Provision and Customization of ATM Virtual Networks for Supporting IP Services

Walfrey Ng, Raouf Boutaba and Alberto Leon-Garcia
Department of Electrical and Computer Engineering, University of Toronto
Toronto, Ontario, Canada, M5S 3G4
Phone: (416) 978-4764 Fax: (416) 978-4425
Email: {walfrey, rboutaba, alg}@nal.utoronto.ca

Abstract

This paper addresses the provision and customization of virtual networks, and discusses how a virtual network can be programmed to support different approaches for integration of ATM and IP. First, the virtual network concept is introduced, followed by the discussion of a management system architecture for the control of virtual network. Then, this paper discusses how a virtual network can be provided and customized through the partitioning of the network resources and the programmability of the management system. This paper proceeds with the discussion of how the management system can support the integration of ATM and IP in order to demonstrate the capability of full customization of network environment support. Finally, the partitioning of our NAL (Network Architecture Laboratory) testbed network is presented.

1 Introduction

The purpose of a Virtual Network (VN) is to allow a *"soft networking environment"*. Soft networking environment means flexible resource capacity, configurable network topology, and programmable network management/control architecture. For this purpose, the Virtual Network Resource Management (VNRM) Architecture has been designed at the University of Toronto's Network Architecture Lab (NAL).

The VNRM system operates the partitioning of an ATM physical network to provide ATM virtual networks to customers. In this paper, the partitioning basically involves ATM switching resources and is handled at the level of *Resource Agents*. A Resource Agent associated with a physical ATM switch mainly supports *MIBlet Controllers* and *Programmable Controllers*. A MIBlet Controller is the customer access interface to the switching resources allocated to this customer in the scope of the corresponding virtual network. Programmable Controller offers the Customer Network Management System (CNRMS) the means to download its own control and management algorithms. Through this programmability, the VNRM architecture can support full customization of network control and management. For instance, if a VN customer chooses to use a specific signaling and routing mechanism, the VNRM system is able to create such a VN environment for the customer. To demonstrate this capability of supporting full customization of network environment, we will discuss how the VNRM system can support different existing techniques for integration of ATM and IP, such as IP switching [4,5] and Classical IP over ATM [7].

2 Virtual Network Resources Management

In this section, we will briefly introduce the VN concepts and the Virtual Network Resources Management (VNRM) architecture [1]. A Virtual Network (VN) is defined as a collection of Virtual Network Resources (VNRs), where a VNR is defined as a logical subset of a Physical Network Resource (PNR). PNRs may include transmission bandwidth, buffer size, switch processing power, address space (VPIs and VCIs for ATM) or scheduling mechanism. In order to allow logical operations on network resources, PNRs in the physical domain are given logical representation to be projected into the virtual domain (Figure 1). This is called *abstraction* of network resources. Through abstraction processes, PNRs become Root[1]-VNRs. For management purposes, processing of information is organized into two layers: *network management layer* and *resource management layer*. The abstraction process in the network management layer is translated into a series of abstraction processes in the resource management layer, through which a group of PNRs becomes a group of corresponding Root-VNRs.

Figure 1 illustrates the relationships amongst a PN, VNs, PNRs, and VNRs. Through an abstraction process, a PN becomes a Root-VN. Once the Root-VN is established, multiple child VNs can be generated from the Root-VN through *spawning processes*. Spawning a VN corresponds to partitioning a group of VNRs: the notion of VNRs effectively translates the problem of creating a VN into the problem of creating a group of VNRs. It is generally assumed that the aggregated capacity of child VNRs is less than or equal to the capacity of the parent VNR. At the expense of lower Grade-of-Service (GoS), however, oversubscription of resource capacity may be allowed to VN customers in order to allow leverage sharing of resources at the level of networks.

[1] The term "Root" is used to emphasize that a Root-VNR is the very origin of other (child) VNRs.

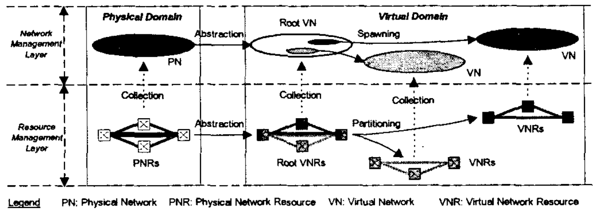

Figure 1. Physical and Virtual Networks and their Resources.

