the presentation data. An instance of a widget will either be editable (i.e., user enters the input) or subscribed (i.e., server initiates changes).

All widgets will implement the Model-View-Controller paradigm (MVC).

![Model-View-Controller paradigm](image)

**Figure 5.68 Model-View-Controller paradigm**

- **View**: A visible gadget used for user interaction
- **Model**: Stores the data
- **Controller**: Manages the view, generates behaviour out of metadata, acts as driving API for the entire widget

Other gadgets include non data-aware GUI components such as:
- Menus (pull-down + popup), toolbars
- Question of context – disable when an option is invalid
- System menus and other navigation options
- Buttons

Here are some examples of the widgets and their behaviour:

#### 5.6.3.2.2.1 Table

- **View**: Presents the columns, labels, fonts, colours
- **Model**: Stored data of alarms or Network inventory
- **Controller**: Monitor, static data display, Filtering, sort, Actions done on the table such as alarm acknowledge

#### 5.6.3.2.2.2 Charts

- **View**: 2D, 3D, pie chart, colours
- **Model**: Stored data of performance counters
- **Controller**: Trend graphs

#### 5.6.3.2.2.3 Tree

- **View**: Icons, number of steps
Model
Controller

5.6.3.2.2.4 Maps

View
Model
Controller

5.6.3.2.3 Session Manager

The session manager manages all the transactions done in one client session. This is needed in a web environment in order to prevent login on each window that opens. The session manager handles the user parameters and its preferences.
6. Conclusions

In this document, we propose an alternative approach versus most trends in IP/WDM integration. Instead of extending the distributed Internet network control approach to the Optical Layer using signalling, WINMAN proposes an alternative approach by employing the telecom-style network management in both the WDM and IP layer in an integrated way. The feasibility of the approach is guaranteed by the deployment of the MPLS Internet packet switching protocol, which resembles a connection-oriented network. This approach has been adopted and is being carried out in WINMAN, whose aim is to offer an integrated network management solution for the provisioning of end-to-end IP connectivity services. WINMAN consists of an open, distributed, and scalable management architecture supporting multi-vendor, multi-technology environments. The project will contribute to the establishment and operation of worldwide IP over WDM networks. The trials envisaged in the project would demonstrate inter-connectivity across a worldwide network management infrastructure. The proposed business model and system architecture facilitate the development, provision and validation of a novel Integrated Network Management architecture for future IP networks. Example scenarios have been given showing basic path provisioning and alarm reporting functionality, and these will be dealt in detail through the project lifetime, and they will be properly designed, implemented, tested in-site and in-field, to provide the appropriate results to the research community.
References

Chapter 2

[6BONE] 6Bone site, www.6bone.net


[CISCO-01] Network Management Solutions for IP-VPN Services (nvpn_wp.pdf)

[CISCO-02] Managed MPLS VPNs (VPN_Manged MPLS_TSD.pdf)


[I2] Internet 2 website, www.internet2.edu


[IETF-2] A Framework For Integrated Services Operation Over DiffServ Networks (draft-ietf-issl-diffserv-rsvp-05.txt)


[INFOR] [IN Forum, «http://www.inf.org]


[INTCOMP] IEEE Internet Computing, Special Issue: Converging on Internet Telephony, May/June 1999


[IntServ-DiffServ] A Framework For Integrated Services Operation Over DiffServ Networks (draft-ietf-issl-diffserv-rsvp-05.txt)


ITU-T, "Listener echo in telephone networks". Recommendation G.126 (03/93).
ITU-T, "One-way transmission time". Recommendation G.114 (05/00).
ITU-T, "Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s". Recommendation G.723.1 (03/96).
ITU-T, "C source code and test vectors for implementation verification of the G.729 8 kbit/s CS-ACELP speech coder". Recommendation G.729 (03/96).
ITU-T, "Video codec for audiovisual services at p x 64 kbit/s". Recommendation H.261 (03/93).
ITU-T, "Video coding for low bit rate communication". Recommendation H.263 (02/98).
ITU-T G.872, Architecture of Optical Transport Networks
ITU-T, "LISTENER ECHO IN TELEPHONE NETWORKS" Rec. G.126 (03/93)
MEM Membership information http://www.internet2.edu/ucaid/html/membership.html
MEM Membership information http://www.internet2.edu/ucaid/html/membership.html
MEM-AF Internet2 University members, http://www.internet2.edu/html/affiliates.html
MEM-UNI Internet2 University members, http://www.internet2.edu/html/universities.html
MPLS, Optical Networks Magazine, April 2000 MPLS, Optical Networks Magazine, April 2000
NGI NGI.gov website, www.ngi.gov
NGI-2 NGI.org website, www.ngi.org
NGI-3 Alternative NGI site, www.ccic.gov/ngi
NGI-4 NGI Initiative site @ www.jhu.edu/~hac_ns/ngi
ORCHESTREAM-01 One Touch IP Service Activation (prov_datatsheet.pdf)
QoS protocols & architectures (qosprot_v3.pdf)
SBM-01 SBM (Subnet Bandwidth Manager): A Protocol for RSVP-based Admission Control over IEEE 802-style networks (RFC 2814)
Chapter 3


[23] Peter Ashwood-Smith, Bilel Jamoussi, Don Fedyk, Darek Skalecki; “Improving Topology Data Base Accuracy with LSP Feedback”; Internet draft; July 2000; draft-ietf-mpls-te-feed-01.txt


[25] Dave Katz; Derek Yeung “Traffic Engineering Extensions to OSPF”; Internet draft; October 1999; draft-katz-yeung-ospf-traffic-01.txt

Chapter 4

[UC1] Use Cases in Theory & Practice Online Presentation by Alistair Cockburn


[G.852.2] ITU-T Recommendation G.852.2 - Enterprise viewpoint description of transport network resource model; 03/99


[X.731] ITU-T Recommendation, State Management Function, Rec. X.731

Chapter 5


Abbreviations

ATM Asynchronous Transfer Mode
ATM NE ATM Network Element
BPF Business Process Framework (TMF)
BWDM Broadband WDM
CCD Call Clearing Delay
CTP Connection Termination Point
DM Domain Manager
DSL Digital Subscriber Line
DWDM Dense WDM
EML Element Management Layer
EMS Element Management System
EMSL Element Management Layer
EMS Element Management System
FTTB Fiber To The Business
FTTC Fiber To The Curb
FTTcab Fiber To The Cabinet
FTTD Fiber To The Desktop
FTTO Fiber To The Office
FTTT Fiber To The Town
FTTV Fiber To The Village
GUI Graphical User Interface
ICS IP Connectivity Service
IDM Integrated Domain Manager
INMS Integrated Network Management System
IP Internet Protocol
IP NE IP Network Element
IP-NMS IP Network Management System
LSP Label Switched Path
ME Managed Element
MF Management Function (ITU M.3400)
MoIP Multimedia Over IP
MPEG Moving Pictures Experts Group
MPLS Multi-Protocol Label Switching
MVC Model-View-Controller
NI Northbound Interface
NLA Network Level Agreement
NMS Network Management System
OADM Optical Add-Drop Multiplexer
OSPF Open Shortest Path Forwarding
OTN Optical Transport Network
OVPN Optical VPN
OXC Optical cross-connect
OXC PBNM Optical Cross Connect
OXC PBNM Policy Based Network Management
OXC Optical Cross Connect
PBNM Policy Based Network Management
PDD Post Dialling Delay
PDP Policy Decision Point
PEP Policy Enforcement Point
PM Performance Management
PTP Physical Termination Point
QoS Quality of Service
R0 WINMAN Release 0
R1 WINMAN Release 1
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>SDH</td>
<td>Synchronous Digital Hierarchy</td>
</tr>
<tr>
<td>SDH NE</td>
<td>SDH Network Element</td>
</tr>
<tr>
<td>SDSD</td>
<td>Start Dial Signal Delay</td>
</tr>
<tr>
<td>SI</td>
<td>Southbound Interface</td>
</tr>
<tr>
<td>SLA</td>
<td>Service Level Agreement</td>
</tr>
<tr>
<td>SM</td>
<td>Service Management</td>
</tr>
<tr>
<td>SMS</td>
<td>Service Management System</td>
</tr>
<tr>
<td>SONET</td>
<td>Synchronous Optical Network</td>
</tr>
<tr>
<td>TDM</td>
<td>Time Division Multiplexing</td>
</tr>
<tr>
<td>TMF</td>
<td>TeleManagement Forum</td>
</tr>
<tr>
<td>TOM</td>
<td>Telecom Operations Map (TMF)</td>
</tr>
<tr>
<td>TTP</td>
<td>Trail Termination Point</td>
</tr>
<tr>
<td>UML</td>
<td>Unified Modelling Language</td>
</tr>
<tr>
<td>VoIP</td>
<td>Voice Over IP</td>
</tr>
<tr>
<td>VPN</td>
<td>Virtual Private Network</td>
</tr>
<tr>
<td>WADM</td>
<td>Wavelength Add/Drop Multiplexer</td>
</tr>
<tr>
<td>WDM</td>
<td>Wavelength Division Multiplexing</td>
</tr>
<tr>
<td>WDM NE</td>
<td>WDM Network Element</td>
</tr>
<tr>
<td>WDM-NMS</td>
<td>WDM Network Management System</td>
</tr>
<tr>
<td>WXC</td>
<td>Wavelength cross-connects</td>
</tr>
</tbody>
</table>
Appendix A. Requirements Capture Methodology

This section describes how to use the natural language (English language) to write requirements. For the descriptions one should use simple and concise language. Wherever possible complex sentence constructions, long sentences, and paragraphs and ambiguous terminology should be avoided. See also [REQ1].

- Keep sentences short.

  Because of the limitation of our short-term memory, one often needs to read long sentences more than once before she/he can understand them.

- Never express more that one requirement in a single sentence.

  Avoid the use of ‘and’ in sentences. This implies that more than one requirement or concept is being discussed.

- Avoid the use of jargon, abbreviations and acronyms unless it is certain that all readers of the document will understand them.

  Even then be very careful; the same acronym may have different meanings in different domains. For example, the acronym ‘ATM’ is used in both banking systems and networking application domains. In a requirements document, which describes how ATMs (Automatic Teller Machines) should be connected using ATM (Asynchronous Transfer Mode), one can imagine the confusion that could arise.

- All acronyms used in requirements shall be defined in the project glossary.

- Keep paragraphs short.

  As a general rule, no paragraph should be made up of more than seven sentences. Again, the reason for this is due to the limits on people’s short-term memory.

- Use lists and tables wherever possible to present information sequences.

  Lists (like this one) are much easier to understand than sequences that are presented as a single paragraph.

- Use terminology consistently.

  Do not use a term to mean one thing in one place in the document and something different somewhere else. This is very difficult to achieve especially when different people are responsible for writing different parts of the document. Using a data dictionary to define the names of system entities can be helpful.

- Use words such as ‘shall’, ‘should’, ‘will’ and ‘must’ in a consistent way with the following meanings:

  ‘shall’ indicates that the requirement is mandatory

  ‘should’ indicates that the requirement is desirable but not mandatory

  ‘will’ indicates something that will be externally provided

  ‘must’ is best avoided. If used, it should be a synonym for ‘shall’.

- Do not express requirements using nested conditional clauses [i.e. if X then if Y then R1a else if Z then R1b else R1c).

  These are very easy to misunderstand. If one cannot find a convenient way of expressing a requirement in natural language without nested conditional clauses, she/he should use a different notation such as a decision table.
• Use the active rather than the passive voice, particularly when describing actions taken by people or the system.

  Unfortunately, most technical writing is written impersonally and many people find it difficult to change to a more personal style.

• Do not try to express complex relationships in a natural language description. Diagrams are much more effective for this purpose (see the next guideline).

• Never use anonymous references. When one refers to other requirements, tables or diagrams, she/he should give a brief description of what she/he is referring to, as well as the reference number. If one simply uses a reference number, the readers may need to look it up specially to remind themselves what it means.

• Pay attention to spelling and grammar. Poor spelling and grammar can obscure the meaning. A spelling checker should always be used. The style guide should be included as part of the standard for specifying requirements. When one organises requirement reviews, she/he should ask reviewers to comment on the writing style used and to highlight requirements that are particularly difficult to understand.

• Requirements and descriptive text shall be separated.

Clearly identify which is the requirement. Additional text that helps to understand the requirement should be identified as descriptive text.
## Appendix B. Use Cases Detailed Description

### B-1. UCs Template

In the following table the conventions for describing UCs in a formal way are listed:

<table>
<thead>
<tr>
<th>USE CASE #</th>
<th>&lt;Use Case number&gt; ; &lt;Use Case name&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Use Case number</td>
</tr>
<tr>
<td></td>
<td>According the Use Case numbering scheme. This could also be an indicator consisting of an alphanumeric part (describing the UC category) and numeric part.</td>
</tr>
<tr>
<td></td>
<td>Use Case name</td>
</tr>
<tr>
<td></td>
<td>The Use Case name is a short active verb phrase version of the goal statement</td>
</tr>
<tr>
<td>Goal in Context</td>
<td>&lt;a longer statement of the goal in context if needed&gt;</td>
</tr>
<tr>
<td>Scope &amp; Level</td>
<td>&lt;scope: what system is being considered black box under design&gt;</td>
</tr>
<tr>
<td></td>
<td>This should be in most cases the &quot;WINMAN Solution&quot;</td>
</tr>
<tr>
<td></td>
<td>&lt;level: one of: Summary, User, Atomic&gt;</td>
</tr>
<tr>
<td></td>
<td>The levels identified above, refer to the functional context of the UC under specification. The criterion for this classification would be the level of abstraction and related broad/narrow context and modularity of the UC. In more detail:</td>
</tr>
<tr>
<td></td>
<td><strong>Summary level</strong></td>
</tr>
<tr>
<td></td>
<td>Represents collections of User and/or Atomic Level Goals. These UCs are normally categories of actions related to complex and generic tasks.</td>
</tr>
<tr>
<td></td>
<td><strong>User level</strong></td>
</tr>
<tr>
<td></td>
<td>It represents a primary task or elementary business process. It may also be an intermediate collection of atomic goals. This is the main level of interest for an Actor.</td>
</tr>
<tr>
<td></td>
<td><strong>Atomic level</strong></td>
</tr>
<tr>
<td></td>
<td>It represents a self-contained (max modularity) sub-goal, below the main level of interest for an Actor. The functionality contained here should be orthogonal to the rest of the Atomic level UCs.</td>
</tr>
<tr>
<td>Pre-conditions</td>
<td>&lt;what we expect is already the state of the world&gt;</td>
</tr>
<tr>
<td></td>
<td>Any prerequisite before the Use Case can be started.</td>
</tr>
<tr>
<td>Post-conditions</td>
<td>&lt;The state of the system after the Use Case ends&gt;</td>
</tr>
<tr>
<td>Success End Condition</td>
<td>&lt;the state of the world upon successful completion&gt;</td>
</tr>
<tr>
<td>Failed End Condition</td>
<td>&lt;the state of the world if goal abandoned&gt;</td>
</tr>
<tr>
<td>Primary Actors</td>
<td>&lt;a role name or description for the primary actor&gt;</td>
</tr>
<tr>
<td></td>
<td>A Primary Actor is one having a goal requiring the assistance of the System Under Development. The goal of the Actor is addressed by the UC. Therefore, it may be assumed that in most cases it is the Primary Actor that triggers the execution of the UC.</td>
</tr>
<tr>
<td>Secondary Actors</td>
<td>&lt;other systems relied upon to accomplish use case&gt;</td>
</tr>
<tr>
<td></td>
<td>An Actor from which the System Under Development needs assistance to satisfy its goal. Therefore the Secondary Actor may only participate in the execution of the UC process.</td>
</tr>
<tr>
<td>Trigger</td>
<td>&lt;the action upon the system that initiates the use case&gt;</td>
</tr>
</tbody>
</table>
External event that causes the Use Case to start. Triggering may be
effected by a Primary Actor directly, or indirectly through another UC.

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
</table>
| 1    | The main scenario is described here in steps from trigger to
goal delivery, and any cleanup after. The scenario describes
what the Actor provides and what the System does in response> |
| 2    | <..>   |
| 3    |        |

**EXTENSIONS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
</table>
| 1a   | <condition causing branching> : <action or name of sub-use case>
At certain steps in the main scenarios there may be certain
conditions that cause branching, i.e., deviation from the main
scenario flow into alternative flows. These conditions are dealt
with in this part of the Use Case describing the condition and
the steps that form an alternative route resulting into an
alternative scenario to get to the Actor's goal.
Extensions can be thought of as inserted functionality in the
main Use Case, such that the main Use Case is not aware of it.
This allows to change or add functionality to the main Use Case
without changing it. Such extensions are used to model optional
parts of the main Use Case in order to get a specialization.
Exceptions:
Failure conditions are natural extensions also called exceptions.
They cause the Use Case to end up in its Fail End Condition
(Actor's goal is not met), if the failure is not recoverable. If the
failure is recoverable then the extension will eventually end up
into a Success End Condition of the Use Case.
References must be made to any figure or diagram (Use Case
Graphs, Message Sequence Charts)

**SUB-VARIATIONS**

<table>
<thead>
<tr>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>l</td>
</tr>
</tbody>
</table>

Some extensions are too low level to cover at this moment. For
example variations in the data that belong to a certain step
caus ing branching (alternative flows). Handling these variations
is deferred and will be described later in Sub-Use Cases.
Mentioning them in this stage supports tracking of requirements
in the future.

