Using Policies on Management of Hybrid Networks

N Vardalachos†, E Grampin ‡, A Galis† and J Serrat‡

† University College London, ‡ Universitat Politècnica de Catalunya

Abstract: This paper describes the policy based management system devised for management of IP over WDM networks. The concepts and systems presented here have being developed in the framework of the IST Project WINMAN[1], whose aim is to develop and validate an open and flexible integrated management system for this type of networks.

1 Introduction

In recent years it has become apparent that the transportation of IP based applications will be the dominant factor in future networks. At the same time equipment for Wavelength Division Multiplexing (WDM) has matured sufficiently to give the very high capacity networks (terabit transport networks [2], [3], [4]) that will be needed for the ever-increasing amount of information. It was soon realised by the telecom industry and the research community that the convergence of those two technologies would form the solution to future networking, offering a universal, reliable and ultra-fast solution. Yet today's transport networks are primarily based on ATM and SDH technologies. Some operators started deploying WDM technology for bandwidth capacity extension between network nodes by means of point-to-point connections. Next generation network architectures focus on eliminating the intermediate layers between IP and optical, thus minimising encapsulation overheads and complexity. The next challenge in order to deploy these networks is that there is a need to manage their resources efficiently and how to migrate the existing network and management infrastructure to the new one. Management of such networks in an integrated fashion is a large research area.

Most trends in IP-WDM integration are extensions of the distributed Internet network control approach to the Optical Layer using signalling mechanisms either in an Overlay model or a Peer model. This paper proposes an alternative approach for managing Internet services over the Optical Transport Network by extending the telecom-style policy based network management approach to the IP layer over WDM. The proposed management solution has been adopted and is being investigated by WINMAN an ongoing European research and development project, whose aim is to offer an integrated network management solution for the provisioning of end-to-end IP connectivity services derived from Service Level Agreements (SLAs).

2. The WINMAN System

The WINMAN management system has been designed by applying mainly Open Distributed Processing (ODP) principles taking also into consideration the Telecommunications Management Network (TMN) framework. The TMN architecture structures the management complexity 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, enabling the design of management applications that are independent of distribution, the underlying infrastructure and management protocols.

As shown in Figure 1, the WINMAN architecture consists of an Inter-Domain Network Management System (INMS) for Configuration, Fault and Performance Management on top of two Network Management Systems (NMS) for both IP and WDM technologies. As shown in the figure, the management systems for ATM and SDH are considered as well although they are not going to be taken into consideration at the design and development phases. The INMS has open interfaces to the Service Management and the Network Management Systems of the different domains (WDM, IP, ATM, and SDH) and it is accessed by some categories of users through a GUI. The roles and actors in WINMAN are so diverse that other categories of users may prefer the access though an API instead of a closed GUI.

The development of the WINMAN architecture was based on a subset of the CORBA Component Model (CCM) [5] with extensions specially conceived for the integrated management of IP and WDM. This approach adheres to WINMAN all the benefits of the component-based technology. The components of the WINMAN systems could be distributed over a number of nodes connected by a Data Communication Network. The degree of distribution in that case is transparent to the components of the WINMAN solution. The components do not have knowledge on the location of the other components, whether they are collocated on the same node or running on a node thousands of kilometres away.

