The TRUMPET Service Management Architecture

L. Sacks, M. Verdier, O. Prnjat, M. Loryman, University College London, Dept. Of E & EE, UK
M. Wittig , M. M. Kande, B. Bhushan, GMD Fokus, Germany
S. Mazaher, Norwegian Computing Center, Norway
D.Maillot, Telis, France
C. Autant, N. Ganivet, Alcatel ISR, France

- Full paper -
Contact person:
Marcus Wittig, GMD FOKUS
Kaiserin Augusta Allee 31, D-10589 Berlin, Germany
Tel: +49 30 34 63 7218, Fax: +49 30 34 63 8218
E-mail: Wittig@fokus.gmd.de

For the TRUMPET Consortium (DGXIIIB, AC112)

***Abstract*:**
This paper describes the TRUMPET service management architecture. This architecture was developed to support the aims of the TRUMPET (Inter-Domain Management with Integrity) project: to investigate secure, high integrity interactions between administratively separate bodies concerned with the provisioning of broad-band telecommunications services. To allow these issues to be fully investigated a system was designed that included a number of bodies, or organisations, and in which these players interact over a mixture of technologies - specifically Java, CMIP and CORBA. These technologies were deployed so as to provide a technology independent interface information model following the TMN recommendations. The service architecture provides ATM Virtual Path connections, within specified Quality of Service parameters, across two or more Public Network Operators offering service to a Values Added Service Provider; who in turn is offering the full end-to-end or 'One Stop Shopping' service to a number of customers.

1. Introduction

This paper presents the service level architecture designed for the TRUMPET consortium. This architecture is intended to be used to explore integrity and security of open network management provisioning in the liberalised market. To this end, the design described here takes into consideration a number of factors seen as being important in the future. Each of the players in the model - the customer, the 3rd party retailer and the network provider - has a fully active management element under their control with a full featured interface with the other appropriate players. Consideration is given to security and performance issues across each of these interfaces. Technologically, a mixture of systems has been used to enable us to explore many aspects of the emerging distributed processing environments being deployed. Principle amongst these aspects is the development of a fully distributed managed object model in Java™ compatible with the TMN philosophy.

2. Background

The progressive liberalisation of the telecommunications market has placed high demands for technical innovation on both service providers and vendors. On one hand new regulation forces incumbent operators to provide access to their systems to new operators. On the other hand new kinds of operators are emerging, those who may be selling part solutions such as trunk routing or may not be network operators at all, but may be involved in retail sales of network services. The open market drives the needs for providing interconnection of bearer services between operators; this in turn drives the need for open network management platforms.

The context of the TRUMPET project is interconnection of service level management systems between network operators, customers and third party service retailers. The architecture described below has two such interconnection points. The first one is between a customer premises network management system and a third party retailer. The second interconnection point is between the third part retailer and several network service providers. For these interconnection points, the project is concerned in particular with issues relating to the integrity of these interconnections. The issues of integrity concern the correct and proper operation of the management systems when they are interconnected. This correct and proper operation can be jeopardised either through systematic faults in the operation and communications systems implemented or through intentional external attack. For the former problem area, thorough testing is required prior to the assumption of a full working system. For the latter issue, security measures need to be in place. The appropriate security issues are only touched on below, the full details of the TRUMPET security architecture can be found in [i].

Another consequence of the liberalised market is that new operators are looking at alternative technologies for supporting their network and service management tasks. Two principle demands have emerged; price and versatility. For the first part; as reliance on network provisioning - particularly data networks - becomes more pervasive for all market sectors, entrants into the market, as both consumers and providers, may be smaller and less willing to invest in large scale high end platforms. For the second part, with more competitors in the market, product differentiation becomes critical to a companies survival. To maintain a differentiated product, it is necessary for a company to be able to implement new services quickly. Until recently the assumption was that the appropriate technology for open network management provisioning should be based on OSI facilities and CMIP. However alternative technologies are being explored by the industry. Examples of these alternatives are Java and CORBA - these are included in the architecture described below. Another aspect of a companies' ability to differentiate its self is in its support of the customer.