**RELATED INFORMATION**

<table>
<thead>
<tr>
<th>&lt;Use case name&gt;</th>
</tr>
</thead>
</table>

**Priority:**
This defines the priority for implementing the requirement.
High: This requirement should be considered first
during design
Medium: Moderate priority
Low: This requirement should be considered last during design

**Channels to actors**
<e.g. interactive, static files, database, timeouts>

**OPEN ISSUES**
<list of issues awaiting decision affecting this use case >

**Release**
<date or release needed>
R0, R1

...any other
management
information...
<...as needed>
B-2. Policy UCs

B.2.1. Provide Policies

<table>
<thead>
<tr>
<th>USE CASE # PS1</th>
<th>Provide Policies</th>
</tr>
</thead>
</table>

**Goal in Context**
The purpose of “Provide policies” is to allow the WINMAN operator to add/modify/execute/delete a policy in/from the WINMAN system.

**Scope & Level**
Summary level

**Preconditions**
The Winman system is a Policy based system so that the Configuration subsystem, Fault subsystem and the Performance subsystem work according to Policies.

**Success End Condition**
WINMAN will have stored/modified or removed a given policy in the policy repository. The above-mentioned policy might be executed or not.

**Failed End Condition**
WINMAN has not stored/removed/modified policy in the policy repository, nor has the policy started the execution process.

**Primary, Secondary Actors**
WINMAN Operator

**Trigger**
The trigger comes from the WINMAN operator that requests an action from the WINMAN system

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The system determines the type of request. A sub-ordinate policy UC is called upon type of request</td>
</tr>
</tbody>
</table>

**EXTENSIONS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>The required request is not defined Then the system sends a notification message and the UC is ended.</td>
</tr>
</tbody>
</table>

**SUB-VARIATIONS**

<table>
<thead>
<tr>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1b If an add policy request was made then the Add Policies use case is being used</td>
</tr>
<tr>
<td>1c If a modify policy request was made then the Modify Policies use case is being used</td>
</tr>
<tr>
<td>1d If a remove policy request was made then the Remove Policies use case is being used</td>
</tr>
<tr>
<td>1e If an activate policy request was made then the Activate Policies use case is being used</td>
</tr>
</tbody>
</table>

**RELATED INFORMATION**
Provide Policies

**Priority:** High

**Channels to actors**
GUI to WINMAN Operator

**OPEN ISSUES**

**Release**
R0

**...any other**

**Superordinates**
None
Deliverable (D2.1) - Integrated network management requirements and architecture for IP transport services

B.2.2. Provide Policies Report

USE CASE # PS2 Provide Policies Report
Goal in Context Create reports on the policies existing in the system
Scope & Level Secondary Task for WINMAN.
Preconditions The policies are available in the policy repository
Success End Condition The required report is presented
Failed End Condition Such a report cannot be generated – the operator is informed.
Primary Actors WINMAN operator
Secondary Actors
Trigger An operator uses the GUI console to request a report.
DESCRIPTION
Step 1 The user uses a predefined report template asking for policies of a given type or for all of them
Step 2 Create the query according to the way the required information
Step 3 Query the list for the relevant policies into the repository
Step 4 Create the required report and send it to the user
EXTENSIONS Step Branching Action
SUB-VARIATIONS Branching Action

RELATED INFORMATION
Priority: Low
Channels to actors GUI console.
OPEN ISSUES
Release R1
...any other management information...
Superordinates
Subordinates
Requirement traceability

B.2.3. Add Policies

USE CASE # PU1 Add Policies
Goal in Context The purpose of “Add Policies” is to install a new policy in the WINMAN system leaving it ready to be activated.
Scope & Level User level
Preconditions WINMAN must have a policy validation module
<table>
<thead>
<tr>
<th><strong>WINMAN must have an installed repository.</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Success End Condition</strong></td>
</tr>
<tr>
<td>WINMAN will have stored the new policy in the policy repository</td>
</tr>
<tr>
<td><strong>Failed End Condition</strong></td>
</tr>
<tr>
<td>WINMAN will not have stored the new policy in the policy repository</td>
</tr>
<tr>
<td><strong>Primary, Secondary Actors</strong></td>
</tr>
<tr>
<td>WINMAN Operator</td>
</tr>
<tr>
<td><strong>Trigger</strong></td>
</tr>
<tr>
<td>Request from the WINMAN operator</td>
</tr>
<tr>
<td><strong>DESCRIPTION</strong></td>
</tr>
<tr>
<td><strong>Step</strong></td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td><strong>EXTENSIONS</strong></td>
</tr>
<tr>
<td><strong>Step</strong></td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>3a</td>
</tr>
<tr>
<td>4a</td>
</tr>
<tr>
<td>5a</td>
</tr>
<tr>
<td><strong>SUB-VARIATIONS</strong></td>
</tr>
<tr>
<td><strong>Branching Action</strong></td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td><strong>RELATED INFORMATION</strong></td>
</tr>
<tr>
<td><strong>Add Policies</strong></td>
</tr>
<tr>
<td><strong>Priority:</strong></td>
</tr>
<tr>
<td>High</td>
</tr>
<tr>
<td><strong>Channels to actors</strong></td>
</tr>
<tr>
<td>GUI to WINMAN Operator</td>
</tr>
<tr>
<td><strong>OPEN ISSUES</strong></td>
</tr>
<tr>
<td><strong>Release</strong></td>
</tr>
<tr>
<td>R0</td>
</tr>
<tr>
<td><strong>...any other management information...</strong></td>
</tr>
<tr>
<td><strong>Superordinates</strong></td>
</tr>
<tr>
<td>Provide Policies</td>
</tr>
<tr>
<td><strong>Subordinates</strong></td>
</tr>
<tr>
<td>Check Policy</td>
</tr>
<tr>
<td><strong>Requirements tracing</strong></td>
</tr>
<tr>
<td>WIN_CM_121</td>
</tr>
<tr>
<td>WIN_FM_007</td>
</tr>
<tr>
<td>WIN_FM_008</td>
</tr>
<tr>
<td>WIN_FM_009</td>
</tr>
<tr>
<td>WIN_FM_010</td>
</tr>
<tr>
<td>WIN_PM_002</td>
</tr>
<tr>
<td>WIN_PM_009</td>
</tr>
<tr>
<td><strong>Attachments</strong></td>
</tr>
<tr>
<td>None.</td>
</tr>
</tbody>
</table>

### B.2.4. Modify Policies
USE CASE # PU2  Modify policies

Goal in Context  The WINMAN operator modifies a policy

Scope & Level  User level

Preconditions  WINMAN must have a policy validation module
               The policy to be modified must be in the repository

Success End Condition  Policy has been modified.

Failed End Condition  Policy has not been modified.

Primary, Secondary Actors  WINMAN Operator.

Trigger  The trigger comes from WINMAN operator that asks to update a policy.

DESCRIPTION

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The WINMAN Operator sends the identifier of the policy to be modified</td>
</tr>
<tr>
<td>2</td>
<td>The WINMAN system finds the requested policy and displays it for update in the GUI console. The policy is modified.</td>
</tr>
<tr>
<td>3</td>
<td>The policy is syntactically and semantically checked using the “Check Policy” use case</td>
</tr>
<tr>
<td>4</td>
<td>The policy is checked for consistency against previously installed policies using the “Check Policy” use case</td>
</tr>
<tr>
<td>5</td>
<td>The policy is stored in the policy repository instead of the policy that was the subject of modification with the time stamp of the change and the user name</td>
</tr>
</tbody>
</table>

EXTENSIONS

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>2a</td>
<td>The WINMAN system couldn’t find the policy that was requested by the operator. A notification is returned to the operator.</td>
</tr>
<tr>
<td>3a</td>
<td>If the policy is not valid an error message is issued to the WINMAN operator and the modify policy process is aborted.</td>
</tr>
<tr>
<td>4a</td>
<td>If the policy is in conflict with others an error message is sent back to the WINMAN operator and the process is aborted.</td>
</tr>
</tbody>
</table>

SUB-VARIATIONS

<table>
<thead>
<tr>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

RELATED INFORMATION  Modify Policies

Priority:  High

Channels to actors  GUI to WINMAN Operator

OPEN ISSUES  How are we updating a policy that is now running? In our opinion the update of an already instantiated policy (“running policy”) should not be allowed.

Release  R0

...any other management information...

Superordinates  Provide Policies

Subordinates  Check Policy

Find Policy

Requirements tracing  WIN_CM_121
                      WIN_FM_007
                      WIN_FM_008
                      WIN_FM_009
                      WIN_FM_010
                      WIN_PM_002
### B.2.5. Activate Policies

**USE CASE # PU3**  
Activate Policies

**Goal in Context**  
The purpose of “Activate Policies” is to instantiate a certain (set of) policies in the appropriate system components, leaving it ready to be used by the Execute Policies UC.

**Scope & Level**  
User level

**Preconditions**  
The policy to be activated must be available into the policy repository

**Success End Condition**  
The policy is properly instantiated

**Failed End Condition**  
The policy is not instantiated.

**Primary, Secondary Actors**  
WINMAN Operator

**Trigger**  
WINMAN operator. Scheduler.

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The WINMAN system receives a triggering event from one of the two above-mentioned entities. This event must contain the identifier of the policy to be instantiated.</td>
</tr>
<tr>
<td>2</td>
<td>The WINMAN system finds the policy to be activated and retrieves it from the repository and instantiates it in the system</td>
</tr>
</tbody>
</table>

**EXTENSIONS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>2a</td>
<td>The policy is not found or the system is not ready to admit the policy instance. Then an error message is sent to the WINMAN operator with the cause of the failure.</td>
</tr>
</tbody>
</table>

**SUB-VARIATIONS**

<table>
<thead>
<tr>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 &lt;list of variation a&gt;</td>
</tr>
</tbody>
</table>

**RELATED INFORMATION**  
Activate Policies

**Priority:** High

**Channels to actors**  
GUI to WINMAN Operator

**OPEN ISSUES**  
...any other management information...

**Superordinates**  
Provide Policies

**Subordinates**  
Find Policy

**Requirements tracing**

- WIN_CM_121
- WIN_FM_007
- WIN_FM_008
- WIN_FM_009
- WIN_FM_010
- WIN_PM_002
- WIN_PM_009

**Attachments**  
None.

### B.2.6. Execute Policies

**USE CASE # PU4**  
Execute Policies
<table>
<thead>
<tr>
<th><strong>Goal in Context</strong></th>
<th>The purpose of “Execute Policies” is to carry out actions according to an instantiated policy</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Scope &amp; Level</strong></td>
<td>User level</td>
</tr>
<tr>
<td><strong>Preconditions</strong></td>
<td>The policy to be executed must be already instantiated</td>
</tr>
<tr>
<td><strong>Success End</strong></td>
<td>WINMAN system has executed an action in accordance to the policy.</td>
</tr>
<tr>
<td><strong>Failed End</strong></td>
<td>WINMAN system is unable to carry out an action according to the policy.</td>
</tr>
<tr>
<td><strong>Primary, Secondary Actors</strong></td>
<td>Any EMS, the SMS, the WO</td>
</tr>
<tr>
<td><strong>Trigger</strong></td>
<td>SMS, EMS Request from:</td>
</tr>
<tr>
<td></td>
<td>Network Provisioning domain.</td>
</tr>
<tr>
<td></td>
<td>Network Data Management domain.</td>
</tr>
<tr>
<td></td>
<td>Network Maintenance &amp; Restoration domain.</td>
</tr>
<tr>
<td><strong>DESCRIPTION</strong></td>
<td><strong>Step</strong></td>
</tr>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>3</td>
</tr>
<tr>
<td><strong>EXTENSIONS</strong></td>
<td><strong>Step</strong></td>
</tr>
<tr>
<td></td>
<td>2a</td>
</tr>
<tr>
<td></td>
<td>3a</td>
</tr>
<tr>
<td><strong>SUB- VARIATIONS</strong></td>
<td><strong>Branching Action</strong></td>
</tr>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td><strong>RELATED INFORMATION</strong></td>
<td>Execute Policies</td>
</tr>
<tr>
<td><strong>Priority:</strong></td>
<td>High</td>
</tr>
<tr>
<td><strong>Channels to actors</strong></td>
<td>Northbound interface to the SMS</td>
</tr>
<tr>
<td></td>
<td>Southbound interface to the EMSs</td>
</tr>
<tr>
<td><strong>OPEN ISSUES</strong></td>
<td>Release R0</td>
</tr>
<tr>
<td></td>
<td>...any other management information...</td>
</tr>
<tr>
<td><strong>Superordinates</strong></td>
<td>Verify CS Request</td>
</tr>
<tr>
<td></td>
<td>Revert Network</td>
</tr>
<tr>
<td></td>
<td>Recover Network</td>
</tr>
<tr>
<td></td>
<td>Open Alarm</td>
</tr>
<tr>
<td></td>
<td>Identify Root Cause</td>
</tr>
<tr>
<td></td>
<td>Process Performance Measurements</td>
</tr>
<tr>
<td></td>
<td>Calculate Aggregations &amp; Statistics</td>
</tr>
<tr>
<td><strong>Subordinates</strong></td>
<td>WIN_CM_121</td>
</tr>
<tr>
<td></td>
<td>WIN_FM_007</td>
</tr>
<tr>
<td></td>
<td>WIN_FM_008</td>
</tr>
<tr>
<td></td>
<td>WIN_FM_009</td>
</tr>
<tr>
<td></td>
<td>WIN_PM_002</td>
</tr>
<tr>
<td></td>
<td>WIN_PM_009</td>
</tr>
</tbody>
</table>
B.2.7. Remove Policies

<table>
<thead>
<tr>
<th>USE CASE # PU5</th>
<th>Remove policies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal in Context</td>
<td>This use case allows the WINMAN operator to remove a policy because it will be no longer used</td>
</tr>
<tr>
<td>Scope &amp; Level</td>
<td>User level</td>
</tr>
</tbody>
</table>
| Preconditions | The policy to be removed is in the repository  
The system must have a policy validation module |
| Success End Condition | Policy has been removed from the repository. |
| Failed End Condition | Policy has not been removed |
| Primary, Secondary Actors | WINMAN Operator. |
| Trigger | The trigger comes from WINMAN operator that asks to delete a policy. |

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The WINMAN Operator sends the identifier of the policy to be removed.</td>
</tr>
<tr>
<td>2</td>
<td>The WINMAN system finds the requested policy and displays it for deletion in the GUI console.</td>
</tr>
<tr>
<td>3</td>
<td>The WINMAN system will check for interdependencies between the policy to be removed and the remaining policies</td>
</tr>
<tr>
<td>4</td>
<td>The WINMAN system removes the policy from the repository.</td>
</tr>
</tbody>
</table>

**EXTENSIONS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>2a</td>
<td>If the policy is not found an error message is send to the requestor.</td>
</tr>
<tr>
<td>3a</td>
<td>If interdependencies with other system policies exist, the WINMAN operator will be notified and the policy will not be removed</td>
</tr>
</tbody>
</table>

**SUB-VARIATIONS**

<table>
<thead>
<tr>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

**RELATED INFORMATION**

- Remove Policies
- Priority: High
- Channels to actors: GUI to WINMAN Operator

**OPEN ISSUES**

- R0
- ...any other management information...

**Superordinates**

- Provide Policies

**Subordinates**

- Find policy.
- Check policy.

**Requirements tracing**

- WIN_CM_121
- WIN_FM_007
- WIN_FM_008
- WIN_FM_009
- WIN_FM_010
- WIN_PM_002
- WIN_PM_009

**Attachments**

- None.

B.2.8. Check Policy
USE CASE # PA1  
Check Policy

Goal in Context  
The purpose of this use case is to validate the data types of the terms of the specified policy rule and interpret its semantics, ensuring that the construction, the conditions and the actions of the policy rule make sense. The policies are validated according to the category where they belong to. The check process also entails the verification that the policy is not in conflict with other policies.

Scope & Level  
Atomic level

Preconditions
WINMAN has policy validation and policy conflict detection rules. The Metapolicy is active

Success End Condition
The policy is valid and free of conflict.

Failed End Condition
The policy is invalid or with conflict

Primary, Secondary Actors
WO, SMS

Trigger
Add / Modify / Remove Policies

DESCRIPTION

Step | Action
--- | ---
1 | The parameters in the policy are checked and completed with the default values if absent.
2 | The policy is syntactically and semantically checked by means of the meta-policies
3 | The policy is checked against conflicts with other policies by means of the meta-policies.

EXTENSIONS

Step | Branching Action
--- | ---
2a | If the policy is not valid a message is issued and the new policy rejected
3a | If the policy is in conflict a message is issued and the policy is rejected

SUB-VARIATIONS

Branching Action

1 | <list of variation s>

RELATED INFORMATION

Check Policy
Priority: High
Channels to actors
OPEN ISSUES
Release R0
...any other management information...

Superordinates
Add policies. Modify Policies. Remove Policies

Subordinates
None
Requirements tracing
Attachments None.

B.2.9. Find Policy

USE CASE # PA2  
Find Policy

Goal in Context
The purpose of this use case is to find in the repository the policy required.

Scope & Level  
Atomic level
WINMAN has a repository and a location service.

The policy will be found in the repository.

WINMAN has not found the policy in the repository.

WO, SMS

WO, SMS

1 The WINMAN checks the repository for an existing policy according to its name and objects.

2 The WINMAN System locates the policy to be retrieved from the repository

3 The policy is retrieved and ready to be activated, updated or deleted.

1a If policy to be retrieved is not found, an error message will be sent to the WINMAN operator

1b More then one policy is found that answers the required query, an error message will be sent to the WINMAN operator

SUB-VARIATIONS Branching Action

1 <list of variation s>

Find Policy.

High

GUI to WINMAN Operator

R0


None

None.

B-3. Scheduler UCs

B.3.1. Schedule Task

Schedule Task

A task is added to the task list of the scheduler

User level for WINMAN

The WINMAN scheduler is operational. WINMAN holds all necessary data about the managed networks and policy rules.

Requested task is added to the task list of the WINMAN scheduler, and can be viewed.

