Integrated Network Management: From Concepts to Application to ATM-Based Networks.

Raouf Boutaba[1] and Simon Znaty[2]

[1] Laboratoire PRISM, Université de Versailles, 45, av. des Etats-Unis, 78035 Versailles, France.

[2] Ecole Polytechnique Fédérale de Lausanne. Telecommunications Laboratory. CH-1015 Lausanne, Switzerland.

Abstract

In this paper we present structuring concepts for networked systems management. The concepts mainly deal with organizing and structuring management systems by introducing building blocks (called domains) and managed resources abstract representations (called shields). Domains are used as a means for grouping resources according to different criteria and shields are used, within domains, to reveal the relevant parts of those managed resources interfaces. As an example, we apply our concepts to structure the management of an ATM-based network. Emphasis lies on Virtual Path and Virtual Channel management.

I. Introduction

Asynchronous Transfer Mode (ATM) has been adopted by CCITT as the transport and switching technique for the Broadband Integrated Services Digital Network (B-ISDN). One reason industry has turned its attention to the management of ATM-based Broadband networks is to ensure that operations impacts and needs are identified and addressed early in the BISDN development process. Without the appropriate tools to anticipate, detect, and overcome failures, congestion and periods of degraded performance that may occur within BISDN, the viability of this broadband network will be jeopardy.

To ensure the integrated management of all resources in a networked system (in this paper emphasis lies on an ATM-based network), we define a set of structuring principles that can be used when building the management system. We make a distinction between management policies (i.e. the objectives of management) from the resources and activities being managed (i.e. managed objects). The management system may be structured into management subsystems and the management responsibilities distributed over these subsystems. We allow such a partitioning by use of *domains* as a means for grouping resources for management purposes (e.g. a domain applying a common management policy). The dependencies between the emerging domains allow to model the authority delegation process as well as peer-to-peer interactions between managers. This work has been partly conducted in the scope of the Esprit II project DOMAINS (Distributed Open Management Architecture In Networked Systems). The basic concepts of the DOMAINS project are presented in [1].

In a first section, this paper presents our structuring principles for network management. It focuses on the internal structure of management domains highlighting the shield concept : an abstract and selective representation of managed resources.

In a second section, we apply our concepts to structure the management of an ATM-based network. We also show the relationship between the traffic control process and the management process within ATM.

II. Structuring Principles for Network Management

II.1. Management Structuring

Management of large scale networked systems is a complex task due to the number and the diversity of resources and activities attached to them. In practice, this complexity may be reduced by structuring the management system into management subsystems and by distributing management responsibilities over these subsystems. This leads to multiple coexisting managements that can be autonomous or interacting in hierarchical or peer-to-peer relationships. Therefore, we have defined a methodology for management organizing and structuring which consist of two structuring principles:

- use of *domains* as the unit of management structuring,
- make a clear distinction between management policies (i.e. the objectives of management) from the resources and activities being managed (i.e. managed objects).

The domain concept provides a flexible means for grouping resources and defining management boundaries. Furthermore, it allows hierarchical structuring in that domains can be members (subordinates) of other domains. Management tasks are provided by individual domains which cooperate to achieve the overall management objectives. The interactions between domains (hierarchical or peer-to-peer) may be formally defined which make easier to set up the corresponding protocols. Another advantage of structuring the real world onto management domains is to easily support dynamic changes without disruption of the entire management system, e.g. new domains joining the system.

The domain concept has been used by a number of groups in the USA for security purposes [2], [3], [4]. They are also used by other research groups (e.g. in the DOMINO Project [5]), [6] and standards [7] for explicit grouping of resources. While in previous works domains were either managers or managed, our domains are composed by a set of managed resources but also encapsulate the components performing management. This choice allows for less flexibility when building the management system but allows the management activity to be applied the same way at all levels of the hierarchy and thus make easier its automation. Management components and resources may then be related to construct domains according to different criteria that may be relative to the contained managed resources (type, location, functionality, ownership, ...) or to the contained management components and objectives (management function, applied policy, authority, ...) [8].

II.2. Domain Internal Structure

Our goal is to ensure a uniform management of all resources and activities in a networked system. These last range from hardware resources such as lines and switches to software resources such as databases. Therefore, the second structuring principle has been defined to separate the management activity from the resources that are subject to management. This allows us to define a uniform model for the management activity in order to automate it.