For this reason, TRUMPET has put a great emphasis on the customer interface with the service level architecture. In the architecture described here, the combination of easily available and versatile software being made available to the customer enables the customer to smoothly incorporate the externally purchased network facilities into both its private network and to its business processes as a whole at the service management level.

2.1 The TRUMPET Trials

The above sets the scene for the work of TRUMPET. The concepts developed here will, however, be tested in real operational environments. Two trials context have been developed. The first, based at EPFL Lausanne, provides broadband connectivity between a number of medical sites, using the EXPERT test-bed located in Basel. Provisioning and control of such connectivity for the medical community is seen as very important. When transmitting mission critical information upon which good diagnostics - and a persons life - depend, the network connectivity becomes an important link in the in-

formation chain. The second trial scenario concerns the provisioning of connectivity to and between WWW cach servers. This trials will be based between a WWW server based at Salford UK, and users based on the Scottish Telecommunications IP network. Each trial requires that ATM virtual paths are established with specified quality of service parameters between the host sites, through which the user applications may establish Virtual Connections. These two scenarios give a spread of quality of service parameters and configuration requirements to test the system with.
These contexts form the base of the TRUMPET trials. Once the system is proven in these contexts, the overall implementation is expected to be extended to join sites in the UK, Switzerland, Sophia (France) and other trials developed by the MISA (DGXIIIB ACTS project AC080) collaboration.
These extended trials will prove many aspects of inter-working such as accessibility by the customer to management functionality as well as bearer services and the secured inter-working between various kinds of service providers.
This diverse and incrementally more complex scenario will allow us not only to demonstrate that high integrity and highly secure interpretation of management systems may be achieved, but will afford us an opportunity of actually measuring the performance and robustness of such systems. For this reason, some emphasis is being given to developing some test software based on TTCN [xii] to automate these measurements.

3. The Service Architecture

The TRUMPET service architecture ranges over three principle players. There are service level components in the Customer Premises Network (CPN), in the Public Network Operator (PNO) and in a third party, the network retailer - the Value Added Service Provider (VASP), as illustrated in Fig. 2. The objective is to hide detail as appropriate (illustrated in Fig. 1).

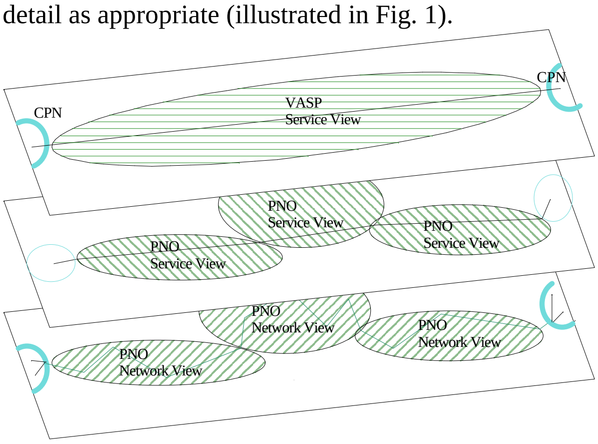

Fig. 1. TRUMPET Service Layers

Thus a customer of the VASP only needs to know about connectivity between his various CPNs; he is presented with the VASP Service view of the network. The VASP provides services from many independent Public Network Operators. As such, the PNOs present the VASP with their respective service views. These represent end to end connections between ports on their networks. Naturally, the PNOs have a detailed view of their underlying networks and it is the job of their network layer OSes to provide connectivity as offered by their service layers to the VASP.