Requested task is not added to the task list of the WINMAN scheduler

WINMAN Operator

None
### Trigger
Caller use case or actor tries to schedule a task

#### DESCRIPTION

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Schedule task request received</td>
</tr>
<tr>
<td>2</td>
<td>Received schedule task request is validated on policies and properties, as mentioned at 4.3.2.2.</td>
</tr>
<tr>
<td>3</td>
<td>Add new task to task list, taking policies into account, of scheduler and use task type specific parameters from request and missing parameters from task type definition.</td>
</tr>
<tr>
<td>4</td>
<td>Notify caller use case that task is added successfully</td>
</tr>
</tbody>
</table>

#### EXTENSIONS

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>2a</td>
<td>Schedule task request is invalid. Notify calling Use Case that Schedule task request is invalid. Probable causes: request syntax is invalid policy conflict</td>
</tr>
<tr>
<td>3a</td>
<td>Error occurred while adding the task, task cannot be added due to policy. Notify calling Use Case that Schedule task failed.</td>
</tr>
</tbody>
</table>

#### RELATED INFORMATION

**Schedule Task**

**Priority:** High

**Channels to actors**

**OPEN ISSUES**

Release: R0

...any other management information...

**Superordinates**

Provide ICS

Modify ICS

Delete ICS

Provide Policies

**Subordinates**

None

**Requirement Traceability**

WIN_CM_104

WIN_CM_114

WIN_CM_115

WIN_CM_116

WIN_CM_118

WIN_FM_027

WIN_FM_028

### B.3.2. Delete Task

#### USE CASE # SU2
Delete Task

#### Goal in Context
A task is deleted from the task list of the scheduler.

#### Scope & Level
User level for WINMAN

#### Preconditions
The WINMAN scheduler is operational, the different types of tasks are defined. WINMAN holds all necessary data about the managed networks and policy rules

#### Success End Condition
Requested task is deleted from the task list of the WINMAN scheduler

#### Failed End Condition
Requested task cannot be deleted from task list of the WINMAN scheduler.

#### Primary Actors
WINMAN Operator

#### Secondary Actors
None
### Trigger

Caller use case tries to delete a scheduled task from the task list of the scheduler.

<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>Delete task request received</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Received delete task request is validated on ownership, policies and properties, as mentioned at 4.3.2.2.</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>One Task or group of applicable tasks is deleted in the task list of the WINMAN scheduler</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Notify caller use case that task is deleted successfully</td>
</tr>
</tbody>
</table>

### EXTENSIONS

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>2a</td>
<td>Delete task request is invalid. Notify calling Use Case/actor that delete task request is invalid. Probable causes: request syntax is invalid ownership violation task is in progress policy conflict</td>
</tr>
<tr>
<td>3a</td>
<td>Error occurred while the task is deleted. Notify calling Use Case/actor that delete task failed.</td>
</tr>
</tbody>
</table>

### SUB-VARIATIONS

### RELATED INFORMATION

Delete Task

**Priority:** High

**Channels to actors**

**OPEN ISSUES**

Release: R0

...any other management information...

**Superordinates**

Modify ICS

Delete ICS

**Subordinates**

None

**Requirement**

WIN_CM_109

WIN_FM_027

**Traceability**

### B.3.3. Modify Task

**USE CASE # SU3**

Modify Task

**Goal in Context**

A task is modified in the task list of the scheduler

**Scope & Level**

User level for WINMAN

**Preconditions**

The WINMAN scheduler is operational, the different types of tasks are defined. WINMAN holds all necessary data about the managed networks and policy rules

**Success End Condition**

Requested task is modified in the task list of the WINMAN scheduler

**Failed End Condition**

Requested task is not modified in the task list of the WINMAN scheduler

**Primary Actors**

WINMAN Operator

**Secondary Actors**

None

**Trigger**

Caller use case tries to modify a scheduled task

<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>Modify task request received</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Received modify task request is validated on ownership, policies and properties, as mentioned at 4.3.2.2.</td>
</tr>
</tbody>
</table>
One Task or group of applicable tasks is Modified in the task list of the WINMAN scheduler.

Notify caller use case that task is Modified successfully

EXTENSIONS

Step | Branching Action
--- | ---
2a | Modify task request is invalid. Notify calling Use Case that modify task request is invalid. Probable causes:
request syntax is invalid
ownership violation
task is in progress
policy conflict

3a | Error occurred while the task is modified. Notify calling Use Case that Modify task failed.

SUB-VARIATIONS

Branching Action

RELATED INFORMATION

Modify Task

Priority: High

Channels to actors

OPEN ISSUES

Release
R0

Superordinates
Modify ICS
Delete ICS

Subordinates
None

Requirement Traceability
WIN_CM_109
WIN_CM_114
WIN_CM_116
WIN_CM_118
WIN_FM_027
WIN_FM_028

B-4. Network Provisioning UCs

B.4.1. Provide ICS UC

B.4.1.1 Provide ICS-SMS

USE CASE # NS1
Provide ICS (SMS)

Goal in Context
The Service Management System requests from the WINMAN system to provide an ICS.

Level
Summary level

Preconditions
The WINMAN system is ready for providing an ICS. WINMAN holds all necessary data about the managed networks and policy rules

Success End Condition
The ICS is in-effect, i.e., the requested end-to-end ICS is operational, and the resources are allocated. A message is sent to the SMS.

Failed End Condition
The ICS is not delivered; the resources are not allocated and thus free for other ICSs. A failure message is sent to SMS.

Primary Actors
SMS system

Secondary Actors
IP EMS, WDM EMS

Trigger
SMS sends request to WINMAN

DESCRIPTION

Step | Action
--- | ---
1 | The WINMAN system receives a request from the SMS system
to provide an ICS. This request includes the attributes mentioned in 4.3.3.4

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>2a</td>
<td>Invalid attributes, or Invalid parameter values, or Request does not comply with Policy Rules</td>
</tr>
<tr>
<td>2a1</td>
<td>WINMAN returns request denied message to SMS</td>
</tr>
<tr>
<td>3a</td>
<td>WINMAN is unable to create the required network connection(s)</td>
</tr>
<tr>
<td>3a1</td>
<td>WINMAN returns create fail message to SMS</td>
</tr>
</tbody>
</table>

**EXTENSIONS**

**Step**

**Branching Action**

**2a**

Invalid attributes, or
Invalid parameter values, or
Request does not comply with Policy Rules

**2a1**

WINMAN returns request denied message to SMS
Failure end condition reached; use case ends.

**3a**

WINMAN is unable to create the required network connection(s)

**3a1**

WINMAN returns create fail message to SMS
Failure end condition reached; use case ends.

<table>
<thead>
<tr>
<th>SUB-VARIATIONS</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Scheduling (yes/no) Service class (QoS parameters set)</td>
</tr>
</tbody>
</table>

**RELATED INFORMATION**

<table>
<thead>
<tr>
<th>Provide ICS</th>
</tr>
</thead>
</table>

**Priority:**

High

**Channels to actors**

Northbound interface to SMS
Southbound interfaces to EMSs

**OPEN ISSUES**

The failure end condition should include the reason of failure
No manual routing in failure cases

**Release**

R0

...any other management information...

<table>
<thead>
<tr>
<th>Superordinates</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Subordinates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Verify CS Request</td>
</tr>
<tr>
<td>Create CS</td>
</tr>
<tr>
<td>Implement CS</td>
</tr>
<tr>
<td>Activate CS</td>
</tr>
<tr>
<td>Schedule Task</td>
</tr>
</tbody>
</table>

**Requirement Traceability**

| WIN_CM_101 |
| WIN_CM_102 |
| WIN_CM_103 |
| WIN_CM_104 |
| WIN_CM_105 |
| WIN_CM_106 |
| WIN_CM_111 |
| WIN_CM_112 |
B.4.1.2 Provide ICS-WO

<table>
<thead>
<tr>
<th>USE CASE # NS2</th>
<th>Provide ICS (WINMAN Operator)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Goal in Context</strong></td>
<td>The WINMAN Operator requests the WINMAN system to provide an ICS.</td>
</tr>
<tr>
<td><strong>Level</strong></td>
<td>Summary</td>
</tr>
<tr>
<td><strong>Preconditions</strong></td>
<td>The WINMAN system is ready for providing an ICS. WINMAN holds all necessary data about the managed networks and policy rules</td>
</tr>
<tr>
<td><strong>Success End Condition</strong></td>
<td>The ICS is in-effect, i.e., the requested end-to-end ICS is operational, and the resources are allocated. A message is sent to the Operator.</td>
</tr>
<tr>
<td><strong>Failed End Condition</strong></td>
<td>The ICS is not delivered; the resources are not allocated and thus free for other ICSs. A failure message is sent to the Operator.</td>
</tr>
<tr>
<td><strong>Primary Actors</strong></td>
<td>WINMAN Operator</td>
</tr>
<tr>
<td><strong>Secondary Actors</strong></td>
<td>IP EMS, WDM EMS</td>
</tr>
<tr>
<td><strong>Trigger</strong></td>
<td>Operator sends request to WINMAN</td>
</tr>
</tbody>
</table>

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The WINMAN system receives a request from the Operator for the creation of an ICS. This request includes the attributes mentioned in 4.3.3.4</td>
</tr>
<tr>
<td>2</td>
<td>WINMAN verifies the ICS request: The attributes and the values of their parameters Against the Policy Rules This is done by the <em>Verify CS request</em> use case</td>
</tr>
<tr>
<td>3</td>
<td>WINMAN creates the ICS according to applicable policies. This is done by the <em>Create CS</em> use case</td>
</tr>
<tr>
<td>4</td>
<td>WINMAN schedules all Implement/activate/de-activate/Release ICS tasks, taking the policy rules into account, for the entire ICS, using <em>Schedule Task</em>.</td>
</tr>
<tr>
<td>5</td>
<td>WINMAN sends a message to SMS to notify that the ICS is operational</td>
</tr>
</tbody>
</table>

**EXTENSIONS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>2a</td>
<td>Invalid attributes, or Invalid parameter values, or Request does not comply with Policy Rules 2a1: WINMAN returns request denied message to Operator Failure end condition reached; use case ends.</td>
</tr>
<tr>
<td>3a</td>
<td>WINMAN is unable to create the required network connection(s) 3a1: WINMAN returns create fail message to Operator Failure end condition reached; use case ends.</td>
</tr>
<tr>
<td>4a</td>
<td>Scheduling the tasks fails. Use Case ends.</td>
</tr>
</tbody>
</table>

**SUB-**

| Branching Action |
VARIATIONS

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Scheduling (yes/no) Service class (QoS parameters set)</td>
</tr>
</tbody>
</table>

RELATED INFORMATION

| Priority: | High |
| Channels to actors | GUI to WINMAN Operator Southbound interfaces to EMSs |

OPEN ISSUES

The failure end condition should include the reason of failure

Release

R0

...any other management information...

Superordinates

None

Subordinates

Verify CS Request Create CS Implement CS Activate CS Schedule Task

Requirement Traceability

WIN CM_101
WIN CM_102
WIN CM_103
WIN CM_104
WIN CM_105
WIN CM_106
WIN CM_111
WIN CM_112
WIN CM_113
WIN CM_114
WIN CM_115
WIN CM_119
WIN CM_120
WIN CM_121
WIN CM_126
WIN CM_127
WIN CM_128

B.4.2. Modify ICS

B.4.2.1 Modify ICS-SMS

USE CASE # NS3

Modify ICS (SMS)

Goal in Context

WINMAN receives a request through the northbound interface to modify an existing IP connectivity service.

Scope & Level

Summary Level

Preconditions

A Service Management System exists and is able to communicate with WINMAN through the northbound interface to request the modification of an existing IP connectivity service. The WINMAN system is ready for service modifications and holds all necessary data about the configured IP and WDM networks and network policies rules.

Success End Condition

The IP connectivity service parameters are changed and the modified service is operational. A notification has been sent to the SMS.

Failed End

The service has not been modified.
<table>
<thead>
<tr>
<th>Condition</th>
<th>A failure notification has been sent to the SMS.</th>
</tr>
</thead>
</table>
| Primary, Secondary Actors | SMS  
IP EMS and WDM EMS |
| Trigger | WINMAN receives a request to modify an IP connectivity service. |
| DESCRIPTION | **Step** | **Action** |
| 1 | The SMS requests an IP connectivity service to be modified. The request shall include the IP connectivity service parameters to be modified, and the new value. This request includes the attributes mentioned in 4.3.3.4 |
| 2 | WINMAN validates the correctness of the parameters included in the request. WINMAN validates the request against system policies.  
This step uses to the *Verify CS Request UC* |
| 3 | WINMAN calculates the network changes needed to meet the service modifications and the effect that these changes can have over other services. |
| 4 | WINMAN sends commands to the IP EMS system through the southbound interface to carry out the appropriate actions on the network. WINMAN will be responsible to planning the actions over the network in order to minimize the effect on the service. In this sense, the Modify can be service Affecting (SA) and Non Service Affecting (NSA) with respect to the ICS, as appears in the description of this Use Case.  
This step can use the "Implement CS" and "Release CS" sub-use cases to modify the network resources as needed. |
| 5 | WINMAN checks that the network changes have been carried out properly, processing the messages received from the IP EMS. |
| 6 | WINMAN updates the internal information and send a message to notify that the service |
| EXTENSIONS | **Step** | **Branching Action** |
| 2a | If the validation fails a failure condition is reached. The use case ends. |
| 3a | If not enough resources are identified to satisfy the connectivity request in the IP connectivity layer without affecting other services, the system calculates the network reconfiguration actions to be carried out, in both the IP and the WDM networks, to fulfil the request.  
If after this step no resources are found a failure condition is reached. The use case ends. |
| 3b | If a service interruption is needed in order to implement the changes, WINMAN will send a message. |
| 4ª | If the IP connectivity service needs to be interrupted then the De-activate CS, Release CS, Remove CS, Create CS, Implement CS, Activate CS UCs may need to be used. |
| 5ª | If the actions to be carried out on the network fail, a failure condition is reached. The use case ends. |
| RELATED INFORMATION | Modify ICS |
| Priority: | High |
| Channels to actors | Northbound interface to SMS  
Southbound interfaces to EMSs |
**OPEN ISSUES**

OPEN ISSUES:
- How to calculate the network changes needed.
- How to calculate how the change effects over the established services and define the policies to apply.

<table>
<thead>
<tr>
<th>Release</th>
<th>R0</th>
</tr>
</thead>
</table>

**...any other management information...**

<table>
<thead>
<tr>
<th>Superordinates</th>
<th>Recover Network</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Subordinates</th>
<th>Verify CS Request</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Create CS</td>
</tr>
<tr>
<td></td>
<td>Implement CS</td>
</tr>
<tr>
<td></td>
<td>Activate CS</td>
</tr>
<tr>
<td></td>
<td>De-activate CS</td>
</tr>
<tr>
<td></td>
<td>Release CS</td>
</tr>
<tr>
<td></td>
<td>Remove CS</td>
</tr>
<tr>
<td></td>
<td>Schedule Task</td>
</tr>
<tr>
<td></td>
<td>Modify Task</td>
</tr>
<tr>
<td></td>
<td>Delete Task</td>
</tr>
</tbody>
</table>

**Requirement Traceability**

WIN_CM_101
WIN_CM_102
WIN_CM_103
WIN_CM_104
WIN_CM_105
WIN_CM_106
WIN_CM_109
WIN_CM_111
WIN_CM_112
WIN_CM_113
WIN_CM_114
WIN_CM_115
WIN_CM_116
WIN_CM_117
WIN_CM_118
WIN_CM_119
WIN_CM_120
WIN_CM_121
WIN_CM_126
WIN_CM_127
WIN_CM_129
WIN_SM_001
WIN_SM_002

**B.4.2.2 Modify ICS-WO**

<table>
<thead>
<tr>
<th>USE CASE # NS4</th>
<th>Modify service (WINMAN operator)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Goal in Context</strong></td>
<td>WINMAN receives a request from the operator through the GUI interface to modify an existing IP connectivity service.</td>
</tr>
<tr>
<td><strong>Scope &amp; Level</strong></td>
<td>Summary Level</td>
</tr>
<tr>
<td><strong>Preconditions</strong></td>
<td>An option to modify a service is available for WINMAN operator in the GUI. The WINMAN system is ready for service modifications and holds all necessary data about the configured IP and WDM networks and network policies rules.</td>
</tr>
<tr>
<td><strong>Success End Condition</strong></td>
<td>The IP connectivity service parameters are changed and the modified service is operational. A notification has been sent to WINMAN operator.</td>
</tr>
<tr>
<td><strong>Failed End</strong></td>
<td>The service has not been modified.</td>
</tr>
<tr>
<td>Condition</td>
<td>A failure notification has been sent to WINMAN operator.</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Primary, Secondary Actors | WINMAN operator  
                         | IP EMS and WDM EMS                                      |
| Trigger            | WINMAN receives a request to modify an IP connectivity service. |

<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>The WINMAN operator requests an IP connectivity service to be modified. The request shall include the IP connectivity service parameters to be modified, and the new value. This request includes the attributes mentioned in 4.3.3.4</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>WINMAN validates the correctness of the parameters included in the request. WINMAN validates the request against system policies. This step uses the &quot;Verify CS Request&quot; use case</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>WINMAN calculates the network changes needed to meet the service modifications and the effect that these changes can have over other services.</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>WINMAN sends commands to the IP EMS system through the southbound interface to carry out the appropriate actions on the network. WINMAN will be responsible of planning the actions over the network in order to minimize the effect on the service. In this sense, the Modify can be service Affecting (SA) and Non Service Affecting (NSA) with respect to the ICS, as appears in the description of this Use Case. This step can use the &quot;Implement CS&quot; and &quot;Release CS&quot; sub-use cases to modify the network resources as needed.</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>WINMAN checks that the network changes have been carried out properly, processing the messages received from the IP EMS.</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>WINMAN updates the internal information and sends a message to WINMAN operator to notify that the service has been modify.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>EXTENSIONS</th>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2ª</td>
<td>If the validation fails a failure condition is reached. The use case ends.</td>
</tr>
<tr>
<td></td>
<td>3a</td>
<td>If not enough resources are identified to satisfy the connectivity request in the IP connectivity layer without affecting other services, the system calculates the network reconfiguration actions to be carried out, in both the IP and the WDM networks, to fulfil the request. If after this step no resources are found a failure condition is reached. The use case ends.</td>
</tr>
<tr>
<td></td>
<td>3b</td>
<td>If a service interruption is needed in order to implement the changes, WINMAN will send a message.</td>
</tr>
<tr>
<td></td>
<td>4a</td>
<td>If the IP connectivity service needs to be interrupted then the de-activate, release, remove, create, implement, activate use cases need to be use.</td>
</tr>
<tr>
<td></td>
<td>5a</td>
<td>If the actions to be carried out on the network fail, a failure condition is reached. The use case ends.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SUB-VARIATIONS</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3c</td>
</tr>
<tr>
<td></td>
<td>3d</td>
</tr>
</tbody>
</table>
### B.4.3. Delete ICS

#### B.4.3.1 Delete ICS-SMS

<table>
<thead>
<tr>
<th>USE CASE # NS5</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NS5</td>
<td>Delete ICS (SMS)</td>
</tr>
</tbody>
</table>

**Goal in Context**
The Service Management System requests from the WINMAN system the deletion of an ICS. It is assumed that an ICS is de-activated and
The WINMAN system is ready to receive the request. WINMAN holds all necessary data about the managed networks and policy rules.