As stated in the previous section our domain consists of a set of resources and a management part applying a certain policy or an aspect of a policy (see figure 1), the management part of a domain is called the *Domain Management System* ("DMS"). We identify two basic parts of a DMS: the abstract representation of the managed resources (called the *Shield*) and the managing part (called the *Kernel*). The kernel performs control actions over the managed resources through their representation in the shield. Managed resources are called *Target Resources*. Each target resource is either a real resource (e.g. a switch, a printer, etc.) or another domain of a lower level in the management hierarchy. The shield is introduced for openness and reusability purposes; it hides the resource's functional interface and reveals the management interface only. It is the domain's component which provides the demarcation between the management activity and the resources being managed.

0-7803-1820-X/94 $4.00 © 1994 IEEE

The shield has no autonomous management activities. It presents a *uniform*, *selective* and *abstract* view of the domain's target resources to the managing kernel. Indeed, the kernel may need to have a uniform view of a number of different resources. The abstract view allows to hide irrelevant details from the kernel and the selective view allows to restrict the kernel access to the part of the resource interface relevant to the management objectives of that kernel.

III. ATM-based Network Management

In this section, we show how to structure an ATM based network according to domains and develop the shield within each domain.

III.1. ATM-based Network Domains

In this example of a distributed management system we present the criteria that enable us to construct domains.

• The *Organizational* criterion is applied to the ATM-based network which management entities are distributed among ATM switches and connectionless servers (see figure 2). Note that the organizational criterion defines boundaries reflecting the organizational structure of a given system. The application of this rule leads us to consider the ATM network as a set of ATM switch domains and ATM connectionless domains.

• Within the ATM switch domain, subdomains have been defined according to the *manager functionality* criterion (see figure 3&4). Every ATM Switch which is the basic network element of B-ISDN is composed of ports. Every port supports many connections. A port identifies a virtual path (VP), a connection identifies a virtual channel (VC). Note that a VP is a generic term for a bundle of virtual channel links while a VC is a generic term used to describe a unidirectional communication capability for the transport of ATM cells. These three management domain levels, namely Switch, Port, and Connection do not share the same management functionalities, i.e. the management is performed differently on each of these domains.

In an ATM network, according to the large number of VP connections and VC connections that may be configured it is not practical to *manage* every one of them, although they will be systematically *controlled*. VP and VC connections that have relatively long or permanent holding times are mainly those that are candidate for management and are those for which domains are created.

As presented earlier, every domain is composed of one kernel (managing part), one shield (abstract representation of the managed resources) and a set of target resources (managed part).

In the next section, emphasis lies on the shield. We first present our *abstract* and *recursive* network management information model [9]. Abstraction provides an unified view of the network resources within every domain (this functionality is implemented within the shield) and enables us to model any network element whatever the management level, while recursiveness is consistent with the recursiveness of the domains. Indeed our domain encapsulates lower level domains as our information model is expressed in terms of information models of lower levels.

Then we focus on the representation of an ATM switch, an ATM Port and an ATM Connection according to our abstract information model. These modelings constitute the shield of the Switch, VP and VC domains respectively.

III.2. The Shield Element

III.2.1. The Abstract Information Model

A *Network Element* (NE) located in a particular management level, noted N, is characterized by its own *architecture*, the *service* it provides and the *NEs of the (N-1) level*. Indeed, a NE aggregates the NEs of the lower level since it depends on them in order to operate.

The *service* part is modeled as a set of management functions, i.e., fault management, performance management, etc. Every function is modeled as a set of *parameters*. These are associated with a specific flow of data that is represented by the means of a *data flow matrix*. As an example, the focus on fault management may exhibit the availability, reliability, error rate and loss rate parameters.

In terms of *architecture*, a NE can be divided into its *software* and its *hardware*. It may be either a software entity (e.g., a transport protocol) or a hardware entity (e.g., a repeater) or both (e.g., a router). Thus, the *architecture* is modeled as a composition of a software and a hardware part. The software part is composed of a *Service Access Point* (SAP) through which we access the NE, a *Connection Table* that enables the NE to have information of the NEs of the same level that can be reached by this NE. The software of a NE may be a layer protocol, thus may have (e.g., DQDB) or may not have (e.g., IP) a *management*. A *hardware* element is an *equipment* (modem, machine, etc.). Figure 5 illustrates our abstract and recursive information model.