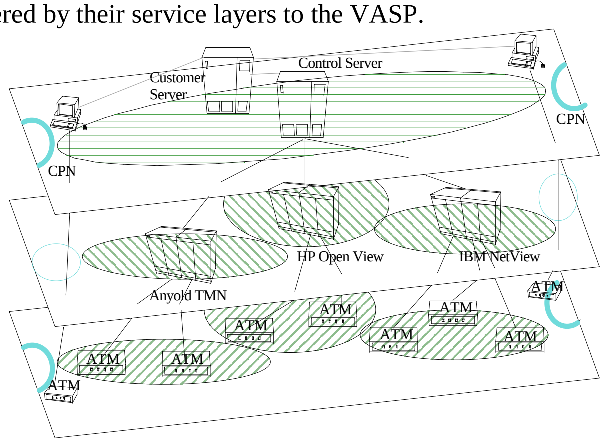

Fig. 2. TRUMPET Service Points

In TRUMPET the network layer support is taken as being provided by the host sites of the project and is not a development of the project. In practice an implementation of the ATM-Forum M4 [iii] interface for ATM network management has been deployed.
The CPN is thus the customer of - one or more - VASP, who in turn is a customer of the PNOs. In this model, the VASP is a role completely separate from the PNOs. Alternative models would have the VASP as a role played by one or other PNO[iv]. The architecture described here was chosen for a number of reasons. On one hand, the VASP as an independent third party is becoming an increasingly popular model as legislation in the US and Europe facilitates the open market. The role of the VASP is already appearing as 'Bandwidth Arbitrage' - although not automated. Thus the architecture presented here represents a possible technology to allow an automation of this market. A second motivation for this architecture can be found when compared with alternatives which require that the CPN supports the same technology as used by the PNO (in this case, a CMIP based TMN architecture, described in more detail below).
Although CMIP is the *defacto* network management protocol technology, evolving requirements for alternatives lead by the smaller operators drives requirements that alternative communications technologies are explored. The interface between the CPN and the VASP has been developed using Java based communications technology. On the other hand, the interface between the VASP and PNOs is CMIP based. With this mixture of technologies, it was also proven necessary that a CORBA gateway between the Java and the CMIP worlds is deployed. This gives us a variety of interfaces and requirements to explore for the various research aspects of the TRUMPET consortium; *viz.* security, integrity and service management.
Each of the players in the model is located within a policy domain as shown in the enterprise diagram in Fig. 3. These define domains in which certain administrative or interface policies hold. Principle amongst these for TRUMPET are the security policies. Each of the zone thus covers a particular policy for security levels of inter-domain service management interactions. Other contractual agreements may be agreed within these zones. In the figure it can be seen that the VASP participated in two interaction agreements; each of which may have different policies.

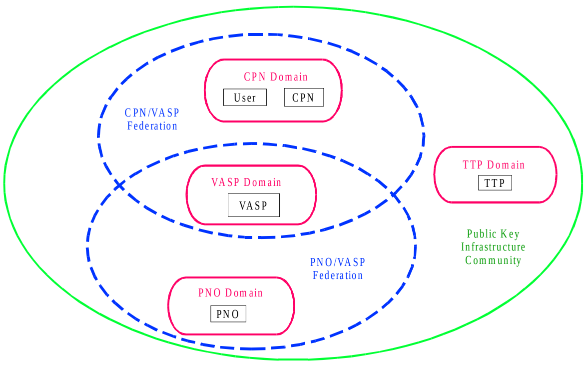

Fig. 3. Enterprise Domains

Overall, the players participate in one domain which covers wide scope agreements. The overall domain includes each player and a common Trusted Third Party authority which they each use for certification and possibly non-repudiation.

3.1 Designing the TRUMPET Architecture

The TRUMPET architecture [v] was designed using a combination of three methodologies or guide lines: the Telecommunications Management Network architecture (TMN) [vi] models, the Open Distributed Processing (ODP) concepts [vii] and techniques, and the Unified Modelling Language (UML) [viii] analysis and design notation schemes. The consideration of TMN is essential as much of the system is targeted at the world of large scale commercial public network management. TMN presents both a considered model of the components and functionality of the entities found within data and telecommunications networks and network management systems; and an engineered concept of how management functions should be supported across distributed systems. The ODP framework provides a highly structured model in which to design distributed systems. Above all, ODP requires that a system be decomposed into several divisions: the Enterprise, Information, Computation, Engineering and Technology viewpoints. One major problem with the analysis of a model built in this way is that there is often no clear mapping between elements occurring in each view point. The outcome of this is that it can be

difficult to 'prove' the system for consistency and completeness. The UML methodology presents a coherent way of expressing a software engineering design using Object Oriented techniques. Models are included for class inheritance and inter-class relationship definitions, for describing the states of classes and for defining how classes interact.
There are, thus, models in UML which fit well into the ODP viewpoints of information and computational modelling. The detailed mapping between UML diagram semantics and the ODP viewpoints is discussed in [ix]. This applies in particular to the Enterprise, Information and Computational models.