The ICS is removed, and all the resources are released and available for other ICSs. A message is sent to the SMS.

The ICS is not removed; the resources are not released and thus not free for other ICSs. A failure message is sent to SMS.

SMS system

IP EMS and WDM EMS

SMS sends request to WINMAN

The WINMAN system receives a request from the SMS system for the deletion of an ICS. This request includes the following attributes:

ICS ID

WINMAN verifies the request, taking applicable policies into account:
ICS ID

This is done by the Verify CS request use case

WINMAN removes possible future sessions of this ICS, by using "Delete Task" to remove all tasks involved in future sessions of this ICS.

WINMAN schedules task for removing the ICS.

WINMAN returns message to notify SMS system that the ICS has been deleted successfully

Invalid attributes, or
Invalid parameter values

2a1: WINMAN returns request denied message to SMS
Failure end condition reached; use case ends

Scheduling the tasks fails. Use Case ends, WINMAN returns remove fail message to SMS

3a

Remove process fails

4a1: WINMAN returns remove fail message to SMS
Failure end condition reached; use case ends

If there is a session on-going of the concerning ICS, WINMAN uses "Schedule Task" to have "Remove CS" to be triggered when the current session is over.

If there is no session on-going of the concerning ICS, WINMAN uses "Schedule Task" to have "Remove CS" to be triggered immediately.

Delete ICS

High

Northbound interface to SMS
Southbound interfaces to EMSs

- scheduler will also trigger Remove CS

R0

...any other management
### B.4.3.2 Delete ICS-WO

<table>
<thead>
<tr>
<th>USE CASE # NS6</th>
<th>Delete ICS (WINMAN Operator)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Goal in Context</strong></td>
<td>The Operator requests from the WINMAN system the deletion of an ICS. It is assumed that an ICS is de-activated and released at the end of a session.</td>
</tr>
<tr>
<td><strong>Level</strong></td>
<td>Summary</td>
</tr>
<tr>
<td><strong>Preconditions</strong></td>
<td>The WINMAN system is ready to receive the request. WINMAN holds all necessary data about the managed networks and policy rules.</td>
</tr>
<tr>
<td><strong>Success End Condition</strong></td>
<td>The ICS is removed, and all the resources are released and available for other ICSs. A message is sent to the Operator.</td>
</tr>
<tr>
<td><strong>Failed End Condition</strong></td>
<td>The ICS is not removed; the resources are not released and thus not free for other ICSs. A failure message is sent to Operator.</td>
</tr>
<tr>
<td><strong>Primary Actors</strong></td>
<td>WINMAN Operator</td>
</tr>
<tr>
<td><strong>Secondary Actors</strong></td>
<td>IP EMS and WDM EMS</td>
</tr>
<tr>
<td><strong>Trigger</strong></td>
<td>Operator sends request to WINMAN</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>DESCRIPTION</strong></th>
<th><strong>Step</strong></th>
<th><strong>Action</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>The WINMAN system receives a request from the Operator for the deletion of an ICS. This request includes the following attributes: ICS ID</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>WINMAN verifies the request and taking applicable policies into account: ICS ID</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This is done by the <em>Verify CS Request</em> use case</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>WINMAN removes possible future sessions of this ICS, by using &quot;Delete Task&quot; to remove all tasks involved in future sessions of this ICS.</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>WINMAN schedules task for removing the ICS.</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>WINMAN returns message to notify the Operator that the ICS has been deleted successfully</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>EXTENSIONS</strong></th>
<th><strong>Step</strong></th>
<th><strong>Branching Action</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2a</td>
<td>Invalid attributes, or</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Invalid parameter values</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2a1: WINMAN returns request denied message to the Operator</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Failure end condition reached; use case ends</td>
</tr>
<tr>
<td></td>
<td>3a</td>
<td>Scheduled tasks cannot be removed. Notify Operator</td>
</tr>
<tr>
<td></td>
<td>4a</td>
<td>Scheduling the tasks fails. Use case ends. Notify Operator.</td>
</tr>
</tbody>
</table>
### SUB-VARIATIONS

<table>
<thead>
<tr>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>4a If there is a session on-going of the concerning ICS, WINMAN uses &quot;Schedule Task&quot; to have &quot;Remove CS&quot; to be triggered when the current session is over.</td>
</tr>
<tr>
<td>4b If there is no session on-going of the concerning ICS, WINMAN uses &quot;Schedule Task&quot; to have &quot;Remove CS&quot; to be triggered immediately.</td>
</tr>
</tbody>
</table>

### RELATED INFORMATION

<table>
<thead>
<tr>
<th>Priority:</th>
<th>High</th>
</tr>
</thead>
<tbody>
<tr>
<td>Channels to actors</td>
<td>GUI to WINMAN Operator&lt;br&gt;Southbound interfaces to EMSs</td>
</tr>
<tr>
<td>OPEN ISSUES</td>
<td>- scheduler will also trigger remove ICS</td>
</tr>
<tr>
<td>Release</td>
<td>R0</td>
</tr>
<tr>
<td>...any other management information...</td>
<td></td>
</tr>
<tr>
<td>Superordinates</td>
<td>None</td>
</tr>
<tr>
<td>Subordinates</td>
<td>Delete Task&lt;br&gt;Verify CS Request&lt;br&gt;Remove CS&lt;br&gt;Delete Task&lt;br&gt;Schedule Task</td>
</tr>
<tr>
<td>Requirement Traceability</td>
<td>WIN_CM_109&lt;br&gt;WIN_CM_112&lt;br&gt;WIN_CM_116&lt;br&gt;WIN_CM_118&lt;br&gt;WIN_CM_126&lt;br&gt;WIN_CM_127&lt;br&gt;WIN_CM_128&lt;br&gt;WIN_SM_003</td>
</tr>
</tbody>
</table>

## B.4.4. Verify CS Request

<table>
<thead>
<tr>
<th>USE CASE # NU1</th>
<th>Verify CS Request</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal in Context</td>
<td>Verify the legitimacy of the request (semantics, value ranges) and its compliance to current WINMAN policies</td>
</tr>
<tr>
<td>Scope &amp; Level</td>
<td>User level - Network Provisioning</td>
</tr>
<tr>
<td>Preconditions</td>
<td>A WINMAN system exists and is able to issue CS requests (upon actors triggering) to be verified before processing. A policy for serving requests could be in practice. This policy can either be part of this UC algorithm, or be retrieved dependent on the type of request. Certain types of request and a corresponding list of attributes to be processed dependent on the type of request, should be available. Appropriate ranges of values of every possible attribute. An algorithm to efficiently check the legitimacy of attributes both standalone and combined. A number of WINMAN ICS policies and a procedure to check an attribute combination against these policies.</td>
</tr>
<tr>
<td>Success End Condition</td>
<td>The request is successfully verified regarding both the acceptability of the attribute values (standalone and combined) and compliance to WINMAN policies in effect.</td>
</tr>
<tr>
<td>Failed End Condition</td>
<td>The request is not propagated for further processing. It is either deleted or kept to a list for future use (e.g. wait until policy changes). The requesting entity is notified accordingly.</td>
</tr>
<tr>
<td>Primary Actors</td>
<td>WINMAN Operator, SMS</td>
</tr>
</tbody>
</table>
Secondary Actors

Trigger | A request is issued by the Provide / Modify / Delete ICS UCs.

<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>The requesting user (WO / SMS) triggers the procedure by issuing a request with certain attributes and assigned values. WINMAN should be able to resolve concurrency conflicts via a prioritised scheme. Different serving policies could be used for requests with different priority designation.</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>The request is checked for completeness. According to the type of request (Provide/Modify/Delete) certain fields should be present. The values assigned to attributes are checked for legitimacy against the corresponding legal WINMAN value ranges. This should be a typical out-of-range check. Being semantically acceptable the request may be recorded in the WINMAN system (e.g. request history inventory).</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>The request is validated against current WINMAN policies (Execute Policy UC). The selection of the specific policies for cross-check may be based on the type of the CS request. The check itself might include certain value restrictions for some attributes and / or combinations of values that may not be acceptable. Other rules may also apply.</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>If successful, the sender entity is notified with a success return value. Use case ends.</td>
</tr>
</tbody>
</table>

EXTENSIONS | Step | Branching Action |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2a</td>
<td>If mandatory fields for this type of request are not present, the request is aborted. User notified. Use case ends.</td>
</tr>
<tr>
<td></td>
<td>2b</td>
<td>If a mandatory field for this type of request is out of legal range, the request is aborted. User notified. Use case ends.</td>
</tr>
<tr>
<td></td>
<td>2c</td>
<td>If an optional field for this type of request is out of legal range, a pre-decided default for this field is used. Alternatively the request could be aborted. User entity notified. Use case ends.</td>
</tr>
<tr>
<td></td>
<td>2d</td>
<td>If the request takes more than a predefined time to be evaluated, the user entity is notified accordingly.</td>
</tr>
<tr>
<td></td>
<td>3a</td>
<td>If a individual values or combinations are not acceptable the request is aborted. User entity notified. Use Case ends.</td>
</tr>
<tr>
<td></td>
<td>3b</td>
<td>If the prescribed CS is semantically correct, but cannot be currently accommodated by WINMAN (e.g. network congestion) the request could be: Aborted. User entity notified. Use case ends. Stored for future use. A timer (WINMAN default or provided by the request itself) could be used to time out the request and permanently delete it.</td>
</tr>
</tbody>
</table>

SUB-VARIATIONS | Branching Action |

RELATED INFORMATION

| Priority: | High |
| Channels to actors | WINMAN operator interfaces to the system, WINMAN northbound interfaces. |

OPEN ISSUES

Request Serving Policy including queueing of requests (e.g. FIFO) and serving requests according to priority schemes related to e.g. the type of request. Should the system allow the WO to implement a CS, even in cases where...
the CS request is not successfully processed by the Verify CS UC?
Is the Verify CS Request applicable in all operating modes (e.g. maybe
excluded from manual mode)
What could be the responsibility of this UC in resolving scheduler related
conflicts (time / resources)?
A special field could be reserved in the request, to denote that if the
request is not fulfilled due to inconsistent policy, or insufficient
resources, the request should be stored for a default / designated time
in the queue. This would mean that the scheduling parameters could
be relaxed, or unspecified.
How is the "degree of success" of a request described (req.
WIN_CM_126)?

Release
R0

...any other
management
information...

Superordinates
Provide ICS
Modify ICS
Delete ICS

Subordinates
Execute Policy

Requirement
WIN_COM_008
WIN_CM_101
WIN_CM_102
WIN_CM_103
WIN_CM_104
WIN_CM_105
WIN_CM_109
WIN_CM_114
WIN_CM_126

Traceability

B.4.5. Create CS UC

B.4.5.1 Create CS-SMS

USE CASE # NU2
Create CS (SMS)

Goal in Context
This use case is used by another use case. The CS is designed and the necessary
network resources are reserved.

Scope & Level
User Level

Preconditions
The WINMAN system is ready for CS creation and holds all necessary data about the
configured IP and WDM networks.

Success End Condition
The design of the CS is recorded and the resources are reserved.
A message is sent to the SMS.

Failed End Condition
The system was unable to create the CS with the specified parameters and returns an
error message.

Primary Actors
SMS

Secondary Actors
EMS

Trigger
A summary level use case asks for the reservation of resources to support a service.

DESCRIPTION
Step Action
1 The caller use case requests the creation of a CS between two or more end
points. In the parameters of the request should be mentioned:
CS ID
Connection end points
Scheduling parameters
Policies
Connection directionality
NLA parameters
2 The system computes the IP paths fulfilling the connectivity request.
This a 2 step process:

2.1 - The system finds the different combination of IP links that lead to the destination points, without concerning about the service parameters. This is done in the use case Compute list of IP routes.

2.2 - The system selects the combinations of IP links that satisfy the CS requirements - Traffic Engineering. This is done in the use case Compute IP paths under constraints.

3 The system computes the cost of each one of the IP paths.

4 The system locks (reserves) the network resources used by preferred IP path.

4 The system computes the cost of each one of the IP paths.

5 The system returns the reserved IP path and its cost. A message is sent to the SMS informing successful reservation of resources. Success end condition reached. Use case ends.

EXTENSIONS | Step | Branching Action |
--- | --- | --- |
2a | If the system does not find a combination of IP links that satisfy the connectivity request it will request optical path connectivity. This is done in the use case Compute list of optical trails. The process here should be similar. A few possible arrangements on the network are considered and the resources for them (lambdas) are reserved. The system should also consider the solution of reinforce the IP links (add optical paths - \( \lambda \)) that were considered on the IP calculation and were unable to support the type of service required. The system goes back to the beginning of point 2.

2a1 | The WDM layer can’t create or reinforce any IP link between the end points. The system can’t support the service requested. Failure end Condition reached. Use case ends.

4a | If the system is operating in semi-automatic mode: The system prompts the WINMAN Operator to choose one of the paths from the list of possible IP paths.

SUB-VARIATIONS | Branching Action |
--- | --- |
Semi-automatic
Manual

RELATED INFORMATION
Priority: High

Channels to actors

OPEN ISSUES
Release: R0

...any other management information...

Superordinates Provide CS
Modify CS

Subordinates Compute possible list of IP routes
Compute IP paths under constraints
Compute list of optical trails

Requirement traceability
WIN_COM_001
WIN_CM_101
WIN_CM_102
WIN_CM_104
WIN_CM_105
WIN_CM_111
WIN_CM_114
B.4.5.2 Create CS-WO

**USE CASE # NU3** Create CS (WINMAN Operator)

**Goal in Context** This use case is used directly by the WINMAN Operator. The CS is designed and the necessary network resources are reserved.

**Scope & Level** User Level

**Preconditions** The WINMAN system is ready for service creation and holds all necessary data about the configured IP and WDM networks and network policy rules

**Success End Condition** The design of the CS is recorded and the resources are reserved. A message is sent to the SMS or the WINMAN operator.

**Failed End Condition** The system was unable to create the CS with the specified parameters and, returns an error message.

**Primary Actors** WINMAN Operator

**Secondary Actors** IP-EMS, WDM-EMS

**Trigger** The WINMAN Operator requests the creation of an Optical trail.

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The WINMAN Operator requests the creation of a CS between two or more end points residing at the same layer (IP or WDM). In the parameters of the request should be mentioned: Connectivity Service ID, Connection end points, Scheduling parameters, Policies, Connection directionality, NLA parameters, Mode of Operation.</td>
</tr>
<tr>
<td>2</td>
<td>WINMAN verifies the service request: The attributes and the values of their parameters Against the Policy Rules. This is done by the Verify CS request use case.</td>
</tr>
<tr>
<td>3</td>
<td>If the end points given are IP NE then the request is for IP connectivity. All possible IP routes are calculated by the “Compute list of IP routes” use case. The routes are compared with the requested parameters by the “Compute IP paths under constraints” use case. If the end points are WDM NE then the request is for WDM connectivity. All possible optical trails are calculated by the “Compute list of Optical Trails” use case.</td>
</tr>
<tr>
<td>4</td>
<td>The system computes the cost of each one of the solutions.</td>
</tr>
<tr>
<td>5</td>
<td>The system chooses the cheapest IP path or optical trail that satisfies the requested constraints.</td>
</tr>
<tr>
<td>6</td>
<td>The system returns a message to the WINMAN Operator indicating the reservation made. Success end condition reached. Use case ends.</td>
</tr>
</tbody>
</table>

**EXTENSIONS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>2a</td>
<td>Invalid attributes, or Invalid parameter values, or Request does not comply with Policy Rules. 2a1: WINMAN returns request denied message to WINMAN Operator Failure end condition reached; use case ends.</td>
</tr>
<tr>
<td>3a</td>
<td>Manual mode of operation has been requested. The WINMAN Operator indicates the links that it wants to use to build the optical trail or IP path.</td>
</tr>
</tbody>
</table>
The path does not connect the endpoints or cannot support the parameters. Error message is sent to WINMAN Operator. Failure end condition reached; use case ends.

5a Semi-automatic mode of operation has been requested. The WINMAN system prompts the WINMAN Operator to choose the solution it wants.