Ultimately, VNs are created for the use of customers who, in turn, provide network services to end-users. Depending upon network management/control objectives, each customer may want to have full control over network topology, resource capacity, and network management/control capability. The VNRM architecture is designed to support customized control and management of virtual networks at three different levels. At the level of resources, a Resource Agent flexibly assigns capacity to the network resource. At the level of networks, the VNRM system flexibly configures the topology of a VN. Through the concept of *dynamic system binding*, another level of customization can be exercised for VNs: by selecting appropriate communication interfaces, a VN can be associated with a control system with desirable control algorithms and mechanisms.

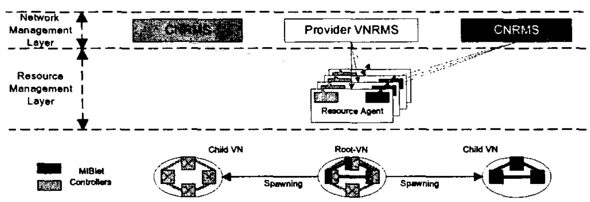

Figure 2. Dynamic System Binding

In order to allow Customer Network Resources Management System (CNRMS) to manage only a subset of resources in a network element, the Management Information Base (MIB) in the network element is logically partitioned into multiple MIBs (*MIBlets*) [2]. The MIB of the network element can be effectively identified with Root-VNRs, and the MIBlet can be identified with VNRs of a virtual network. A group of MIBlet Controllers interacts with the corresponding CNRMS for the provisioning of customer domain network services (Figure 2). By delivering abstract and aggregated information about status of MIBlets to the CNRMS, the Resource Agents effectively hide the details of the resource information that are not relevant to the CNRMS. The operations of a MIBlet Controller voluntarily abide by rules and policies of its own Resource Agent, which is in the domain of the VN service provider. During the lifetime of the virtual network, the MIBlet Controller performs usage control (or policing) to ensure conformance in the use of the VNRs to the reservation contract. Figure 2 shows one Root-VN and two child VNs. Each and every MIBlet Controller is directly accessed by the corresponding customer management/control system. Each MIBlet Controller serves its own CNRMS as an element of the virtual network for the customer.

3 Provision and Customization of ATM Virtual Network

Before discussing how ATM virtual network is provided and customized, partitioning of ATM switch's resources and architecture of Resource Agent are described in the first two subsections, which are important for the understanding of provision and customization of VN. After these two subsections, the communications and procedures for provision and customization of VN will be discussed.

3.1 Resources Partitioning

The ATM switch resources considered for partitioning are: port, VPI/VCI space and bandwidth. Three levels of granularity have been proposed in [3]: port level, VPI level and VCI level and we adopt this scheme as part of our resources partitioning.

a) VCI level: each MIBlet Controller is assigned certain VCI ranges on certain VPIs on certain ports.
b) VPI level: each MIBlet Controller is assigned certain VPI ranges on certain ports.
c) Port level: each MIBlet Controller can reserve certain ports within a switch.

For bandwidth partitioning, two bandwidth types are defined: *Hard* bandwidth and *Soft* bandwidth [2].

a) Hard: A certain amount of bandwidth on each port is reserved for each MIBlet Controller.
b) Soft: No bandwidth is reserved; bandwidth is allocated on a demand basis.

Once the MIBlet creation request is accepted, a customer management/control system can with certainty obtain the requested bandwidth if the bandwidth used by the customer system does not exceed the amount of hard bandwidth. When all the hard bandwidth is allocated, the customer system may request an additional amount of bandwidth (soft bandwidth). For soft bandwidth, there is no guarantee that the requested bandwidth can be obtained. Hard bandwidth reservation may result in inefficient bandwidth utilization, because bandwidth reserved by one MIBlet Controller cannot be used by other MIBlet Controllers, even though the MIBlet Controller that reserves the bandwidth is not using all the reserved bandwidth. However, this reservation scheme allows the MIBlet Controller to guarantee that certain bandwidth can be assigned at the moment of request. To make the hard reservation scheme more efficient, VNRMS may request different amounts of bandwidth at different times, based on the timely profile of its network traffic.

3.2 Architecture of Resource Agent

There are four functional building blocks identified in the Resource Agent: (1) Request Controller, (2) MIBlet Controller, (3) Resource Controller, and (4) Programmable Controller. The first 3 building blocks focus on how MIBlet provides CNRMS with a limited view of the switch's resources (more details of the first 3 building blocks can be found in [2]); while the Programmable Controller is responsible for the customization and programmability of the virtual network. Figure 3 shows the architecture of the Resource Agent.