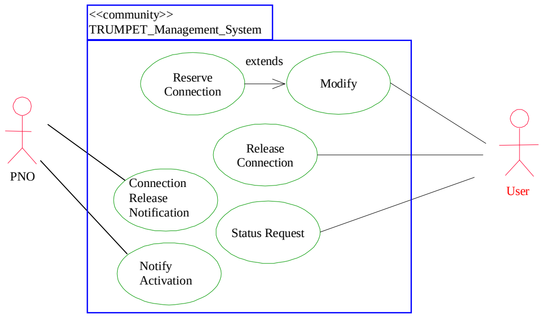

Fig. 4. The Use-Case Diagram

For illustration, an example of the kind of diagram used for the enterprise model is shown in Fig. 4. Here the desired functionality of the system is described in the form of the UML use-case diagrams depicting the scenarios of the system's use.
For the Information viewpoint, the UML static structure diagram shown in Fig. 5 shows the high level information objects in the TRUMPET architecture. Lower level diagrams further detail the low level design of the information model.

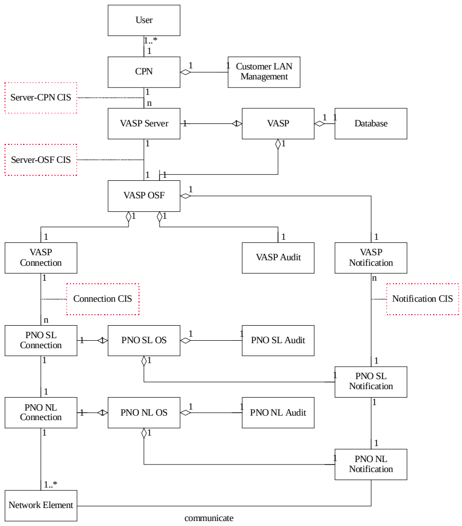

Fig. 5. High Level Information View Point

The Computational view point is principally modelled using component diagrams, showing organisation of computational objects, and collaboration diagrams showing interactions between these objects. Collaboration diagrams were built for each role in the model (CPN, VASP, PNO).
It was found that the usage of UML fitted well with that of ODP, allowing a coherent model to be built up between a number of collaborating engineers.

3.2 The Customer Premises Network Interface

The conceptual model for the Customer Premises equipment developed in TRUMPET is one of a 'glue centre'. The idea is that the interface is provided through which the Customer may interface to the VASP and also which may be used to automate interactions between the VASP and other elements within the Customers premises - the examples being local network management; local data base management of accounts or usage or local bespoke applications which might require close control of broadband network management. The motivation for this is that in realistic situations a customer network manager may have to manage many hundreds of connections and it is more than likely that this would be done through a data base system rather than a custom Graphical User Interface (GUI). Thus this architecture leaves all possibilities open. The UML static structure diagram for the computational objects necessary to support this is shown in Fig. 6.

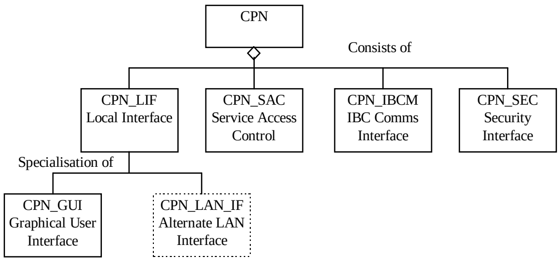

Fig. 6. The Overall Static Structure of CPN

Two popular technologies exist which may support this: CORBA and Java. Both are viable and have their own advantages. An interface based on CORBA is platform independent due to the fact that CORBA provides a platform independent transport mechanism - as does CMIP. Java achieves the same results by defining a platform within a platform; *i.e.* the Java Virtual Machine (JVM). The Java platform was selected for the CPN technology as it provides three critical facilities: LAN management interfaces (SNMP), distributed data transport and versatile user interface capabilities using its built-in GUI libraries or the WWW - the latter being suitable for distributed interfacing.