We distinguish the MIB (Management Information Base) attributes of the network element to be managed according to their nature: static, dynamic and functional.

•*Static* information correspond to initial values concerning data transfer and management protocols. It is the manager responsibility to initialize these values. They may be changed by this manager but do not vary in time. Default buffer allocation space and timer value are examples of static objects. Static attributes are located in both entity and management classes.

•*Dynamic* information are inherent of the data transfer protocol (for instance ATM). They express the protocol operation and behavior. These may be for instance Number of PDUs received, Number of PDUs processed, Number of erroneous PDUs, etc. Dynamic attributes are located in the entity class.

•*Functional* information are concerned with resource control and management (for instance call admission control, usage parameter control). Most of the time, they are related to the control and management protocols, and functions that these protocols apply. These functional objects may be for instance an error rate computed by resource management or a current state of a management protocol. They are located in the management class.

In the next section, we refine and instantiate our abstract and recursive information model at the Switch, Port and Connection levels.

III.2.2. Application to ATM Switch, Port and Connection

The ATM switch is the network element charged with the realization of connections, based on a sophisticated allocation scheme that guaranties the required QoS of each connection.

The ATM switch management model represents the vision of OSI management for this network resource. The switch model is the abstraction of the real equipment representing its properties as they are seen from the OSI point of view.

The ATM switch is a particular case of Network Element. The switch information model aggregates *the ports* (VP) and *the connections* (VC) information models. Thanks to the abstraction and recursiveness rules that our information model matches, it can be applied to an ATM Switch, an ATM Port or an ATM Connection.

The cells arriving at the switch are buffered while waiting for their turn to be forwarded. This buffering occurs at the *port* level. The counter *ATMNb_cel_tamp* present at this level, gives an indication of the number of cells buffered at a specific instance.

The *entity* class provides us the information concerning the operational parameters at three levels of the network element: the different events dealing with the cells (number of arriving cells *(ATMPoNb_cel_r)*, lost *(ATMPoNbcel_pert)*, misinserted etc) and also the events concerning the network element (number of times that the switch or the port has been reinitialised *(ATMSwNb_reset etc)*. At connection and port levels we can find information dealing with the different types of cells arriving or leaving the switch

(Assigned, Unassigned, Metasignaling, General broadcast, Physical layer maintenance).
The values of the operational parameters dealing with the cells at the level of each particular port are obtained through aggregation of the corresponding parameters of all the connections of every particular port. For the switch we consider the aggregation of all the ports.
Our model considers three tables (connection table within the model), one for each of the three levels mentioned above:
•The *ATMSwitching Table* responsible for routing cells;
•The *ATMPort table* and the *VC connection table* providing the identifier of the switch to which the virtual path or the virtual channel is destinated.

To update the management class (functional attributes) of our information model, the consideration of ATM traffic control mechanisms is required. We briefly describe these traffic control mechanisms and describe the functional attributes that will serve the management class of the Port and Connection information models.

III.2.2.1. Network Management Versus Traffic Control

To ensure the desired broadband network performance, an ATM based network will have to provide a set of traffic control capabilities namely *Connection Admission Control* (CAC), *Usage Parameter Control* (UPC), *priority control* and congestion control functions [10].
The two basic traffic control functions, connection admission control (CAC) and usage parameter control (UPC) are necessary to guarantee QoS [11].
A CAC has to be implemented in order to achieve fairness and to guarantee required quality of service for diverse traffic sources. When a new call arrives at an ATM network, the call is admitted as long as the network can support the accepted traffic. However, connection acceptance is not sufficient as users may deviate from their negotiated terms, leading to degradation of network performance. Once a new connection is admitted by the connection admission control, a usage parameter control (UPC) function or source policing function is required to ensure that traffic submitted into the network does not exceed the parameters defined for that connection.
The user may generate different priority traffic flows by using the cell loss capability bit. If buffer overflow occurs, network elements may selectively discard cells of the lower priority flow while still meeting the network performance objectives required of both traffic flows.

In this paper, ATM network management and ATM traffic control are separated. Indeed traffic controls are performed automatically by the set of real time algorithms with response time of the order of microseconds while management tasks are inherently slow and may sometimes involve (if automated network management tools are not provided) a human manager [12]. However, these network management and traffic control operations interact each other.