### SUB-VARIATIONS

**Branching Action**

### RELATED INFORMATION

| Priority: | High |
| Channels to actors |  |
| OPEN ISSUES |  |
| Release | R0 |
| ...any other management information... |  |
| Superordinates | None |
| Subordinates | Verify CS Request
Compute possible list of IP connectivity paths
Compute end-to-end paths with constraints
Compute list of Optical Trails |

### Requirement traceability

| WIN_COM_001 |
| WIN_CM_101 |
| WIN_CM_102 |
| WIN_CM_104 |
| WIN_CM_105 |
| WIN_CM_111 |
| WIN_CM_114 |
| WIN_CM_119 |
| WIN_CM_120 |
| WIN_CM_121 |
| WIN_CM_122 |
| WIN_CM_128 |
| WIN_CM_129 |

### B.4.6. Implement CS

**USE CASE # NU4**

**Goal in Context**
The scheduler uses this use case when the time for implementing the CS is reached.

**Scope & Level**
User level

**Preconditions**
The CS is in the implemented state. WINMAN holds all necessary data about the managed networks and policy rules

**Success End Condition**
The CS has been implemented.
The state of resources is *Allocated*
The state of the CS is *Implemented*
A notification has been sent to the SMS

**Failed End Condition**
The CS remains in the *Reserved* state
A failure notification has been sent to the SMS

**Primary Actors**
None (Scheduler)

**Secondary Actors**
IP EMS and WDM EMS

**Trigger**
The scheduler starts the implementation process.

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>WINMAN is requested to implement the CS.</td>
</tr>
<tr>
<td></td>
<td>WINMAN requests the involved EMSs to establish and test the</td>
</tr>
</tbody>
</table>
connectivity.

WINMAN changes the state of the CS to *implemented*

WINMAN changes the state of the resources to *allocated*.

WINMAN notifies the SMS c.q operator that the CS is *implemented*.

### EXTENSIONS

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>2a</td>
<td>The communication with EMSs is not possible. A &quot;Communications error&quot; failure message is sent to the SMS.</td>
</tr>
</tbody>
</table>

### SUB-VARIATIONS

<table>
<thead>
<tr>
<th>Branching Action</th>
</tr>
</thead>
</table>

### RELATED INFORMATION

| Priority: | High |
| Channels to actors | Northbound interface to SMS or GUI interface to WINMAN Operator, Southbound interfaces to EMSs |

### OPEN ISSUES

| Release | R0 |
| ...any other management information... |

| Superordinates | Provide ICS, Modify ICS |
| Subordinates | None |

### Requirement Traceability

| WIN_CM_103 |
| WIN_CM_106 |
| WIN_CM_111 |
| WIN_CM_112 |
| WIN_CM_113 |
| WIN_CM_114 |
| WIN_CM_118 |
| WIN_CM_119 |
| WIN_CM_120 |
| WIN_CM_126 |
| WIN_CM_127 |
| WIN_CM_128 |
| WIN_SM_004 |

### B.4.7. Activate CS

#### USE CASE # NU5

| Activate CS |
| Goal in Context | The scheduler uses this use case when the time for activate the CS is reached |
| Scope & Level | User level |
| Preconditions | CS state is *Implemented* |

| Success End Condition | CS activated |
| A notification has been sent to the SMS |
| The state of the CS is *Operational* |

| Failed End Condition | The CS remains in the *Implemented* state |
| A failure notification has been sent to the SMS |

| Primary, Secondary Actors | SMS or WINMAN Operator, IP EMS, WDM EMS |
| Trigger | The scheduler starts the activation process indirectly via the Provision ICS and Modify ICS summary level use cases. |

#### DESCRIPTION

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
</table>
B.4.8. De-Activate CS

**USE CASE # NU6**
De-activate CS

**Goal in Context**
The scheduler uses this use case when the time for deactivation of the CS is reached

**Scope & Level**
User level

**Preconditions**
CS state is operational

**Success End Condition**
CS de-activated  
A notification has been sent to the SMS  
The state of the CS is Implemented

**Failed End Condition**
The CS remains in the Operational state  
A failure notification has been sent to the SMS

**Primary, Secondary Actors**
SMS or WINMAN Operator  
IP EMS, WDM EMS

**Trigger**
The scheduler starts the activation process indirectly via the Delete ICS and Modify ICS summary level use cases.

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>WINMAN is requested to deactivate the CS.</td>
</tr>
<tr>
<td>2</td>
<td>WINMAN requests the involved WDM EMSs to deactivate the optical connectivity.</td>
</tr>
<tr>
<td>3</td>
<td>WINMAN changes the state of the CS to Implemented.</td>
</tr>
<tr>
<td>4</td>
<td>WINMAN notifies the SMS or WINMAN Operator that the CS is Implemented.</td>
</tr>
</tbody>
</table>

**EXTENSIONS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>2a</td>
<td>The communication with the WDM EMSs is not possible. A &quot;Communications error&quot; failure message is sent to the SMS.</td>
</tr>
</tbody>
</table>

**SUB-VARIATIONS**

<table>
<thead>
<tr>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

**RELATED INFORMATION**

<table>
<thead>
<tr>
<th>Activate CS</th>
</tr>
</thead>
</table>

**Priority:** High

**Channels to actors**

| Northbound interface to SMS or GUI interface to WINMAN Operator |
| Southbound interfaces to EMSs |

**OPEN ISSUES**

<table>
<thead>
<tr>
<th>Release</th>
</tr>
</thead>
<tbody>
<tr>
<td>R0</td>
</tr>
</tbody>
</table>

...any other management information...

**Superordinates**

<table>
<thead>
<tr>
<th>Provide ICS, Modify ICS</th>
</tr>
</thead>
</table>

**Subordinates**

None

**Requirements**

| WIN_CMT_115, WIN_CMT_117, WIN_CMT_119 |

**Traceability**
B.4.9. Release CS

**USE CASE # NU7**

**Release CS**

**Goal in Context**
This use case is used by the Delete IP connectivity service use case, in order to de-allocate the resources occupied by a connection.

**Scope & Level**
User level

**Preconditions**
The service is in the implemented state

**Success End Condition**
The IP connectivity service has been released.
The state of resources is reserved.
The state of the service is reserved.
A notification is sent to the SMS.

**Failed End Condition**
The IP connectivity service is still implemented.
A failure notification has been sent to the SMS.

**Primary, Secondary Actors**
WO, SMS
IP EMS and WDM EMS

**Trigger**
WINMAN receives a delete service request or the scheduled time for releasing the service is reached.

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>WINMAN is requested to release a service.</td>
</tr>
<tr>
<td>2</td>
<td>WINMAN request the IP-EMS the removal of the implementation of the IP paths, that is, releases the IP physical resources.</td>
</tr>
<tr>
<td>3</td>
<td>WINMAN changes the state of the service to reserved</td>
</tr>
<tr>
<td>4</td>
<td>If the release operation causes that an optical path does not serve clients anymore, that optical path is released.</td>
</tr>
<tr>
<td>5</td>
<td>WINMAN notifies the SMS that the service is released.</td>
</tr>
</tbody>
</table>

**EXTENSIONS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>2a</td>
<td>The communication with the EMSs is not possible. A &quot;Communications error&quot; failure message is sent to the SMS.</td>
</tr>
<tr>
<td>2b</td>
<td>The IP-EMS fails to remove implementation. Then WINMAN sends a messages and the use case ends.</td>
</tr>
</tbody>
</table>

**SUB-VARIATIONS**

<table>
<thead>
<tr>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
</tr>
</tbody>
</table>
**RELATED INFORMATION**

| Release service |

**Priority:**

| High |

**Channels to actors**

| Northbound interface to SMS |
| Southbound interfaces to EMSs |

**OPEN ISSUES**

| Release |

| R0 |

| ...any other management information... |

| Superordinates |

| Delete ICS |

| Subordinates |

| WIN_CM_107 |
| WIN_CM_109 |
| WIN_CM_110 |
| WIN_CM_111 |
| WIN_CM_112 |
| WIN_CM_116 |
| WIN_CM_117 |
| WIN_CM_118 |
| WIN_CM_119 |
| WIN_CM_123 |
| WIN_CM_126 |
| WIN_CM_127 |
| WIN_CM_129 |
| WIN_SM_003 |
| WIN_SM_004 |

**B.4.10. Remove CS**

| USE CASE # NU8 | Remove CS |

**Goal in Context**

This use case is used by the *Delete ICS* use case, or the WINMAN Operator. The remove of reservations is made only on the WINMAN storage facilities.

**Scope & Level**

User

**Preconditions**

The service is in the reserved state.

**Success End Condition**

The system eliminates all the reservations of resources for the CS on the storage facilities.

A notification has been sent to the SMS.

**Failed End Condition**

The system was unable to eliminate all the reservations made in the storage facilities to the CS. An error message is returned.

A failure notification has been sent to the SMS.

**Primary Actors**

SMS, WINMAN Operator

**Secondary Actors**

SMS

**Trigger**

The Delete CS use case asks for the removal of the reservations made, in the storage facilities for one CS.

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>This use case receives a request to eliminate all the reservations made to an CS.</td>
</tr>
<tr>
<td>2</td>
<td>The design of the CS and all the reservations made in the storage facilities are removed.</td>
</tr>
<tr>
<td>3</td>
<td>The CS is removed from WINMAN WINMAN notifies the SMS that the service has been removed.</td>
</tr>
</tbody>
</table>

**EXTENSIONS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>The CS doesn’t exist. Error message is sent. Failed end condition is reached. Use case ends.</td>
</tr>
</tbody>
</table>

**SUB-VARIATIONS**

Branching Action
B.4.11. Compute list of IP Routes

<table>
<thead>
<tr>
<th>USE CASE # NA1</th>
<th>Compute list of IP routes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal in Context</td>
<td>Calculate all the possible IP routes in the entire network between two end points. The calculation is made without any constraints. The use case just search for routes that provide connectivity between the end points. Routing constraints are left for other use cases. In the end, the use case should return the list of all IP routes that connect the end points.</td>
</tr>
<tr>
<td>Scope &amp; Level</td>
<td>Atomic level</td>
</tr>
<tr>
<td>Preconditions</td>
<td>The WINMAN System is appropriately updated regarding network resources and holds all necessary data and algorithms to compute IP routes between endpoints.</td>
</tr>
<tr>
<td>Success End Condition</td>
<td>A list of all possible routes between the end points (if any exists) is returned.</td>
</tr>
<tr>
<td>Failed End Condition</td>
<td>The list could not be calculated.</td>
</tr>
<tr>
<td>Primary Actors</td>
<td>SMS / WINMAN Operator</td>
</tr>
<tr>
<td>Secondary Actors</td>
<td></td>
</tr>
<tr>
<td>Trigger</td>
<td>A use case asks for a list of all the IP routes between two end points.</td>
</tr>
</tbody>
</table>

**DESCRIPTION**

1. This use case receives a list of two end points.
2. The system locates the two end points (routers) in the IP nodes database. The system selects one of them (origin router).
3. The system finds all the IP links connected to the router. The system marks them all as unused.
4. One of the unused IP links is chosen, marked as used, and the router that it connects to is found.
5. The system goes back to point 3, but now considering the new router found.
6. All the routes found are returned. Success end condition reached. Use case ends.

**EXTENSIONS**

2a. The system was unable to locate one of the end points. Failed end condition reached. Use case ends.

4a. There are no unused links.
   * If this is not the origin router the system goes back to the previously selected router. Step 4 is repeated.
   * If this is the origin router the system jumps to point 6.
4b The router that was found already belongs to the route (loop found). Step 4 is repeated.

4c The router found is the other endpoint. The system adds this route (list of IP links) to the possible routes list. Step 4 is repeated.

SUB-VARIATIONS

Branching Action

RELATED INFORMATION

Priority: High

Channels to actors

OPEN ISSUES

Release R0

...any other management information...

Superordinates Create CS

Subordinates

Requirement traceability WIN_COM_001

B.4.12. Compute IP Paths under Constraints

USE CASE # NA2 Compute IP paths under constraints

Goal in Context Discover, from a list of end-to-end IP routes, which routes can support certain constraints. The examination is made on the links. In the end a list of all the IP routes that satisfy the constraints is returned (if any exists). A list of IP links that do not support the constraints should also be returned.

Scope & Level Atomic level

Preconditions Existence of use cases that may need the calculations made by this use case. The list of constraints and the list of routes to examine are provided by the caller use case. WINMAN have to know the resources use state.

Success End Condition A list of all the routes that support the constraints is returned. A list of all the links that can’t support the constraints is also returned.

Failed End Condition The lists could not be calculated.

Primary Actors SMS / WINMAN Operator

Secondary Actors

Trigger A use case asks for a list of all the IP paths that support a list of constraints.

DESCRIPTION

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>This use case is called to examine a list of IP routes and verify which of them fulfills certain requirements. The requirements admitted are: Scheduling parameters Policies Connection directionality NLA parameters</td>
</tr>
<tr>
<td>2</td>
<td>The system builds a list with all the IP links that are used by the IP routes.</td>
</tr>
<tr>
<td>3</td>
<td>The system compares the resources available on each one of the IP links at the time indicated, with the requirements indicated. The links that can’t support the requirements are marked as rejected IP links. The links forbidden are marked as rejected to. Where the algorithm optimises the resources used to provide the service.</td>
</tr>
<tr>
<td>4</td>
<td>All the IP paths that use rejected links are rejected.</td>
</tr>
<tr>
<td>5</td>
<td>If any Mandatory link was indicated all the paths that do not use it are rejected.</td>
</tr>
</tbody>
</table>
The system returns all the accepted IP paths and all the rejected IP links

**RELATED INFORMATION**

**Priority:** High

**Channels to actors**

**OPEN ISSUES**

**Release** R0

**...any other management information...**

**Superordinates** Create CS

**Subordinates**

**Requirement traceability**

WIN_COM_001
WIN_CM_102
WIN_CM_106
WIN_CM_121

### B.4.13. Compute list of Optical Trails

**USE CASE # NA3** Compute list of optical trails

**Goal in Context** Calculate all the possible Optical trails between two end points. Verify if it is possible to implement them. In the end, the use case should return the list of all Optical trails that connect the end points.

**Scope & Level** Atomic level

**Preconditions** Existence of use cases that may need the calculations made by this use case.

**Success End Condition** A list of all the possible Optical trails between the end points (if any exists) is returned.

**Failed End Condition** The list could not be calculated.

**Primary Actors** SMS / WINMAN Operator

**Secondary Actors**

**Trigger** A use case asks for a list of all possible Optical trails between two end points.

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>This use case receives a list of two end points.</td>
</tr>
<tr>
<td>2</td>
<td>The system locates the two end points (routers) in the WDM nodes database. The system selects one of them (origin node).</td>
</tr>
<tr>
<td>3</td>
<td>The system finds all the WDM links connected to the node. The system marks them all as unused.</td>
</tr>
<tr>
<td>4</td>
<td>One of the unused WDM links is chosen, marked as used, and the node that it connects to is found.</td>
</tr>
<tr>
<td>5</td>
<td>The system goes back to point 3, but now considering the new node found.</td>
</tr>
<tr>
<td>6</td>
<td>All the routes found are checked to see if their implementation is possible.</td>
</tr>
<tr>
<td>7</td>
<td>All the routes that can be implemented are returned. Success end condition reached. Use case ends.</td>
</tr>
</tbody>
</table>

**EXTENSIONS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>2a</td>
<td>The system was unable to locate one of the end points. Failed end condition reached. Use case ends.</td>
</tr>
</tbody>
</table>
| 4a   | There are no unused links. If this is not the origin node the system goes back to the previously
selected router. Step 4 is repeated.
If this is the origin node the system jumps to point 6.

4b The node that was found already belongs to the route (loop found). Step 4 is repeated.

4c The node found is the other endpoint. The system adds this route (list of IP links) to the possible routes list. Step 4 is repeated.

SUB-VARIATIONS

Branching Action

RELATED INFORMATION

Priority: High
Channels to actors
OPEN ISSUES
Release R0
...any other management information...
Superordinates Create CS
Subordinates
Requirement traceability WIN_COM_001

B-5. Network Inventory Management UCs

B.5.1. Register EMS

USE CASE # IS1 Register EMS

Goal in Context EMS sends a register to WINMAN, telling which type of Element Management System it is and also indicating that it is operational to receive commands from WINMAN.

Scope & Level Summary level
Preconditions The WINMAN system is ready to receive the registration. And WINMAN holds all necessary data to handle this registration.
Success End Condition EMS is registered by WINMAN. It is able to receive requests and to send out notifications.
Failed End Condition EMS is not able to register by WINMAN.
Primary Actors EMS
Secondary Actors None
Trigger During the initialisation of the EMS.

DESCRIPTION Step Action
The registration request is validated for valid parameters and against policies.
EMS creates an entity in a location that is available to WINMAN
EMS changes its state in such a way to be able to receive requests from WINMAN

EXTENSIONS

1a Invalid parameters or Refused by policies Use case ends
2a Unable to create Use case ends

SUB-VARIATIONS
RELATED INFORMATION

| Register |

Priority: High

Channels to actors: Southbound interfaces to EMSs

OPEN ISSUES

Release: R0

...any other management information...

Superordinates: None

Subordinates: Configure Network

Requirement Traceability: WIN_CM_129

B.5.2. Un-Register EMS

USE CASE # IS2 UnRegister EMS

Goal in Context: EMS wishes to remove the actual connection with WINMAN.

Scope & Level: Summary level

Preconditions: There is an active communication between EMS and WINMAN.

Success End Condition: EMS has been disconnected from WINMAN, and is not able to receive any requests. WINMAN is aware of the removal of the connection to EMS. The communication between WINMAN and EMS is stopped.

Failed End Condition: EMS is not able to unregister by WINMAN.