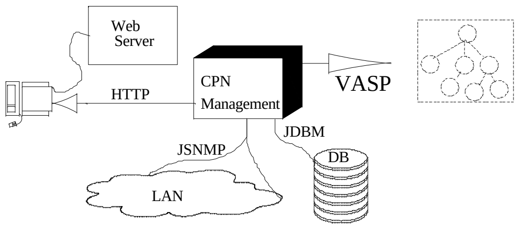

Fig. 7. The CPN Technology Model

Thus the CPN equipment (Fig. 7) consists of a JVM containing an interface with the VASP and local displays for user interfaces and for displaying events generated at the service level from the VASP and of relevance to the particular customer. The local displays can use the Java windowing libraries (*e.g.* JSNMP for SNMP management and JDBM for relational data base interfaces) or be driven through a Web Browser to achieve distributed access.

3.3 The Value Added Service Provider Platform

The VASP supports two roles: that of a provider to the Customer and that of a consumer to the public network operators. The requirements on the VASP equipment are that it supports connections from many customers through Java communications. This is achieved by dividing the VASP into three principle parts (Fig. 8); the Customer Server, the Control Server - for interfacing with the PNOs, and a MIB system for supporting the required data models.

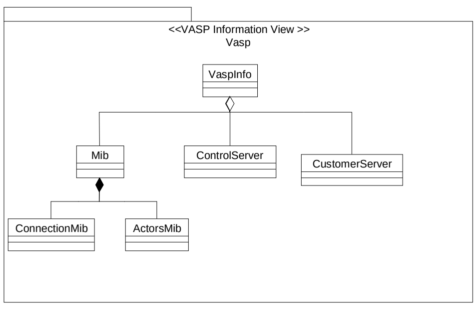

Fig. 8. The Overall Static Structure of the VASP

Internally it is required to support an information model that is both convenient to the customer and which works harmoniously with that of the PNO (which is taken to be TMN based). To meet these requirements a managed object system was constructed in Java. This provides a standard interface both to the PNOs and the CPN as well as giving a uniform information model across several players.

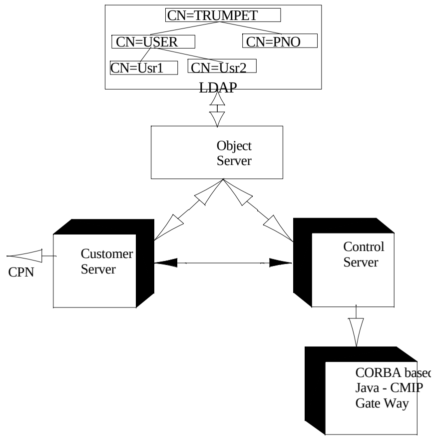

Fig. 9. The VASP Elements

The interface to the VASP was designed to be closely related to the requirements set out by TMN, based on distributed managed objects and supporting a Managed Information Base (MIB). The manager (principally the CPN, but this is used internally as well) can perform an association with managed objects in the VASP agent OSF using scoping and filtering based on X.400 (Lightweight Directory Access Protocol, LDAP [x]) as illustrated in Fig. 9. Having built the association, operations (GET, SET, DELETE, INVOKE) may be performed on the objects selected. Thus this interface not only allows the CPN manager to both work from a GUI in order to query and manipulate individual network connections; but allows bulk operations to be performed on all the connections owned by that customer - possibly driven from a database application.
The fourth component of the VASP interfaces the Control Server to CMIP through a CORBA gateway - since CMIP can be considered to be a distinct platform. The TMN model of the service elements and each of the interfaces is illustrated in Fig. 10.