The Request Controller is responsible for receiving the MIBlet creation request and examining the request against pre-configured policies. It receives the MIBlet creation request from the provider VNRMS and checks to see if there are enough resources to create the requested MIBlet. The MIBlet Controller is responsible for access control and providing the corresponding customer with a limited view of the switch's resources. The Resource Controller acts as an arbiter when MIBlet Controllers belonging to different CNRMSs have to compete with each other for resources.

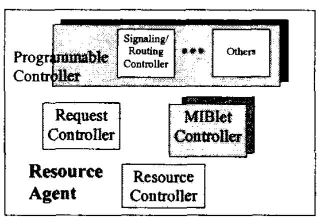

Figure 3: Architecture of Resource Agent.

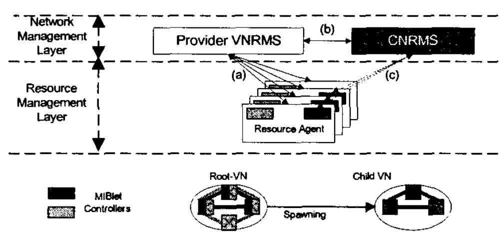

Figure 4: Interfaces for Provision and Customization.

The Programmable Controller is responsible for receiving a variety of functional software packages downloaded from the CNRMS, and these software packages are executed in the Programmable Controller. The Programmable Controller is a key building block that allows customers to have full customization of network control, and provides the Resource Agent with the intelligence for autonomous operation. Each CNRMS has its own Programmable Controller in the Resource Agent. Different kinds of software packages with different functionalities can be downloaded to the Programmable Controller. Example is the signaling and routing control software required for each Virtual Network Switch (VNS), which is a logical subset of the switch. Section 4 will show how this control software allows ATM virtual network to support IP services. Other possible software packages can be used for congestion control [12], video stream management [11], and FCAPS[2].

3.3 Interfaces for Provision and Customization of Virtual Network

In order to provide, customize and manage a virtual network, the CNRMS, the provider VNRMS and the Resource Agents need to communicate with each other (see Figure 4). Three categories of interfaces are specified: (a) communication between Provider VNRMS and Resource Agents, (b) communication between CNRMS and Provider VNRMS, and (c) communication between CNRMS and Resource Agents.

3.3.1 Communication between Provider VNRMS and Resource Agents

There are six messages involved in the communication between Provider VNRMS and Resource Agents: MIBlet creation request/response, MIBlet re-configuration request/response, and MIBlet termination request/response. The requests are sent from the provider VNRMS to the Request Controller in the Resource Agent. For each of the requests, a corresponding response is returned from the Resource Agent to the VNRMS.

MIBlet Creation: To spawn a VN, a MIBlet is created from each ATM switch involved in the VN. In particular, VPI/VCI space and bandwidth are partitioned and allocated to the MIBlet. After each switch's MIBlet is created, a VN can be simply represented as a collection of the newly created MIBlets.

[2] Fault, Configuration, Accounting, Performance, and Security

MIBlet Re-configuration: A CNRMS may require a different amount of bandwidth and possibly a different network topology at different times, based on the timely profile of its network traffic. The MIBlet re-configuration request allows the VNRMS to change the configuration during the lifetime of the MIBlet on behalf of the corresponding CNRMS.

MIBlet Termination: The provider VNRMS sends MIBlet termination requests to all the Resource Agents involved in the VN to terminate the MIBlets, if the customer and/or the provider decide to terminate the VN service.

3.3.2 Communication between CNRMS and Provider VNRMS

The messages in the communication between CNRMS and provider VNRMS are also related to creation, re-configuration and termination. However, these messages refer to the creation, re-configuration and termination of the whole VN, rather than each MIBlet in the VN.

3.3.3 Communication between CNRMS and Resource Agents

To control and manage the VN, the CNRMS interacts with the MIBlet Controllers in the Resource Agents. To manage MIBlet and monitor traffic, a management protocol, such as SNMP or CMIP, is required. Since the testbed in our laboratory uses SNMP, the protocol used for communication between CNRMS and MIBlet Controller is SNMP. Thus, the messages used by a CNRMS to interact with MIBlet Controllers are standard SNMP messages, e.g. *get-request* and *set-request*. The CNRMS accesses the MIBlet as if it is directly accessing the MIB in the switch. However, the CNRMS can only access the values related to the reserved resources. Moreover, to customize virtual network and to allow autonomous operations of the Resource Agents, CNRMS is responsible for downloading appropriate software packages to the associated Programmable Controllers in the Resource Agents.