Primary Actors: EMS

Secondary Actors: None

Trigger: Shutdown of EMS or due to WINMAN request to unregister

DESCRIPTION

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>EMS informs WINMAN that it wishes to delete the communication.</td>
</tr>
<tr>
<td>2.</td>
<td>EMS frees the entity in a location that is available to WINMAN to see that EMS is not accessible anymore</td>
</tr>
<tr>
<td>3.</td>
<td>EMS changes its state in such a way to be unable to receive requests from WINMAN</td>
</tr>
</tbody>
</table>

EXTENSIONS

SUB-VARIATIONS

RELATED INFORMATION

| UnRegister |

Priority: High

Channels to actors: Southbound interfaces to EMSs

OPEN ISSUES

Release: R0

...any other management information...

Superordinates: None

Subordinates: Configure Network

Requirement Traceability: WIN_CM_129
B.5.3. Notify Network Change

<table>
<thead>
<tr>
<th>USE CASE # IS3</th>
<th>Notify Network Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal in Context</td>
<td>A network change notification is received by WINMAN</td>
</tr>
<tr>
<td>Scope &amp; Level</td>
<td>Summary level</td>
</tr>
<tr>
<td>Preconditions</td>
<td>The WINMAN system is ready to receive the network change notification. WINMAN holds all necessary data about the managed networks and policy rules.</td>
</tr>
<tr>
<td>Success End Condition</td>
<td>Network changes made and consequent actions taken according to the changes indicated by the notification.</td>
</tr>
<tr>
<td>Failed End Condition</td>
<td>Network change not made</td>
</tr>
<tr>
<td>Primary Actors</td>
<td>EMS</td>
</tr>
<tr>
<td>Secondary Actors</td>
<td>None</td>
</tr>
<tr>
<td>Trigger</td>
<td>Network change notification received from EMS</td>
</tr>
</tbody>
</table>

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Notification received from EMS and is validated</td>
</tr>
<tr>
<td></td>
<td>Locate network entity that notification applies to.</td>
</tr>
<tr>
<td></td>
<td>Add/Modify/Remove the network entity and depending network entities.</td>
</tr>
<tr>
<td></td>
<td>Take consequent actions defined by policies, using &quot;Configure Network&quot;, notify WINMAN Operator of the actions taken.</td>
</tr>
</tbody>
</table>

**EXTENSIONS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>Change notification is invalid. Use Case ends.</td>
</tr>
<tr>
<td>2a</td>
<td>Network entity not found. Use Case ends.</td>
</tr>
<tr>
<td>3a</td>
<td>Consequent actions can not be taken. Use Case ends.</td>
</tr>
</tbody>
</table>

**RELATED INFORMATION**

<table>
<thead>
<tr>
<th>Priority:</th>
<th>High</th>
</tr>
</thead>
<tbody>
<tr>
<td>Channels to actors</td>
<td>Southbound interfaces to EMSs</td>
</tr>
<tr>
<td>OPEN ISSUES</td>
<td>R0</td>
</tr>
</tbody>
</table>

**SUB-VARIATIONS**

**RELATED INFORMATION**

| Notify Network Change |

**Priorities**

**Channels to actors**

**OPEN ISSUES**

**Superordinates**

**Subordinates**

**Requirement**

**Traceability**

WIN_CM_129
WIN_CM_203

B.5.4. Configure Network

<table>
<thead>
<tr>
<th>USE CASE # IU1</th>
<th>Configure Network</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal in Context</td>
<td>Modify the Network configuration, by adding / deleting elements and / or modifying element parameters.</td>
</tr>
<tr>
<td>Scope &amp; Level</td>
<td>User level</td>
</tr>
<tr>
<td>Preconditions</td>
<td>An option to modify the network inventory is available for the WO in the GUI. The WINMAN system is ready for inventory modifications and holds all necessary data about the configured elements and network policies rules. All underlying EMSs have been successfully registered to WINMAN. Southbound communication links to these EMSs are activated.</td>
</tr>
</tbody>
</table>

WINMAN-WP2-PTI-020a-D2-1-b1 © WINMAN Consortium
### Success End Condition
The network inventory parameters are changed and the modified service and/or elements are operational. A notification has been sent to the WINMAN operator.

### Failed End Condition
The inventory has not been modified. A failure notification has been sent to WINMAN operator.

### Primary
WINMAN operator

### Secondary Actors
IP EMS and WDM EMS

### Trigger
WINMAN receives a request from the operator through the GUI interface to modify the network configuration.

### DESCRIPTION

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The user navigates to the required site on the Topology map displays.</td>
</tr>
</tbody>
</table>
| 2 | The user selects the network element that he wants to update or delete  
   Or  
   The user selects the place where he wants to add the network element |
| 3 | For link configurations the user selects the link on the map for updating it or deleting it  
   Or  
   The users draws the line that connects A node to Z node in order to create a new link |
| 4 | The WINMAN operator requests the network inventory to be modified. The request shall include:  
   Type of transaction: (add, delete, modify)  
   Identification of a network element (not needed for additions).  
   New element parameters (not needed for delete).  
   Scheduling parameters (e.g. activation time)  
   Operation mode (automatic, semiautomatic or manual) |
| 5 | WINMAN validates the correctness of the parameters included in the request. WINMAN validates the request against system policies. |
| 6 | WINMAN calculates the network changes needed to meet the inventory modifications and the effect that these changes can have over running and scheduled services. This could be included in a report. |
| 7 | WINMAN sends commands to the IP EMS or WDM EMS through the southbound interface to carry out the appropriate actions on the network. This is done by triggering the Atomic Level UCs that correspond to the desired change. |
| 8 | WINMAN checks that the network changes have been carried out properly, processing the messages received from the IP and WDM EMS’s through the corresponding Atomic Level UCs. |
| 9 | Configure Network sends a notification to the WINMAN operator, or the superordinate UC, indicating that the inventory has been modified as requested. |
| 10 | The network map is updated with all the changes and displays the new network view. |

### EXTENSIONS

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>The Map can not be displayed, an error is displayed to the user.</td>
</tr>
<tr>
<td>5a</td>
<td>If the validation fails a failure condition is reached. The use case ends.</td>
</tr>
<tr>
<td>6a</td>
<td>If the new configuration affects active services, the system tries to allocate other resources for these services. If this attempt fails, the use-case ends with a failure condition.</td>
</tr>
</tbody>
</table>
6b If a service interruption is needed in order to implement the changes, WINMAN will send a message.

8a If the actions to be carried out on the network fail, a failure condition is reached. The use case ends.

### RELATED INFORMATION

**Priority:** High

**Channels to actors**
- GUI towards the WO
- Southbound interfaces to EMS’s

**OPEN ISSUES**
- How to reassign resources to affected services.
- How to calculate how the change effects over the established services and define the policies to apply.

**Release**
- R0

...any other management information...

**Superordinates**
- Register EMS
- Unregister EMS
- Notify Network Change
- Revert Network
- Recover Network.

**Subordinates**
- Add IP node
- Delete IP node
- Add WDM node
- Delete WDM node
- Add WDM link
- Delete WDM link
- Modify WDM link
- Add EMS
- Delete EMS

**Requirements Traceability**
- WIN_COM_002
- WIN_COM_007
- WIN_CM_109
- WIN_CM_110
- WIN_CM_120
- WIN_CM_122
- WIN_CM_127
- WIN_CM_129
- WIN_CM_202
- WIN_CM_203
- WIN_CM_204
- WIN_FM_013

### B.5.5. Add IP node

<table>
<thead>
<tr>
<th><strong>USE CASE # IA1</strong></th>
<th>Add IP node</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Goal in Context</strong></td>
<td>This use case is used by the Configure Network use case when new IP node-s are going to be added to the WINMAN system</td>
</tr>
<tr>
<td><strong>Level</strong></td>
<td>Atomic</td>
</tr>
<tr>
<td><strong>Preconditions</strong></td>
<td>The new IP nodes are physically connected to the network, configured, and the responsible IP EMS provides a supported, from the WINMAN system, interface.</td>
</tr>
<tr>
<td><strong>Success End Condition</strong></td>
<td>Specific IP nodes have been added to the WINMAN system</td>
</tr>
<tr>
<td><strong>Failed End</strong></td>
<td>Unable to add the specified IP nodes</td>
</tr>
</tbody>
</table>
Condition | Primary Actors | Secondary Actors | IP EMS
---|---|---|
Trigger | Request by the WINMAN Operator, or underlying network topology changes |

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The Configure Network use case requests the addition of specific (IP address) nodes to the WINMAN system.</td>
</tr>
<tr>
<td>2</td>
<td>The WINMAN system contacts the appropriate IP EMSs (distinguished from the IP address of the new nodes) in order to collect the needed information.</td>
</tr>
<tr>
<td>3</td>
<td>The IP EMSs response with their IP nodes information.</td>
</tr>
<tr>
<td>4</td>
<td>The WINMAN system creates a new IP node record to the IP NMS DB for each requested IP node that was successfully contacted.</td>
</tr>
<tr>
<td>5</td>
<td>The WINMAN system notifies the Configure Network use case about the addition of the new IP nodes.</td>
</tr>
<tr>
<td>6</td>
<td>The WINMAN system notifies the Configure Network about the addition of the new IP nodes.</td>
</tr>
</tbody>
</table>

**EXTENSIONS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3a</td>
<td>There is no such IP node. Use case ends.</td>
</tr>
<tr>
<td>5a</td>
<td>The WINMAN system fails to create a new IP nodes records.</td>
</tr>
<tr>
<td>5b</td>
<td>The IP node(s) already exists in the DB. If some change is detected the databases are updated.</td>
</tr>
</tbody>
</table>

**SUB-VARIATIONS**

**RELATED INFORMATION**

<table>
<thead>
<tr>
<th>Channels to actors</th>
<th>Southbound interfaces to EMSs</th>
</tr>
</thead>
</table>

**OPEN ISSUES**

<table>
<thead>
<tr>
<th>Release</th>
<th>R0</th>
</tr>
</thead>
</table>

**Superordinates**

<table>
<thead>
<tr>
<th>Configure Network</th>
</tr>
</thead>
</table>

**Subordinates**

| WIN_CM_002, WIN_CM_131, WIN_CM_203 |

**B.5.6. Delete IP node**

**USE CASE # IA2**

<table>
<thead>
<tr>
<th>Goal in Context</th>
<th>Delete IP node</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Level</th>
<th>Atomic</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Preconditions</th>
<th>The IP nodes are previously added to the WINMAN system (thus stored in the DB)</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Success End Condition</th>
<th>IP nodes have been deleted</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Failed End Condition</th>
<th>Unable to delete the specified IP nodes</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Primary Actors</th>
<th>IP EMS</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Secondary Actors</th>
<th>IP EMS</th>
</tr>
</thead>
</table>
### B.5.7. Add WDM node

**USE CASE # IA3**  
Add WDM node

**Goal in Context**  
The creation of a WDM node by Configure Network use case.

**Level**  
Atomic level.

**Preconditions**  
Communication towards the WDM EMS has been established.  
The IP address of the WDM node is known to WINMAN.

**Success End Condition**  
A WDM node has been added.

**Failed End Condition**  
Failed to Add a WDM node.

**Primary Actors**  
WO

**Secondary Actors**  
WDM EMS

**Trigger**  
Request by the WINMAN operator or underlying topology has changed

<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The Configure Network use case creates an instance of the WDM node.</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>The WINMAN system establishes a connection to the WDM node via</td>
<td></td>
</tr>
<tr>
<td>Step</td>
<td>Branching Action</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>------------------</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>The Configure Network uploads the configuration information of the WDM node.</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>The configuration information is provided to the WINMAN system, updating the NMS database.</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>When the instance is created and configured according to the WDM node, the Configure Network is notified about the success.</td>
<td></td>
</tr>
</tbody>
</table>

**EXTENSIONS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>Cannot create an instance of the WDM node; an instance already exists. Delete existing WDM node instance and start again.</td>
</tr>
<tr>
<td>2a</td>
<td>A connection between the WINMAN system and the WDM node cannot be established. Use case ends.</td>
</tr>
<tr>
<td>3a</td>
<td>WDM node configuration cannot be uploaded. Use case ends.</td>
</tr>
<tr>
<td>4a</td>
<td>The database cannot be updated with configuration information of WDM node. Use case ends.</td>
</tr>
<tr>
<td>5a</td>
<td>The WDM instance cannot be created. Use case ends.</td>
</tr>
</tbody>
</table>

**RELATED INFORMATION**

- **Priority:** High
- **Channels to actors:** Southbound interfaces to actors.

**OPEN ISSUES**

- Release: R0

**Superordinates:** Configure Network

**Subordinates:**

- WIN_COM_001
- WIN_CM_105
- WIN_CM_107
- WIN_CM_111
- WIN_CM_127
- WIN_CM_129

**B.5.8. Delete WDM node**

**USE CASE # IA4**

- **Goal in Context:** The deletion of a WDM node by the Configure Network use case.
- **Level:** Atomic level.

**Preconditions:**

- A WDM node instance exists.

**Success End Condition:**

- A WDM node has been deleted

**Failed End Condition:**

- Unable to delete the WDM node

**Primary Actors:**

- WO

**Secondary Actors:**

- WDM EMS

**Trigger:**

- Request by the WINMAN operator

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Find the WDM node instance.</td>
</tr>
<tr>
<td>2</td>
<td>Check if links have been established from this WDM node instance.</td>
</tr>
<tr>
<td>3</td>
<td>If WDM links exist delete them.</td>
</tr>
</tbody>
</table>
### Delete the WDM node instance from the database.

When the instance is deleted, the Configure Network is notified about the success.

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>No WDM node instance was found in database. Use case ends.</td>
</tr>
<tr>
<td>3a</td>
<td>WDM links cannot be deleted. Use case ends.</td>
</tr>
</tbody>
</table>

#### SUB-VARIATIONS

**Branching Action**

### RELATED INFORMATION

**Priority:** High

**Channels to actors:** Southbound interfaces to actors.

**OPEN ISSUES**

Release R0

...any other management information...

**Superordinates** Configure Network

**Subordinates**

**Requirement traceability** WIN_COM_001, WIN_CM_105, WIN_CM_107, WIN_CM_111, WIN_CM_127, WIN_CM_129

---

## B.5.9. Add WDM link

**USE CASE # IA5** Add WDM link

**Goal in Context** The creation of a WDM link by the Configure Network use case.

**Level** Atomic level.

**Preconditions**

- WDM nodes have been already created by WINMAN System.
- Physical connection between WDM nodes has been established.

**Success End Condition** A WDM link has been created.

**Failed End Condition** Failed to create a WDM link.

**Primary Actors** WO

**Secondary Actors** WDM EMS

**Trigger** Request by the WINMAN operator

### DESCRIPTION

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The WINMAN system creates an instance of the WDM link.</td>
</tr>
<tr>
<td>2</td>
<td>The system checks the available connection between the two nodes.</td>
</tr>
<tr>
<td>3</td>
<td>The Configuration Network use case requests the link establishment for a certain lambda, between the two nodes.</td>
</tr>
<tr>
<td>4</td>
<td>The system updates the WINMAN System database.</td>
</tr>
<tr>
<td>5</td>
<td>When the instance is created and configured according to the WDM link, the Configure Network is notified about the success.</td>
</tr>
</tbody>
</table>

### EXTENSIONS

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>Cannot create an instance of the WDM link. Use case ends.</td>
</tr>
<tr>
<td>2a</td>
<td>No available connection exists. Use case ends.</td>
</tr>
<tr>
<td>3a</td>
<td>Failure of the link establishment. Use case ends.</td>
</tr>
<tr>
<td>4a</td>
<td>The database cannot be updated with configuration information of</td>
</tr>
</tbody>
</table>
B.5.10. Delete WDM link

USE CASE # IA6  | Delete a WDM link
---|---
Goal in Context | The deletion of a WDM link by the Configure Network use case.
Level | Atomic level.
Preconditions | A WDM link instance exists.
Success End Condition | A WDM link has been deleted.
Failed End Condition | Unable to delete the WDM link.
Primary Actors | WO
Secondary Actors | WDM EMS
Trigger | Request by the WINMAN operator

DESCRIPTION

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Find the WDM link instance.</td>
</tr>
<tr>
<td>2</td>
<td>Check if links have traffic on them.</td>
</tr>
<tr>
<td>3</td>
<td>If WDM links exist delete them.</td>
</tr>
<tr>
<td>4</td>
<td>Delete the WDM link instance from the database. When the instance is deleted, the Configure Network is notified about the success.</td>
</tr>
</tbody>
</table>

Step | Branching Action |
---|---|
| 1a | No WDM link instance was found in database. Use case ends. |
| 3a | WDM links cannot be deleted. Use case ends. |
| 4a | The system fails to delete the WDM instance. Use case ends. |

SUB-VARIATIONS | Branching Action

RELATED INFORMATION

Priority: High
Channels to actors: Southbound interfaces to actors.

OPEN ISSUES
B.5.11. Modify WDM link

**USE CASE # IA7**  
Modify a WDM link

**Goal in Context**  
The modification of a WDM link by Configure Network use case.

**Level**  
Atomic level.

**Preconditions**  
A WDM link instance exists. The link characteristics can be modified.

**Success End Condition**  
A WDM link has been modified.

**Failed End Condition**  
Unable to modify the WDM link.

**Primary Actors**  
WO

**Secondary Actors**  
WDM EMS

**Trigger**  
Request by the WINMAN operator

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Find the WDM link instance.</td>
</tr>
<tr>
<td>2</td>
<td>Check if link has traffic on it.</td>
</tr>
<tr>
<td>3</td>
<td>If WDM link exists modify it.</td>
</tr>
</tbody>
</table>
| 4    | Modify the WDM link instance in the database.  
When the instance is modified, the Configure Network is notified about the success. |

**Step**  
**Branching Action**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
</table>
| 1a   | No WDM link instance was found in database.  
Use case ends. |
| 3a   | WDM link cannot be modified.  
Use case ends. |
| 4a   | The system fails to modify the WDM instance.  
Use case ends. |

**SUB-VARIATIONS**  
**Branching Action**

**RELATED INFORMATION**

| Priority: | High |
| Channels to actors | Southbound interfaces to actors. |

**OPEN ISSUES**

**Release**  
R0

...any other management information...