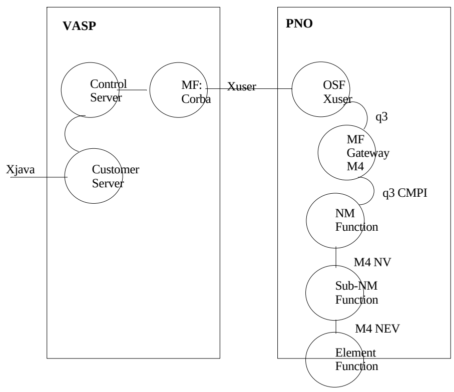

Fig. 10. The VASP / PNO OS's & Interfaces

3.4 The CPN-VASP Java Interface

Particularly important in the architecture is the distributed information model presented by the VASP. The design of this, as mentioned, has been done so as to facilitate the recommendations of TMN. This in turn places requirements both on the communication protocol and on the ability of the system to handle distributed managed objects, and in turn on the ability to make these Managed Objects persistent. There are predominantly two technologies that are used to help facilitate these aims: ObjectSpace's Voyager ORB package and the use of an LDAP Directory Server.
The Customer Server Management Information Base contains Managed Objects that contain all the information about the resources that need to be managed. It has the facility for sophisticated selection of

Managed Objects based on their properties and ensures that these Objects are persistent. There are three types of Managed Object, all of which extend the functionality of the class ManagedObject. These are Customer, VASPVPConnection and VPsegment. Selection of Managed Objects is provided for by having the structure of the MIB reflected by a Directory structure stored in an LDAP Directory Server. These LDAP entries hold the Distinguished Names of the managed objects they represent, as well possibly holding Attributes for use in filtering and scoping operations, and a variable called a Voyager Object Name that allows for the selection and invoking of operations on the corresponding ManagedObject instance. The Directory Server that is being used is Netscape's Directory Server 1.02. Also being employed is Netscape's LDAP Java SDK 1.0. This is a package of classes that allow connection to, and modification of, a Directory held in an LDAP Directory Server from Java.

The class Customer extends ManagedObject and instances of it contain all the necessary information on each Customer that connects to the VASP. Attributes it contains represent information such as the user password and the Customer ID. Instances of VASPVPConnection are created to represent an end to end Connection between two Customer Premises Networks. Attributes they contain represent information about a connection such as Bandwidth, Schedule and CPNConnectionID.

When a VASPVPConnection Object is instantiated it connects to the ControlServer and requests that the Connection it was created to represent is set up. As this is set up the ControlServer sends messages back to it telling it about the individual ATM Connections it has set up. For each of these an instance of VPSegment is created to represent it.

Communication between the elements of the Distributed Information model employed is achieved via the use of ObjectSpace's Voyager ORB technology. In order to incorporate Voyager into the operation of classes they may be processed in single step using a simple utility then they may be compiled and run exactly as normal. However, once this has been done it is possible to obtain references to objects from remote locations and methods may be called remotely on them. It is this mechanism that returns a remote reference to a CustomerService object to the CPN as the result of an associate( ) call, and it is this that is used to communicate to the CustomerServer. Communication is achieved by calling methods on the CustomerService Object such as get( ) and set( ).

Communication between the members of the MIB themselves is achieved by employing a Distributed Database and a Federated Naming Service. In this there is a Voyager enabled Directory at a central point in the CustomerServer with a well known name. All Managed Objects add themselves into this using a unique String of numbers. Objects can then get a reference to the Directory and using this identifying number get a reference to the Managed Object. This identifying number is referred to as the Voyager Object Name and is an Attribute held in the LDAP. In this way Objects can be selected as a result of filtering and scoping and their VON can be retrieved. This can then be used to get a reference to the Object and invoke operations on it. There is some flexibility in the allocation of VONs, and in the case of instances of VASPVPConection, their VON is their ConnectionID. All instances of ManagedObject and the classes that extend it are Serializable and are saved in this central Database, hence providing the stipulated persistence.

3.5 The Public Network Operator Part

The PNO Service Layer is interfaced from the VASP CORBA gateway through the Xuser interface initially defined by the MISA collaboration [iii] and derived from EURESCOM group P408. The implementation of the Xuser interface follows that of MISA very closely so as to allow a possible interworking of the two systems. The actual functionality of the Xuser is provided by an implementation of the ATM-Forum M4 interface providing network and element functionality at each PNO site.