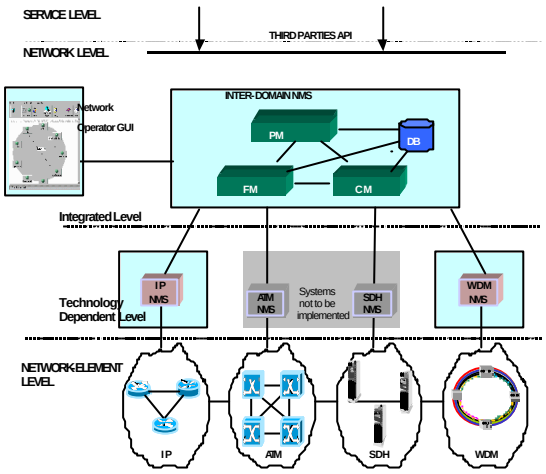

Figure 1:WINMAN Management Architecture

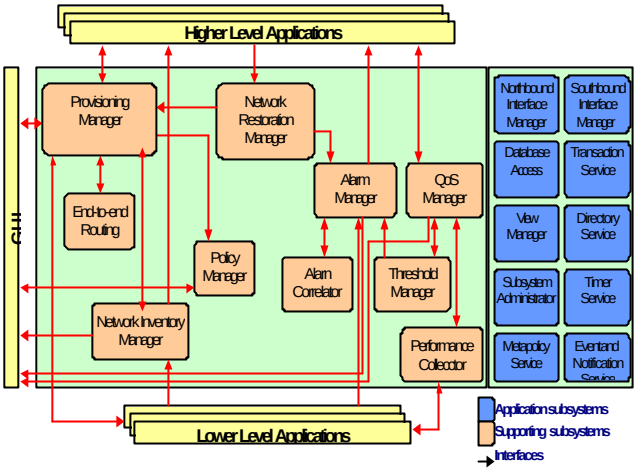

Figure 2: WINMAN Generic NMS Architecture

The INMS views all the inter-technology (inter-NMS) connections and coordinates the network provisioning, fault management and performance management between all technological domains. All these three systems share similar functionalities at their network management layer, so the approach adopted by the WINMAN project was to build a Generic Network Management System (GNMS) [6] with all the common functionalities of the three systems. The GNMS subsystems are shown in Figure 2. From that generic NMS architecture three NMSs have been specialised and further refined, and are currently being developed by the WINMAN consortium.

The Provisioning Manager is in charge for provisioning the IP services. It manages the provisioning process, including scheduling. The End-to-end Routing performs the design of the end-to-end connections inside its own network, taking into account QoS constraints and routing policies. 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.

The Policy Manager 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. The Alarm Manager receives alarms from the Lower Level Application, triggers alarm correlation, stores alarm data and distributes alarms to other systems and subsystems. The Alarm Correlator filters, correlates and evaluates the alarms to find out their root cause and generate new alarms, sending the results to the alarm manager. The QoS Manager 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. The Threshold Manager checks counters against the defined thresholds in order to generate alarms and reports if the thresholds are passed. The Performance Collector collects performance data from the Lower Level Applications. The Network Restoration Manager 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.

The WINMAN system offers a northbound interface where Service Management Systems (SMS) might plug-in and request IP connectivity services. This interface complies with the Connection and Service Management Information Model CaSMIM [7] standard from the TMForum. The system also interfaces through its southbound interface with one or more Element Management Systems (EMSs), through an interface compliant with the Multi-Technology Network Management (MTNM) standard [8].

3 The WINMAN Policy Manager

As a general remark, we have foreseen the operation of the Policy Manager as a support component that becomes active in response to internal or external events. These second type of events are to be raised by the Provisioning Manager and the End To End Routing module. The Performance Manager and the Scheduler could potentially generate events for the Policy Manager, which will be taken into account in the next system release.

Policies allow modifying system behaviour dynamically; they are persistent but can be changed on the fly; this means than the WINMAN system's behaviour can be changed without recompiling, just adding/changing policies. The insertion or deletion of a component is known to the rest of the components by means of the infrastructure naming service. As a supporting component, Policy Manager adds value to the system by providing services to the rest, but its presence shall not be a prerequisite for system operation. The idea is that Policy Manager clients (WINMAN components that use Policy Manager functionality) would have a default "permit any" policy that allows their operation in case of Policy Manager absence. If Policy Manager is running, then policy-based value is added to the system. The Policy Manager interacts with the End to End Routing and the Provisioning Manager acting as a server: the interface with these components is carried out by the class IpolicyManager which contains two methods: getAuthorisation() and get Action().