III.2.2.2. Call Admission Control and VP Management

The call admission control is invoked for each link on the route between the originating and terminating point of an ATM connection. The new connection is only allowed access if it is accepted on each link on the investigated route.
VP configuration allows the network behavior to be changed dynamically as bandwidth allocated to VPs can be altered dynamically and VP routing tables and route selection can be altered to meet changing traffic requirements [13]. Therefore, the VP level is the most relevant control level. We assume in order for that solution to be feasible that the switch' buffers are subdivided at the conception level per VP and not shared among all VPs linked to that switch.
According to the information processed by the call admission controller that operates at the VP level, which information may be relevant for VP management and which interactions may occur between these two operations, respectively control and management?
The functional areas that are considered for VP management are configuration, performance, fault and accounting management. The security aspects are not within the scope of this paper.

• For *accounting management*, the information declared by the customer at the connection setup should ensure user tariffing. Thus, whenever a new call is accepted, the VP management module should be notified with details of the established call.
• For *fault management*, OAM (Operation and Maintenance) flows at the ATM level enables the VP fault manager to be notified of fault occurrence (For instance, an F4 flow indicates that a path is not available. In this case, the VP cannot be guaranteed and requires a system protection action [14] [15] [16]. The Boolean *VPunavailable* provides such information to the VP Kernel.
• For *performance management*, a certain number of performance parameters should be defined and measured in real time in order to have an accurate view of the VP behavior and thus optimize its performances. The VP manager may dispose of these parameters that are maintained in part, by the call admission controller:
These information are:
- Number of accepted connections per unit of time (*NbACut*),
- Number of rejected connections per unit of time (*NbRCut*),
- Number of accepted connections since the VP has been installed (*NbACInst*),
- Number of rejected connections since the VP has been installed (*NbRCInst*),
- Bandwidth currently available on this VP (*BdwthAv*),
- Bandwidth currently allocated on this VP (*BdwthAlloc*),
- Maximum time set-up delay (*MaxTSetupD*),
- Average time bandwidth delay (*AvgTBD*).

This list of parameters does not claim exhaustiveness.
In addition to these information, when slowly degrading VP performance occurs (transmission errors as example), **F 4** operations flows are generated, these enable performance monitoring scheme to be realized.
• VP *configuration management* concerns the bandwidth initially allocated by the manager to that VP. While a call is proposed to that VP, configuration management may check users access rights. When a call is accepted by the call acceptance module, the VP manager should provide some information for the call to be routed.
The set of attributes exhibited is located in the *management* class of the port (VP) information model.

III.2.2.3. Usage Parameter Control and VC Management

Usage Parameter Control (UPC) or source policing function ensures that during the information transfer phase, connections which try to exceed their specified traffic rates do not deteriorate the quality of service (QoS) of other connections. Usage parameter control is performed at the network access node only. Several possible actions may be taken by the policing mechanisms if the connection exceeds the agreed characteristics. These are: limit the cell stream, delay cells, mark extra cells for possible deletion by other ATM network entities, charge the user an extra amount or release the call.

• For *accounting management*, the details of contract violation should ensure user tariffing. Thus, whenever a user deviates from its negotiated terms, the VC management module should be notified with details of the violation.
• For *fault management*, OAM (Operation and Maintenance) flows at the ATM level enables the VC fault manager to be notified of

fault occurrence. For instance, an F5 flow indicates that a path is not available. In this case, the VC cannot be guaranteed and requires a system protection action [17]. The Boolean VCUnavailable provides such information to the VC kernel.

• For *performance management*, a certain number of performance parameters should be defined and measured in real time in order to have an accurate view of the VC behavior and thus optimize its performances. The VC manager may dispose of these parameters that are in part, maintained by the Usage parameter controller:

These information are:

- Rate of violations for that connection per unit of time (several classes of violations may also be created from negligible to serious) (*RVut*).

  (a global rate of violation may also be provided for the VP level, this latter aggregates all the rate of violation of VC connections that are established within this VP).
- Cell rate per unit of time that transit through this VC connection (*CRut*),
- Cell loss rate per unit of time for that connection (*CLRut*), (similarly, a cell loss rate may be provided at the VP level)
- Mean Cell Transfer Delay (through OAM cell delay measurements) (*MCTD*).