**Superordinates**  
Configure Network

**Subordinates**

**Requirement traceability**

WIN_COM_001  
WIN_CM_105  
WIN_CM_107  
WIN_CM_111  
WIN_CM_127  
WIN_CM_129

B.5.12. Add EMS
USE CASE # IA8  Add EMS

Goal in Context  Modify the Network configuration (update inventory), by adding new EMS information.

Scope & Level  Atomic Level

Preconditions  An option to add a new EMS info in the network inventory is available for WINMAN operator in the GUI.
The WINMAN system is ready for inventory modifications and holds all necessary data about the configured elements and network policies rules.
The new EMS has authenticated WINMAN and a communication link has been established between EMS and WINMAN at the S/B i/f.

Success End Condition  The network inventory parameters are changed and the new EMS info is available to WINMAN.
A notification has been sent to WINMAN operator.

Failed End Condition  The inventory has not been modified.
A failure notification has been sent to WINMAN operator.

Primary  WINMAN operator, EMS

Secondary Actors  IP EMS, WDM EMS

Trigger  WINMAN receives a request from the WO through the GUI interface, or from the EMS through the Register EMS procedure, to modify the network configuration by adding a new EMS information in the network inventory.

DESCRIPTION  Step  Action

1  WINMAN requests the network inventory to be modified via the Configure Network UC, to incorporate new EMS inventory data.
The request may include:
Type of transaction: "Add EMS inventory"
New EMS parameters (provided by Register EMS, or known by the WO)
Scheduling parameters (e.g. activation time)
Operation mode (EMS triggered: automatic / semiautomatic, WO triggered: manual)
Inventory Update policy

2  WINMAN creates a root entity in the Network Inventory representing the new EMS and adds new EMS data. This can be done in a number of ways:
WINMAN sends a series of queries to the new EMS, requesting retrieval of specific information
WINMAN sends one request for information retrieval and the EMS responds by producing a chain of network change notifications.

3  WINMAN checks that the network changes have been carried out properly, processing the messages received from the new EMS, regarding the automatic / manual addition via Atomic Level UCs of the entities managed by the EMS. A synchronisation of notifications (effectively alarms) should be effected at this stage (done by the WO?).

4  A message is sent to the Configure Network UC, indicating that the inventory has been modified as requested.

EXTENSIONS  Step  Branching Action

3a  If the actions to be carried out on the network fail, a failure condition is reached. The use case ends.

RELATED INFORMATION  TMF doc 509: NML-EML I/F Business Agreement:
Discovery & Inventory UCs

Priority:  High
Channels to actors | Southbound interfaces to EMS’s
--- | ---
OPEN ISSUES | EMS parameters
Define the updating policies to apply.
Does adding an EMS imply automatic addition of the structure that underlies (i.e. all managed elements), and how is this reported to the Inventory?
Release | R0
...any other management information...
Superordinates | Configure Network
Subordinates | WIN_COM_002
WIN_COM_007
WIN_CM_109
WIN_CM_110
WIN_CM_111
WIN_CM_120
WIN_CM_122
WIN_CM_127
WIN_CM_129
WIN_CM_202
WIN_CM_203
WIN_CM_204
WIN_FM_013

B.5.13. Delete EMS

<table>
<thead>
<tr>
<th>USE CASE # IA9</th>
<th>Delete EMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal in Context</td>
<td>Modify the Network configuration (update inventory), by deleting an EMS.</td>
</tr>
<tr>
<td>Scope &amp; Level</td>
<td>Atomic Level</td>
</tr>
<tr>
<td>Preconditions</td>
<td>An option to modify the network inventory is available for WINMAN operator in the GUI. The WINMAN system is ready for inventory modifications and holds all necessary data about the configured elements and network policies rules.</td>
</tr>
<tr>
<td>Success End Condition</td>
<td>The network inventory parameters are changed and the new configuration of EMSs (reduced by one) is operational. A notification has been sent to WINMAN operator.</td>
</tr>
<tr>
<td>Failed End Condition</td>
<td>The inventory has not been modified. A failure notification has been sent to WINMAN operator.</td>
</tr>
<tr>
<td>Primary</td>
<td>WINMAN operator, EMS</td>
</tr>
<tr>
<td>Secondary Actors</td>
<td>IP EMS and WDM EMS</td>
</tr>
<tr>
<td>Trigger</td>
<td>WINMAN receives a request from the WO through the GUI interface, or from the EMS through the UnRegister EMS procedure, to modify the network configuration by deleting an existing EMS.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>WINMAN requests the network inventory to be modified via the Configure Network UC. The request may include: Type of transaction: &quot;Delete EMS&quot; EMS parameters Scheduling parameters (e.g. de-activation time) Operation mode (automatic, semiautomatic or manual) Inventory update mechanism</td>
</tr>
</tbody>
</table>
|             | 2    | WINMAN deletes the root entity in the Network Inventory representing the EMS under deletion and deletes all relating...
EMS data. The data structure under the root entry for an EMS may be deleted in one step, or WINMAN sends a request to the EMS and EMS issues a series of network change notifications.

3 WINMAN checks that the network changes have been carried out properly, processing the messages received from the S/B i/f. A synchronisation of alarms could be effected at this stage.

4 A message is sent to the Configure Network UC, indicating that the inventory has been modified as requested.

EXTENSIONS

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>3a</td>
<td>If the actions to be carried out on the network fail, a failure condition is reached. The use case ends.</td>
</tr>
</tbody>
</table>

RELATED INFORMATION

| Priority: | High |
| Channels to actors: | Southbound interfaces to EMS’s |

OPEN ISSUES

How to reassign resources to affected services.
How to calculate how the change effects over the established services and define the policies to apply.

Release

R0

...any other management information...

Superordinates

Configure Network

Subordinates

Requirements

WIN_COM_002
WIN_COM_007
WIN_CM_109
WIN_CM_110
WIN_CM_111
WIN_CM_120
WIN_CM_122
WIN_CM_127
WIN_CM_129
WIN_CM_202
WIN_CM_203
WIN_CM_204
WIN_FM_013

B-6. Network Maintenance & Restoration UCs

B.6.1. Recover Network

USE CASE # MS1

Recover network.

Goal in Context

Reroute all connections by-passing faulty parts of the network

Scope & Level

Secondary Task for WINMAN. Summary level.

Preconditions

The alarm is active and there are other nodes that can support the relevant connections.

Success End Condition

All affected connections are successfully recovered using alternative network resources (nodes, links, etc).

Failed End Condition

Some or all of the connections cannot be recovered and are still faulty.

Primary Actors

WINMAN operator, EMS

Secondary Actors

Trigger

The operator asks to recover the network. It can also be automatically triggered when the NLA associated with the connections are not
fulfilled due to faults in the network

<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>Check network recovery policies</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Try to assign the affected CS to alternative network resources (nodes, links, etc) by-passing the faulty parts according to the restoration policies.</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Store any change in the original route of every affected CS in the DB in order to be able to revert the network to its previous state (revert network use case). The term original route refers to the route that was used before the failure occurrence</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Notify the Winman Operator about the result of the recovery actions</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>EXTENSIONS</th>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1a</td>
<td>Network restoration policies are not present or cannot be checked. No restoration is performed. Use case ends.</td>
</tr>
<tr>
<td></td>
<td>2a</td>
<td>Some or all of the CS cannot be recovered. Issue an alarm report and change the status of these CS as faulty. Add the problem to the alarm report. Continue with the use case.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RELATED INFORMATION</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>High</td>
</tr>
<tr>
<td>Channels to actors</td>
<td>GUI console, southbound interface to EMS, internal BUS.</td>
</tr>
</tbody>
</table>

| OPEN ISSUES | The following open issues could be addressed using policies about network recovery: |
|-------------| Do we want to re-route also CSs that are not in use when the problem occurs? |
|             | What about non-active CSs that become active while the node is down? |
|             | Is this another trigger? |
|             | If there is more than one option to handle rerouting, how do we decide which one to use? |

| Release | R1 |

| ...any other management information... | |

<table>
<thead>
<tr>
<th>Superordinates</th>
<th>Identify affected connections</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subordinates</td>
<td>Configure network Execute Policy Modify ICS</td>
</tr>
</tbody>
</table>

| Requirements Traceability | WIN_CM_002 WIN_CM_129, WIN_CM_132, WIN_FM_001, WIN_FM_002, WIN_FM_012, WIN_FM_014 |

| B.6.2. Revert Network | |

<table>
<thead>
<tr>
<th>USE CASE # MS2</th>
<th>Revert network.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal in Context</td>
<td>Return the original routing – as was before the recover network use case invocation.</td>
</tr>
</tbody>
</table>
### Scope & Level
Secondary Task for WINMAN, Summary Level.

### Preconditions
The recover network operation stored recovery records in the DB.

### Success End Condition
The network is reverted to the state before the fault, thus all the CS use the original routes and are in operational status.

### Failed End Condition
The revert network process cannot be carried out.

### Primary Actors
WINMAN operator

### Secondary Actors

### Trigger
The operator asks to revert the network to the state it was before the recover actions carried out

### DESCRIPTION

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Retrieve the recovery records that the recover network use case stored, and verify its validity.</td>
</tr>
<tr>
<td>2</td>
<td>Reassign the original routes.</td>
</tr>
<tr>
<td>3</td>
<td>Notify the Winman Operator about the result of the revert network actions</td>
</tr>
</tbody>
</table>

### EXTENSIONS

<table>
<thead>
<tr>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
</tr>
<tr>
<td>2a</td>
</tr>
</tbody>
</table>

#### Branching Action 1a
The recovery records are missing or invalid. Cancel the revert network actions and issue an internal alarm.

#### Branching Action 2a
The original routing schema is no longer feasible due to changes in the network topology or resources allocation. Add the problems to the report and leave the routing as is. Continue with the use case.

### SUB-VARIATIONS

### RELATED INFORMATION
- **Priority:** High
- **Channels to actors:** GUI console, southbound interface to EMS, internal BUS.

### OPEN ISSUES
What about non-active CSs that become active while the node is down? Is this another trigger? Is this part of the policies?

### Release
R1

### Superordinates
Clear Alarm (which was triggered by the WINMAN operator)

### Subordinates
Configure network, Execute Policy

### Requirements Traceability
WIN_CM_002

## B.6.3. Open Alarm

### USE CASE # MS3
Open alarm.

### Goal in Context
Handle alarm issued by an EMS or by an internal WINMAN application like the PM or the topology manager when the NE status is changed.

### Scope & Level
Primary Task for WINMAN, Summary level.

### Preconditions
The alarms mechanism is active and not in a faulty state.
The system is connected to the EMS

### Success End Condition
The new alarm is added to the list of active alarms.

### Failed End Condition
WINMAN can’t add this alarm to the list.

### Primary Actors
EMS, Internal
<table>
<thead>
<tr>
<th>Secondary Actors</th>
<th>WINMAN operator, SMS.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Trigger</strong></td>
<td>An EMS sends an alarm-notification message via the southbound interface. Some internal WINMAN server decides to issue an alarm in order to notify the operator of some erroneous situation.</td>
</tr>
<tr>
<td><strong>DESCRIPTION</strong></td>
<td><strong>Step</strong></td>
</tr>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>11</td>
</tr>
<tr>
<td><strong>EXTENSIONS</strong></td>
<td><strong>Step</strong></td>
</tr>
<tr>
<td></td>
<td>1a</td>
</tr>
<tr>
<td></td>
<td>2a</td>
</tr>
<tr>
<td></td>
<td>3a</td>
</tr>
<tr>
<td><strong>SUB-VARIATIONS</strong></td>
<td><strong>Branching Action</strong></td>
</tr>
<tr>
<td><strong>RELATED INFORMATION</strong></td>
<td><strong>Priority:</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Channels to actors</strong></td>
</tr>
</tbody>
</table>
|                     | **OPEN ISSUES** | Do we update status information by alarms or only by designated
messages? Do we have to check alarm notifications against policies? Is the GUI pulling the massage of the new alarm or are we pushing it to the GUI?

<table>
<thead>
<tr>
<th>Release</th>
<th>R1</th>
</tr>
</thead>
<tbody>
<tr>
<td>...any other management information...</td>
<td>Identify root cause Process performance measurements Calculate aggregations and statistics Almost every use case needs to issue an alarm on erroneous situations.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Subordinates</th>
<th>Identify affected connections Identify root cause Execute Policy</th>
</tr>
</thead>
</table>

|---------------------------|------------|------------|------------|------------|------------|------------|------------|------------|

| B.6.4. Clear Alarm |

**USE CASE # MS4** Clear alarm.

**Goal in Context** Clear an alarm and remove it from the active alarm list.

**Scope & Level** Primary Task for WINMAN. Summary Level.

**Preconditions**
The alarms mechanism is active and not in a faulty state.
The system is connected to the EMS

**Success End Condition**
The alarm is removed from the list of active alarms and moved to a list of historic alarms.

**Failed End Condition**
The alarm is not found, or the system can’t remove it from the list of active alarms.

**Primary Actors** WINMAN operator, EMS, Internal

**Secondary Actors** WINMAN operator.

**Trigger**
An EMS sends an alarm-cleared notification message via the southbound interface.
Some internal WINMAN server such as the “Execute policy” UC decides to remove an alarm in order to notify the operator that the erroneous situation is over. In this case an "alarm cleared" notification is triggered. An authorized operator decides that the alarm is no longer relevant and removes it manually.

**DESCRIPTION Step Action**

1. Validate that the request is valid.
2. Mach the event of alarm clear to the open alarm using alarm ID if exists or other convention that can be in the policies.
3. Remove the alarm from the list of active alarms and change the alarm status to 'clear'. Log the time the alarm was cleared and who cleared it.
4. Put the alarm in the list of cleared alarms.
5. Update all relevant views on the GUI console.
6. If needed – update the status of the related network elements.
7. Update the alarm count on all the related Network elements from the port and card to the entire NE.
8. Send notification to the operator that the alarm was cleared.
### EXTENSIONS

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>The request is not valid. Issue an alarm to indicate that the EMS is sending bogus messages. If the message is internal, issue an internal fault alarm.</td>
</tr>
<tr>
<td>2a</td>
<td>The Open alarm that the closing event is related to was not found – according to the policies either create a new alarm with the status 'clear' or activate the alarm synchronization mechanism or notify the users.</td>
</tr>
</tbody>
</table>

### RELATED INFORMATION

**Priority:** High

**Channels to actors:** GUI, southbound interface to EMS’s and internal bus.

**OPEN ISSUES**
- Do we update status information by alarms or only by designated messages?
- Can an operator clear an alarm in a NE from GUI although the NE keeps alarmed (the fault condition is present yet)?
- Is the GUI pulling the message of the new alarm or are we pushing it to the GUI?

**Release**
- R1

**...any other management information...**

**Superordinates**
- Identify root cause
- Process performance measurements
- Calculate aggregations and statistics
- Almost every use case needs to remove alarms when erroneous situations end.

**Subordinates**
- Revert network.

**Requirement traceability**
- WIN-FM-001
- WIN-FM-002
- WIN-FM-003
- WIN-FM-005
- WIN-FM-018

### B.6.5. Provide Alarm Reports

**USE CASE # MS5**
- Provide alarm reports

**Goal in Context**
- Create reports on the different attributes of the alarm

**Scope & Level**
- Secondary Task for WINMAN. Summary Level.

**Preconditions**
- The alarms are in the active alarms list or in the history log

**Success End Condition**
- The required report is presented

**Failed End Condition**
- Such a report cannot be generated – the operator is informed.

**Primary Actors**
- WINMAN operator, SMS

**Secondary Actors**

**Trigger**
- An operator uses the GUI console to request a report. SMS asks WINMAN information about alarm status.

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The user uses a predefined report (like all the affected entities report) or creates a new report (like all the alarms that are opened for more then a required period of time). SMS sends a request to WINMAN asking about alarm status information.</td>
</tr>
</tbody>
</table>
2 Create the query according to the way the data is stored

3 Query the active list for the relevant alarms or query the history log if needed.
Query the topology system if the data asked is regarding the affected ICSs or NE.

4 Create the required report and send it to the user, the GUI or the SMS

EXTENSIONS

Step Branching Action

3a Alarms related to the reports query are not found.
Sites, connectivities related to the reports query are not found.
Issue an error message and end use case.

SUB-VARIATIONS Branching Action

RELATED INFORMATION

Priority: Low
Channels to actors GUI console.

OPEN ISSUES

What is the algorithm creating the report?
Are we going to use an external report generator? Or a tool for all Winman reports?
What information do we need to include in this report?
What is the format needed for this report?
Can the user choose among several formats?
Do we need to offer report generator to design this report?
Is there a need to export this data to external tools?

Release R1

...any other management information...

Superordinates
Subordinates

Requirement traceability WIN-FM-015
WIN-FM-016

B.6.6. Identify Affected Connections

USE CASE # MU1 Identify affected connections.

Goal in Context Find all the active connections influenced by a given alarm.

Scope & Level Primary Task for WINMAN. User Level.

Preconditions The alarms mechanism is active and not in a faulty state. The alarm is in the active alarms list.

Success End Condition A list of disturbed connections is returned.

Failed End Condition Such a report cannot be generated – the operator is informed.

Primary Actors WINMAN operator, Internal

Secondary Actors SMS

Trigger An operator uses the GUI console to request a report of affected connections. An internal WINMAN application needs to have a list of affected connections. The "Open Alarm" UC is activating this UC when a new alarm is received.