The information model supported at each PNO site is shown in Fig. 11 for the Virtual Path (VP) connection management.

The VP Service Provider presents the entity within the PNO domain which is responsible for the provisioning of the VP connectivity service. The service may be provided to many customers represented by instances of class VP User. Within the TRUMPET management system the role of the VP User is usually taken by the VASP on behalf of its customers. A VP User may be associated with many access points representing network access points of the public network providing interfaces to adjacent network domains. Each VP user may have many VP connections which have been established by the VP Service Provider upon user's request.

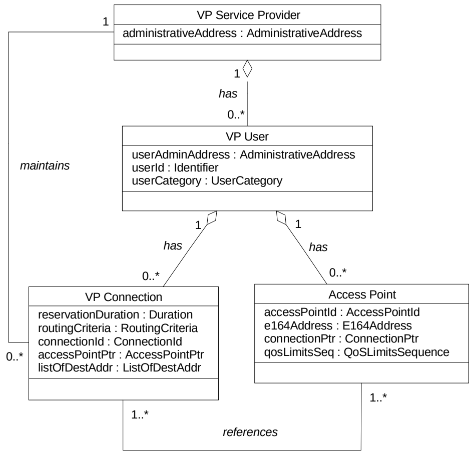

Fig. 11. PNO Information Model

4. Testing

As mentioned before, the TRUMPET project is concerned with two basic issues related to the interconnection points between open service management platforms : security and integrity. Security refers to secure, authenticated, etc. data exchange between two players in different domains, while integrity refers to the ability of interconnected management systems to retain their specified attributes in terms of performance and functionality. The focal point of the testing phase is the Xuser interface between the VASP and the PNO service management systems. The supporting communications mechanism was expanded with the addition of the security features [i]. Thus, the first aim of the testing phase is to demonstrate the correct operation of the support management communications mechanism both with and without security features. This phase aims not only to prove that the integrity of inter-domain communications is preserved when expanding the functionality by adding security, but also to measure the performance of the communications mechanism both with and without security deployed. This is to be done by developing the test-software based on the standardised test-language TTCN, which offers the possibility to verify correctness and measure time-related performance issues.

The possibility of expanding these basic test-cases so as to make them applicable over Java, as well as over the CORBA interface is being considered. This would result in a set of generic test-cases that would have the power to both verify the correct operation of a wide range of management communications mechanisms deployed in TRUMPET, and to measure performance of these mechanisms.

References:

i] TRUMPET Deliverables D3, D4; F.Gagnon et.al., A Security Architecture for TMN Inter-Domain Management, IS&N ’97.

ii] ISO/IEC 9646, Open Systems Interconnection - Conformance Testing Methodology and Framework - Part 3: The Tree and Tabular Combined Notation (TTCN), 1992.

iii] The ATM Forum, Technical Committee, Network Management, M4 Network View CMIP MIB Specification, Version 1.0, January 1997 (af-nm-0073-000).
[ATM-M4-Req] The ATM Forum, Technical Committee, Network Management, M4 Network View Interface, Requirements and Logical MIB, March 1996 (af-nm-0058-000).

iv] Initial MISA High Level Design, Annex A: Xuser Interface Definition, ACTS AC080 MISA Deliverable 3, Annex A, September 1996

v] TRUMPET deliverable A0112/UCL/WP3/DS/P/008/a2

vi] ITU-T Recommendations M3100, M3010

vii] ITU Draft Recommendation X.901, Basic Reference Model of Open Distributed Processing - Part 1: Overview and guide to use of the Reference Model, ISO/IEC, 1995

viii] Unified Modelling Language, RATIONAL Software Corporation, http://www.rational.com/

ix] M.M.Kande, M.Wittig, S.Mazaher, O.Prnjat, L.Sacks - Applying UML to Design an Inter-domain Service Management application, submitted for IS&N ’98.

x] OSI Standard X.400, LDAP: RFC 1977