This list of parameters does not claim exhaustiveness.

In addition to these information, when slowly degrading VC performance occurs (transmission errors as example), **F 5** operations flows are generated, these enable performance monitoring scheme to be realized.

• VC *configuration management* concerns the bandwidth initially allocated to the concerned VC connection.

The set of attributes exhibited here, is located in the *management* class of the connection (VC) information model. Figures 6,7 and 8 represent our abstract and recursive information model applied to the Switch, Port (VP) and connection (VC) levels.

III.2.3. Port domains, Connection domains and their relationship

Figure 9 represents Port and Connection domains and their relationships. For every VP connection (VPC) and every VC connection (VCC) candidate for management, a domain is created within every switch along the path between the two VPC (VCC) end points. The end-to-end view of the VPCs and VCCs is provided by the ATM Network domain. The resource to be managed at the VP level (respectively VC level) is the virtual path (respectively the virtual channel). Thus an information model representing the logical view of the VP resource (respectively VC resource) is implemented within the corresponding shield.

As stated earlier, the Connection domain is seen as the managed resource from the Port domain. For instance, the Port configuration domain manages the Connection domain through connections establishment and release, while the connection domain notifies the VP domain kernel whenever the connection parameters such as throughput exceeds a specific threshold (figure 9).

A shield element that corresponds to the Port information model (without the aggregation of the Connection level) is present in order to provide a unified view of the VP to be managed by the kernel. Also one shield element is present by VC to be managed within the VP. Each of these shield elements corresponds to the Connection information model. The global shield of the Port domain is the sum of the Port shield element and the set of connection shield elements, that is the *global Port information model*. Note that this is in line with the recursiveness of our information model (see figure 5) that states that a NE aggregates the NEs of the lower level.

IV. Conclusion

In this paper, we have presented our structuring principles for integrated network management. The proposed concepts have been applied to structure the management of an ATM-based network. We have defined management domains according to structuring criteria and specified the shield of every domain, i.e. ATM Switch, ATM Port and ATM connection. For the information specification within the shield, we have applied an abstract and recursive network management information model. Also the relationships between ATM traffic control and ATM network management have been enhanced for updating this information model.

Acknowledgment

The Authors gratefully acknowledge Philips Gmbh (Germany), Roke Manor Research Ltd (UK), Synergie-Telesystemes (France), Siemens AG (Germany), AEA (UK), FIT e.V. (Germany), MARI Computer Systems Ltd. (UK) and APM (UK) and the University of Athens (Greece) for their contributions.

References

[1] M. Möller & al. "DOMAINS Basic Concepts for Distributed Systems Management". Proc. of the 5th RACE TMN Conf., 91.

[2] D. Erstin. "Controls for Interorganization Networks". in Gligor 87, pp 249-261.

[3] D. M. Nesset. "The inter-Authentication-Domain (IAD) Logon Protocol, (Preliminary Specification and Implementation Guide)". Lawrence Livermore National Laboratory, P.O. Box 808, Livermore, CA 94550, 6 May 88.

[4] D.A. Gomberg. "A Model of Inter-Administration Network User, Authentication & Access Control". Mitre Corporation, Washinton C3I Division, 7525 Colshire, Drive, McLean VA 22102, MTR-87W00003, Dec 87.

[5] M. Sloman and J. Mofett. "Domain Management for Distributed Systems". IEEE/IFIP ISINM'89.

[6] B. Wang, D. Coffield and D. Hutchison. "Database/Domain Approach to Distributed Systems Management". Comp. Comm. 89.

[7] Information Proc. Syst, OSI Systems Management Overview, ISO/IEC DIS 10040, 91.

[8] R. Boutaba, A. Benkiran. "A Framework for Distributed Systems Management". Proc. of NETWORKS 92, Int. Conf. on Computer Networks, Architecture and Application, Trivandrum (INDIA), Oct. 92.

[9] J. Sclavos, N. Simoni, S. Znaty. "Information Model: From Abstraction to Application", IEEE NOMS'94.

[10] CCITT Recommendation I371- Traffic Control and Congestion Control in B-ISDN, July 92.

[11] N. Yamanaka, Y. Sato, K. Sato. "Usage Parameter Control and Bandwidth Allocation Methods Considering Cell Delay Variation in ATM Networks". IEICE Trans. Comm., Vol. E76-B, N. 3, March 93, pp 270-279.