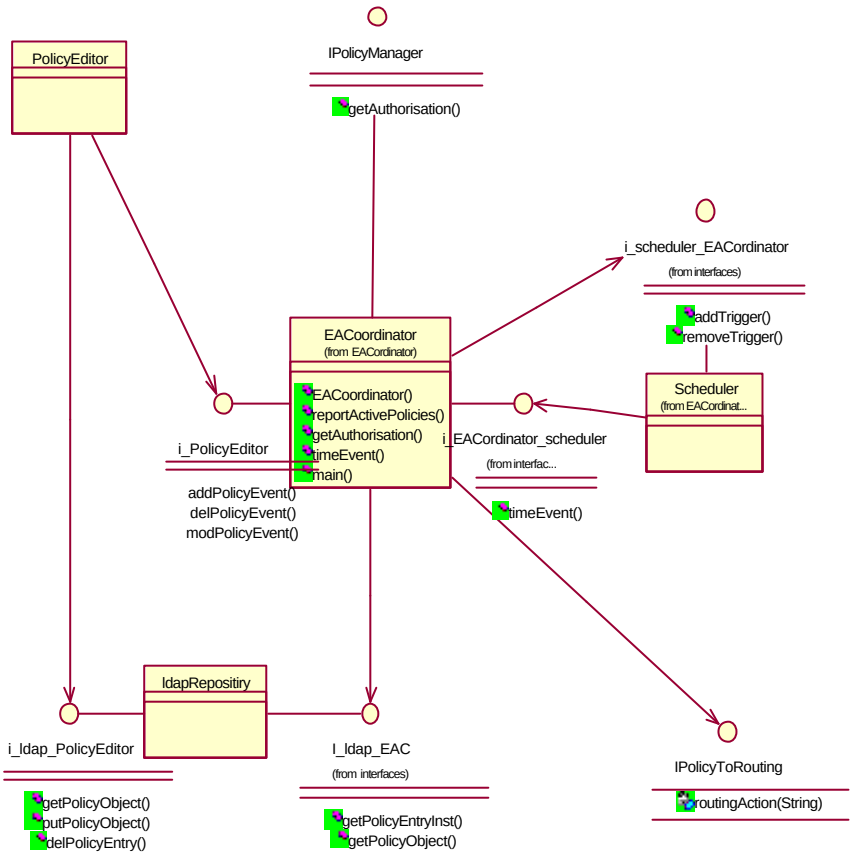

Figure 3: The Policy Manager

On the other hand, the Policy Manager also interacts with the Inventory, the EndToEnd Routing and the Provisioning Manager acting as client to them. This is done through the interfaces offered by these components, as depicted in the Class Diagram.

4 Policy Manager Entities

The IPolicyManager interface provides two main functions: the getAuthorisation and the getAction. The Provisioning Manager module executes the getAuthorisation() method when it is necessary to retrieve the policies associated with a given provisioning request. This method returns a byte indicating if the request is accepted or not, with an associated error code. This error code informs the reason why the provisioning request is not accepted, allowing the provisioning manager to take the appropriate corrective action before denying the request to the user. The getAction() method is issued by external actors, for example the WO or the Performance Manager component when some condition previously programmed is met.

When the Policy Manager receives this method it fires the PDP to verify whether the triggering conditions are met. An example of such conditions are time, existence of special traffic events (eg. a football match TV multicast), other variables reflecting the network status (e.g. some congestion threshold), etc. If some logic

function of the above-mentioned conditions is met (as specified by the relevant policy rule), the Policy Manager issues the method provisioningAction(), that's why an interface to the Provisioning Manager is needed.

The IToPolicyManager interface provides two main functions: the provisioningAction() and the routingAction(). The provisioningAction() is used in order to send the appropriate configuration commands to the Provisioning Manager in order to reconfigure the network. This network reconfiguration may entail the modification, release or establishment of several connections. The Provisioning_object input is related to obligation policies. It returns a Byte indicating if the intended configuration could be performed or not, and an error code. This configuration may be fired for example by the following policy:

'**if** Not Business Hours and Business LSPs Usage < 50%

then *Change Configuration to Non Business Hours*

else *Notify Billing for Out of Business Usage*'

This policy permits that a trivial Scheduler policy could be checked against network conditions to take the appropriate actions, that can be different than the initially intended one, in this case change network layout from business hours to non-business hours.The routingAction is used in order to send the appropriate configuration commands to the EndToEndRouting Manager in order to perform some needed routing action.

5. Conclusions

This paper gives an overview of the work carried out in the IST Project WINMAN (whose main task is to develop and validate an open and flexible integrated management system for IP over WDM networks), focusing on the policy based management approach adopted by the project. The trials in the WINMAN project have demonstrated inter-connectivity across a worldwide network management infrastructure in a multi-provider and multi-domain environment [9]. Furthermore the design and the implementation of the WINMAN policy management system are described.

Acknowledgments

This paper describes work undertaken and in progress in the context of the WINMAN – IST 13305, a two and a half years research and development project during 2000-2002. The IST programme is partially funded by the Commission of the European Union.

References

[1] WINMAN site, www.telecom.ntua.gr/winman

[2] G. Lehr, H. Dassow, P. Zeffler A. Gladisch, N. Hanik, W. Mader, S. Tomic, G. Zou; "Management ofAll-Optical WDM Networks"- IEEE/IFIP 1998 Network Operations and Management Symp.; 15.2.20.02.1998, New Orleans

[3] Draft ITU-T Recommendation G.872 (ex G.otn) "Architecture of optical transport networks" Geneva (1998)

[4] G. Lehr, R. Braun, H. Dassow, G. Carls, U. Hartmer, A. Gladisch, H. Schmid : "WDM Network Management: Experiences gained in a European Field Trial" - IEEE/IFIP 1999 Integrated Network Management Symp.; proceedings, 485-498 pp, 24-28 May 1999, Boston

[5] CORBA Component Model (CCM), http://ditec.um.es/~dsevilla/ccm/index.shtml

[6] WINMAN consortium, "WINMAN Solution description R0", August 2001

[7] "Connection and Service Management Information Model (CaSMIM) Information Agreement TMF 605", Public Evaluation, Version 1.5, June 2001

[8] "Multi Technology Network Management Information Agreement, NML-EML Interface, TMF 608", Member evaluation Version 1.0, May 2001

[9] Galis, A. (ed.), -"Multi-Domain Communication Management," - CRC Press LLC, Boca Raton, Florida, USA, ISBN 0-8493-0587-X, July 2000