DESCRIPTION Step Action

1 Find the nodes affected by this alarm

2 Find all ICSs configured to use these nodes.

3 Among these services, find which are active and working properly.

4 Send a report to the operator.
Generating alarms for each affected connection. These alarms could be associated to the root and the operator could choose to see them or not. These alarms (or information on working time) can be used for calculating the circuit availability.

5 If possible, try to recover the network. This is done by the "Recover Network" use case.

**EXTENSIONS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>Alarm is not found. Issue an error message and end use case.</td>
</tr>
<tr>
<td>2a</td>
<td>No ICSs are configured to this node. Returns an empty list.</td>
</tr>
</tbody>
</table>

**SUB-VARIATIONS**

| Branching Action |

---

**RELATED INFORMATION**

| Priority: | High |
| Channels to actors: | GUI console. |

**OPEN ISSUES**

What is the algorithm to conclude that a node is affected by an alarm? Is this UC finding the NE by itself or should he use the “Configure Network” UC?

**Release**

...any other management information...

**Superordinates**

Open alarm

**Subordinates**

Recover network

**Requirement traceability**

WIN-FM-012

**B.6.7. Identify Root Cause**

**USE CASE # MU2**

Identify root cause.

**Goal in Context**

Identify the root causes in the alarm list.

**Scope & Level**

Secondary Task for WINMAN. User Level.

**Preconditions**

There are active alarms in the list.

**Success End Condition**

WINMAN produces a list of root causes.

**Failed End Condition**

Not root cause can be identified.

**Primary Actors**

WINMAN operator, Internal

**Secondary Actors**

This use case can be automatically triggered upon alarm reception or an operator can request to identify root cause.

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Check if the alarms have common logic according to topology rules, incidental cases or any other customer logic that is described in the “Execute policy” UC.</td>
</tr>
<tr>
<td>2</td>
<td>Use the rules to build a list of root causes. Identify the root cause as a “Parent alarm” and the related alarm as “Child alarms”</td>
</tr>
<tr>
<td>3</td>
<td>Send a report describing identified root causes.</td>
</tr>
<tr>
<td>4</td>
<td>Update all relevant views on the GUI console.</td>
</tr>
<tr>
<td>5</td>
<td>Generate new alarms according to the root cause.</td>
</tr>
</tbody>
</table>

**EXTENSIONS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>No rule can be applied. End the use case.</td>
</tr>
</tbody>
</table>

**SUB-VARIATIONS**

| Branching Action |
### RELATED INFORMATION

<table>
<thead>
<tr>
<th>Priority:</th>
<th>Low</th>
</tr>
</thead>
<tbody>
<tr>
<td>Channels to actors:</td>
<td>GUI console.</td>
</tr>
</tbody>
</table>

### OPEN ISSUES

- **Do we clear related alarms when a root cause alarm is triggered?** It can be filtered by the GUI using the Parent / Child attribute.
- **Do we designate a separated feature for root cause identification or is this feature covered via the standard alarms mechanism?** We can use the Parent / Child attribute in order to notify and mark the alarms as root cause.
- **The Policy management tool and the rule base tool need to be defined.**

### Release

<table>
<thead>
<tr>
<th>Release:</th>
<th>R1</th>
</tr>
</thead>
</table>

### ...any other management information...

### Superordinates

**Open alarm**

### Subordinates

- **Execute policy**
- **Open alarm**

### Requirement traceability

**WIN-FM-011**

### B.6.7.1 Synchronise Alarms

**USE CASE # MU3**

<table>
<thead>
<tr>
<th><strong>Goal in Context</strong></th>
<th>Query the EMS for their open alarms for re-synchronisation of the active alarms in the WINMAN system.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><strong>Scope &amp; Level</strong></th>
<th>Primary Task for WINMAN. User level.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><strong>Preconditions</strong></th>
<th>The alarms mechanism is active and not in a faulty state. There is connection to the EMS.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><strong>Success End Condition</strong></th>
<th>All the open alarms that are in the EMS are displayed and no cleared alarm is displayed</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><strong>Failed End Condition</strong></th>
<th>The active alarm list was not updated</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><strong>Primary Actors</strong></th>
<th>WINMAN operator</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><strong>Secondary Actors</strong></th>
<th>EMS, automatic after a system crash</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><strong>Trigger</strong></th>
<th>The operator requests to synchronise the alarms. The system asks to be synchronised after a crash, many (defined by the customer) faulty events or a connection fault recovery.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th><strong>DESCRIPTION</strong></th>
<th><strong>Step</strong></th>
<th><strong>Action</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1</strong></td>
<td>Flag all the active alarms</td>
<td></td>
</tr>
<tr>
<td><strong>2</strong></td>
<td>Query the EMS about its open alarms</td>
<td></td>
</tr>
<tr>
<td><strong>3</strong></td>
<td>Get all the events of the open alarms from the EMS, and create all the open alarms from the events using use case of Open an alarm</td>
<td></td>
</tr>
<tr>
<td><strong>4</strong></td>
<td>Check for duplicate alarm according to the site, alarm cause and time. Delete all the duplicate alarms with the flag.</td>
<td></td>
</tr>
<tr>
<td><strong>5</strong></td>
<td>All the alarms with the flag that were not deleted are cleared alarms, change their status to clear and remove the flag. Use Clear alarm use case.</td>
<td></td>
</tr>
<tr>
<td><strong>6</strong></td>
<td>Notify the users that the action is finished.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>EXTENSIONS</strong></th>
<th><strong>Step</strong></th>
<th><strong>Branching Action</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>2a</strong></td>
<td>The EMS can not be queried</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Notify the user that the alarms can not be synchronised.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>SUB-VARIATIONS</strong></th>
<th><strong>Branching Action</strong></th>
</tr>
</thead>
</table>

| **RELATED** |---|
B.6.8. Acknowledge Alarm

USE CASE # MU4 Acknowledge alarm.
Goal in Context Move an alarm to an 'acknowledged' status.
Scope & Level Secondary Task for WINMAN. User level.
Preconditions The alarms mechanism is active and not in a faulty state. The alarm is in the active alarms list.
Success End Condition The alarm status becomes acknowledged.
Failed End Condition The alarm is not found or the system can’t change its status.
Primary Actors WINMAN operator
Secondary Actors
Trigger An operator uses the GUI console to acknowledge an alarm.

DESCRIPTION

Step | Action
---|---
1 | Validate that the request is valid.
2 | Check that the alarm is not already acknowledged.
3 | Set the internal status of that alarm to 'acknowledged'. Record the user that acknowledged it and when he did that.
4 | Update all relevant views on the GUI console.

EXTENSIONS

Step | Branching Action
---|---
1a | The request is not valid. Issue an error message and end use case.
2a | The alarm is already acknowledged. Issue an error message and end use case.

SUB-VARIATIONS

Branching Action

B-7. Network Data Management UCs

B.7.1. Process Performance Measurements
**USE CASE # DS1**

**Goal in Context**
Collect the counters according to the defined policy. If the thresholds are crossed an alarm is triggered.

**Scope & Level**
Primary Task for WINMAN. Summary level.

**Preconditions**
The performance log mechanism is active and not in a faulty state.

**Success End Condition**
The counters are collected, stored and presented.

**Failed End Condition**
No counters are collected and displayed

**Primary Actors**
EMS

**Secondary Actors**
WINMAN operator

**Trigger**
The EMS send events of the measured counters

### DESCRIPTION

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Events of the type of counters are received from the EMS.</td>
</tr>
<tr>
<td>2</td>
<td>The events are checked with the collection policy for their type, priority, thresholds and other attributes.</td>
</tr>
<tr>
<td>3</td>
<td>Data regarding the topology and connectivity trail is translated from the topology module.</td>
</tr>
<tr>
<td>4</td>
<td>The counters that match the policy are collected and stored.</td>
</tr>
<tr>
<td>5</td>
<td>If thresholds are crossed an alarm is created using the 'Open Alarm' use case.</td>
</tr>
<tr>
<td>6</td>
<td>Calculate aggregation of the counters for different periods of time and for ICSs QoS and calculate computed counters and statistics using the 'Calculate aggregations and statistics' use case.</td>
</tr>
<tr>
<td>7</td>
<td>The user and GUI are notified that new counters exists and they are presented on the GUI monitors</td>
</tr>
</tbody>
</table>

### EXTENSIONS

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>No counters were received An alarm that a fault happened is generated.</td>
</tr>
<tr>
<td>2a</td>
<td>The events are not matching the collection policy No counters are collected.</td>
</tr>
<tr>
<td>3a</td>
<td>No topology data exists on the counters connectivity The counters are collected and presented without their relevant topology.</td>
</tr>
</tbody>
</table>

### SUB-VARIATIONS

| Branching Action |

### RELATED INFORMATION

**Priority:** Low

**Channels to actors**
GUI console.

**OPEN ISSUES**
What type of performance monitors are we going to create in the GUI? Which counters are we going to collect? Which threshold mechanism are we going to use?

**Release**
R1

...any other management information...

**Superordinates**
Calculate aggregations and statistics
Open alarm
Clear alarm
Execute policy

**Subordinates**
WIN-PM-001
WIN-PM-006
WIN-PM-010

**Requirement traceability**
### USE CASE # DS2  
**Provide PM report.**

#### Goal in Context
Produce a report with an analysis of quality of ICS.

#### Scope & Level
Secondary Task for WINMAN, Summary level

#### Preconditions
The performance log mechanism is active and not in a faulty state.

#### Success End Condition
A report containing quality of ICS is produced.

#### Failed End Condition
Can’t produce such a report or only partial results are available.

#### Primary Actors
WINMAN operator  
SMS

#### Secondary Actors

#### Trigger
The operator or the SMS requests a report with detailed information about the quality of every connection.

#### DESCRIPTION

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Read all relevant performance counters.</td>
</tr>
<tr>
<td>2</td>
<td>Aggregate the data and interpret counters to form integrated results about every connection, using Use case Calculate aggregation and statistics. Include also deviations from NLA’s.</td>
</tr>
<tr>
<td>3</td>
<td>Format the information to form a report readable by managers and network administrators according to predefined report templates.</td>
</tr>
<tr>
<td>4</td>
<td>Send the report to the GUI console or to any other dispatching address such as printer, fax or mail. If requested send the report to the SMS.</td>
</tr>
</tbody>
</table>

#### EXTENSIONS

<table>
<thead>
<tr>
<th>Branching Action</th>
</tr>
</thead>
</table>
| 1a Some counters are not available.  
Try to complete the report without these counters. Issue an alarm. Continue with the use case. |

#### SUB-VARIATIONS

<table>
<thead>
<tr>
<th>Branching Action</th>
</tr>
</thead>
</table>
| 1a1 Not enough data to compute a report for some connection.  
Just list available counters. Continue with the use case. |

### RELATED INFORMATION

| Priority: | Low |
| Channels to actors | GUI console. |

### OPEN ISSUES

What information do we need to include in this report?  
What is the format needed for this report?  
Can the user choose among several formats?  
Do we need to offer report generator to design this report?  
Is there a need to export this data to external tools?  

### Release
R1

...any other management information...

### Requirement traceability
WIN-PM-007  
WIN-PM-008
B.7.3. Calculate Aggregations & Statistics

<table>
<thead>
<tr>
<th>USE CASE # DU1</th>
<th>Calculate aggregations and statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal in Context</td>
<td>Calculate aggregation of the counters for different periods of time; calculate computed counters and statistics (compare to average and standard deviations) according to the customer’s definitions. Calculate QoS counters for ICSs. If some thresholds were crossed by the calculated values alarms are generated and sent to the FM.</td>
</tr>
<tr>
<td>Scope &amp; Level</td>
<td>Secondary Task for WINMAN. User level.</td>
</tr>
<tr>
<td>Preconditions</td>
<td>The performance log mechanism is active and not in a faulty state.</td>
</tr>
<tr>
<td>Success End Condition</td>
<td>The calculated values are stored and displayed on the monitors and reports.</td>
</tr>
<tr>
<td>Failed End Condition</td>
<td>No calculated values are created</td>
</tr>
<tr>
<td>Primary Actors</td>
<td>WINMAN operator</td>
</tr>
<tr>
<td>Secondary Actors</td>
<td></td>
</tr>
<tr>
<td>Trigger</td>
<td>The Winman operator asks to create some calculation on the counters. New counters were collected and they are sent for aggregation and statistics.</td>
</tr>
</tbody>
</table>

**DESCRIPTION**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Receive all the counters and the counters log for history comparison.</td>
</tr>
<tr>
<td>2</td>
<td>Check the calculation rules using the “Execute Policy” UC.</td>
</tr>
<tr>
<td>3</td>
<td>Get all the topology data in order to calculate the counters for the ICSs.</td>
</tr>
<tr>
<td>4</td>
<td>Check if there are any thresholds that were crossed by the new values using the “Execute Policy” UC. If yes create an alarm using the “Open the alarm” UC.</td>
</tr>
<tr>
<td>5</td>
<td>Save all the new values of calculations.</td>
</tr>
<tr>
<td>6</td>
<td>Notify the user and the GUI of the new calculated values.</td>
</tr>
</tbody>
</table>

**EXTENSIONS**

<table>
<thead>
<tr>
<th>Step</th>
<th>Branching Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>No counters are available or no history log of the counters is available. The calculations can not be done. Create a massage to notify the user.</td>
</tr>
<tr>
<td>2a</td>
<td>No aggregations rule exists. The calculations can not be done. Create a massage to notify the user.</td>
</tr>
</tbody>
</table>

**SUB-VARIATIONS**

<table>
<thead>
<tr>
<th>Branching Action</th>
</tr>
</thead>
</table>

**RELATED INFORMATION**

| Priority: | Low |
| Channels to actors: | GUI console. |

**OPEN ISSUES**

| What tool are we using to define the aggregation and statistics rules? |
| Is this UC receiving all the topology data by himself or is he using the “Configure Network” UC? |

**Release**

| RI |
| 1 |

**any other management information...**

| Open alarm |
| Clear alarm |
| Execute policy |

WINMAN-WP2-PTI-020a-D2-1-b1 © WINMAN Consortium
Appendix C. Component based architectures

C-1. What are component based architectures?

C.1.1. Requirements

C.1.1.1 Connection Technology

To connect the components they must be implemented in the same connection technology. It is obvious that a Corba IDL based component cannot be used with an ftp based application. There are different connection technologies to choose from like: Corba IDL, DCOM, JINI, SQL, FTP, etc. However, some interfaces in WINMAN should be defined in Corba IDL like the northbound and southbound interfaces. Other interfaces however can choose another technology, but if that is the most efficient way is yet to be seen.

C.1.1.2 Need for a co-ordinator

With all those components and interfaces and the ability to support the advanced techniques described in chapter C-3 there is a need for a co-ordinator or Component Manager (a component in itself). This co-ordinator uses for example the Naming Service and Trading services as defined by OMG. It takes care of the following tasks:

- Registration of a component
- Starting up / Shutting down a component
- Report the status of a component (not Registered, Registered, Starting up, Started, Shutting down, Stopped)

The co-ordinator is part of the component framework, which provide the facilities for using components

C.1.2. Component types

C.1.2.1 Interface

This type of component has an interface to the 'outside world' (which is outside the component framework). The specification of that external interface should be well described so that others can use it. Examples of these components in the WinMan context are the Northbound and Southbound interface managers.

C.1.2.2 Internal

This type of component has only interfaces to other components in the framework. Their existence is not visible outside the framework. Examples of these components in the WinMan context are the threshold manager and the logical tree managers.

C.1.2.3 Subordinate

This type of component has only one interface with another component in the same framework. It is a special form of an internal component. Its existence is only visible for that other component. The interface between those two components is dedicated and not published. Examples of these components in the WinMan context are the alarm correlator and the end-to-end routing components.
C.1.2.4 Wrappers

This type of component encapsulates an existing component/application/function so that it fits in the current framework. An example of this type is a database wrapper which can encapsulate a commercial database system.

C-2. Advantages of Components

When designed well the use of components can offer the following advantages:

C.2.1. Clear separation of responsibilities

When the interfaces of a component are well defined that component can be developed in complete isolation of the rest of the system. For the users of that component the component can be considered as a black box. This also means that issues like intellectual property rights etc. can be handled on a component basis instead of an application basis. This reduces the potential issues between partners when they both contributed to an application.

C.2.2. Stand-alone testing

Because the interfaces of a component are well defined it is possible to test the implementation of a component in a stand-alone environment. It is also possible to develop the test environment in parallel with the component development.

C.2.3. Free choice of implementation

The interface of a component hides the implementation of a component and therefore one is free to choose in what way (language) the component is implemented as long as the interface is implemented in a proper way. So components related to the user interface could be implemented in Java and time critical components in ANSI C.

C.2.4. Better potential for reusability

When reusability is an issue it is easier to reuse components than complete applications because applications are always tied into an environment and components can be more generic.

C-3. Advanced techniques with Components

When the concept of components is well-developed additional advantages are possible like:

C.3.1. Hot swappable components

When a component does not behave as it should or modifications are needed due to a changing environment then it is possible to replace the not-so-good component with an updated version of that component in the running system.

C.3.2. Load balancing between identical components

When an application uses a component very often and that component is becoming the bottleneck of the system it would be nice if more identical components could be added to the application so that the load can be divided over the components.

C.3.3. Load balancing between machines
The location of a component is not necessarily fixed during the lifetime of an application. This makes it possible to move components between machines to achieve efficient use of resources. It is also possible to add more processor power if the system becomes overloaded. In this way a very scalable solution can be achieved.

C-4. References

Corba Component Model Links: http://www.ditec.um.es/~dsevilla/ccm/

i Example of a ComponentManager interface: http://smartfab.ipa.fhg.de/cim/public/sematech/cim20/ ComponentManager.html (the context is quite different from network management, but the idea is the same.