[12] N.G. Anessouris, A.A. Lazar, M. Tsuchida. "A Multiprocessor Architecture for Real-Time Emulation of Management and control of Broadband Networks". IEEE NOMS'92, pp 346-360.

[13] A.E. Eckberg, D.T. Luan and D.M. Lucantoni. "Meeting the Challenge: Congestion and Flow Control Strategies for Broadband Information Transport". IEEE Globecom 89, pp 1769-1773.

[14] CCITT Study Group XVIII, Recommendation I610- OAM Principles and Functions for BISDN, June 92.

[15] S. Ohta, N. Fujii. "Applying OSI Systems Management Standards to Remotely Controlled Virtual Path Testing in ATM Networks". IEICE Trans. Comm., Vol. E76-B, N. 3, March 93, pp 280-290.

[16] S.C. Farkouh. "Managing ATM Based Broadband Networks", IEEE Comm. Mag., May 93, pp 82-86.

[17] S. Arsenis, N. Simoni, S. Znaty. "An Information Model for the QoS Monitoring of an ATM Switch: from Analysis to Implementation". IBCN&S, Copenhaguen, April 93.

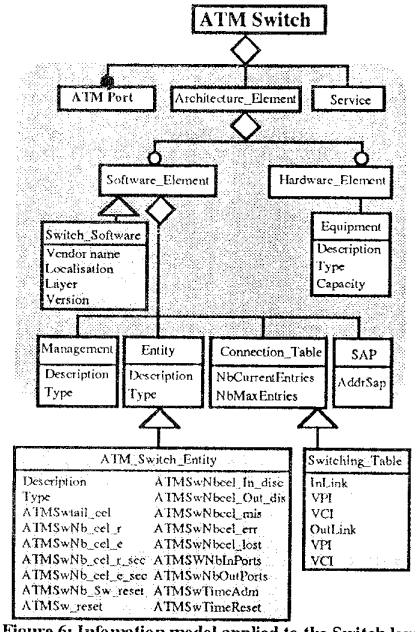

Figure 6: Infomation model applied to the Switch level

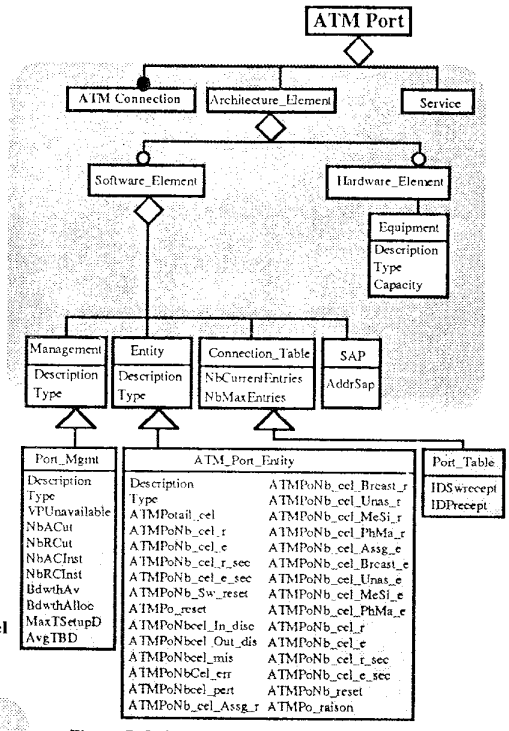

Figure 7: Infomation model applied to the Port level

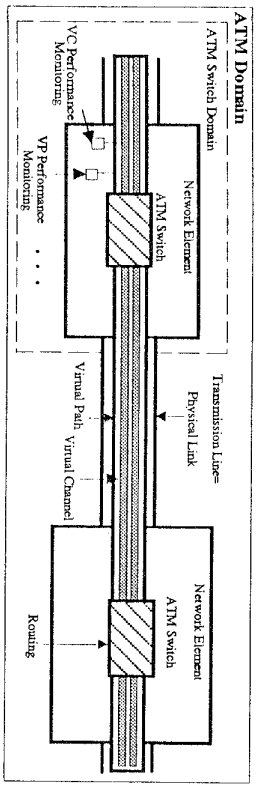

Figure 4: Structuring of the management of an ATM Network (PracticalView)

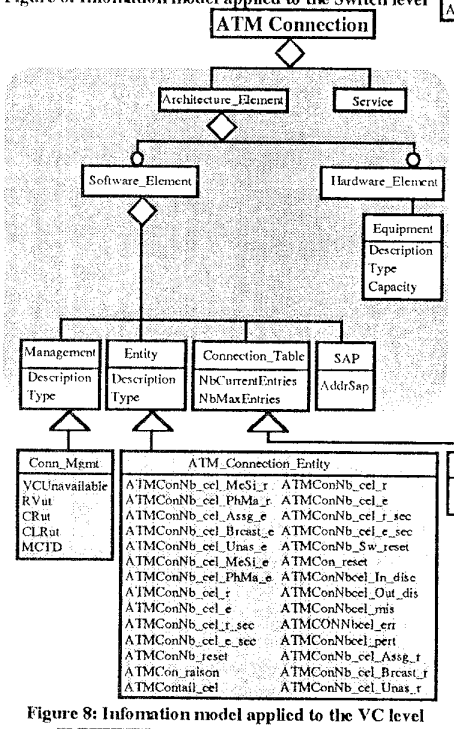

Figure 8: Infomation model applied to the VC level

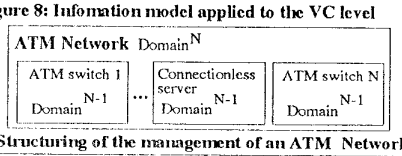

Fig.2: Structuring of the management of an ATM Network

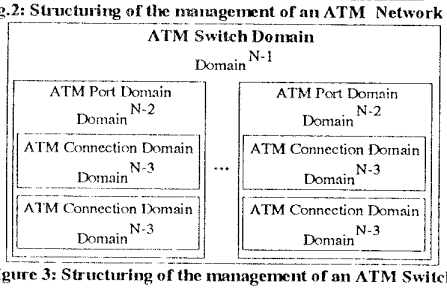

Figure 3: Structuring of the management of an ATM Switch (Abstract View)

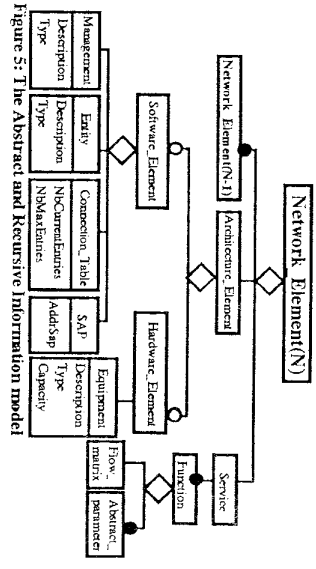

Figure 5: The Abstract and Recursive Information model

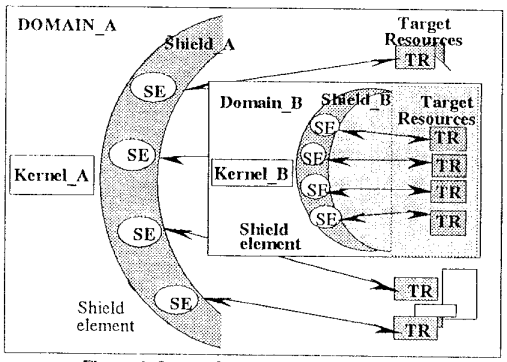

Figure 1: Internal structure of a domain

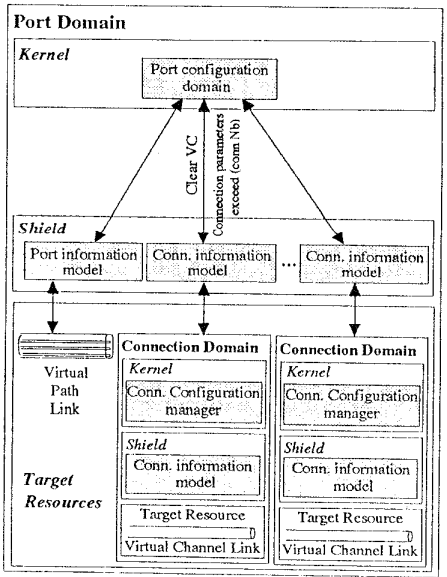

Figure 9: Relationship between particular VP and VC domains