3.4 Procedures for Provision and Customization of Virtual Network

For the provision of a VN, the CNRMS first sends a VN creation request to the provider VNRMS. After receiving the request, the provider VNRMS finds out an appropriate VN topology and determines the resource reservation of each Resource Agent involved in the VN. The VNRMS then sends the MIBlet creation requests to the Resource Agents. If the Resource Agent involved in the VN has enough resources for the creation request, it will then reserve the requested resources for the newly created MIBlet Controller. When the VN is set up successfully, the provider VNRMS sends a VN creation response to the CNRMS indicating that the VN is created. The response also provides the CNRMS with the addresses used to communicate with the MIBlets involved in the VN, and informs the CNRMS about the MIBlets' resource reservations.

For the customization of VN, the CNRMS first downloads the appropriate software packages to the corresponding Programmable Controllers, after receiving a VN creation response from the provider VNRMS indicating that the VN is set up successfully. By executing these downloaded software packages in the Programmable Controllers, the VN can support management/control functions and algorithms of customer's choice. Section 4 will show how signaling and routing software is downloaded to allow customization of ATM virtual network for supporting IP services. Also, the CNRMS communicates with the MIBlet Controller to manage the VN and to monitor the traffic characteristics of the VN. Based on the timely profiles of the traffics, the CNRMS may request for re-configuration of VN to accommodate the rapidly changing VN customer needs. Fault occurrence may also leads to re-configuration of the virtual network.

4 Customization of ATM Virtual Network for Supporting IP services

This section briefly discusses how the VNRM system can support different existing techniques for integration of ATM and IP. Existing technologies for integration of ATM and IP can be categorized into two major approaches: (1) IP over ATM, and (2) IP switching. For IP over ATM, several techniques, such as Classical IP over ATM [7] and LAN Emulation (LANE) [8], have been developed. These techniques propose to "layer" IP over ATM without changing the ATM network. These approaches hide the physical network from the IP layer by treating the ATM layer as an opaque network cloud. For IP switching category, ATM switches are changed to participate in IP routing. One example that falls into this category is the "Integrated Services Internet with RSVP over ATM short-Cuts" (ISAC) [4], developed at the University of Toronto. An ISAC router consists of a core ATM switch connected to a router controller. The controller is responsible for routing and RSVP signaling, whereas the underlying ATM switch is solely responsible for switching data packets. Other examples that fall into this category are Ipsilon's IP switching [5] and Cisco's Tag switching [6].

4.1 Virtual Network for IP Switching

In our testbed, in order to configure the virtual network for IP switching, the ISAC controller is downloaded to all the Programmable Controllers involved in the virtual network. The ISAC controller together with the MIBlet Controller are considered as a Virtual Network Switch (VNS), which is a logical subset of the ATM switch, configured to support integrated services IP. ISAC controller accesses the MIBlet Controller as if it is directly accessing the actual switch. The MIBlet Controller is responsible for access control and allows ISAC controller to use only the reserved resources.

ISAC controller need to inquire its corresponding MIBlet Controller about the reserved VPI/VCI ranges. This information helps the ISAC controller to conform to the reservation condition. Then, ISAC controller starts operating and setting up connections using only the VPI/VCI within the reserved ranges. Also, the ISAC controller only exchanges routing information with other ISAC controllers involved in the VN. This means that each controller only acquires the information of the network subset (virtual network), not the whole network.

Figure 5 shows an example of a switch's Resource Agent with two signaling/routing controllers: ISAC Controller and Ipsilon's IP switching Controller. This switch is considered as two Virtual Network Switches (VNSs), belonging to two different CNRMSs. Ultimately, it is desirable that MIBlet Controllers can provide different kinds of communication interfaces. For example, the interface between MIBlet Controller and signaling/routing controller may use protocol such as GSMP [9] or qGSMP [10], while the management interface between MIBlet Controller and CNRMS may use protocol such as SNMP or CMIP. However, our testbed currently uses SNMP for all the interfaces, because the ATM switch in our lab only supports SNMP.

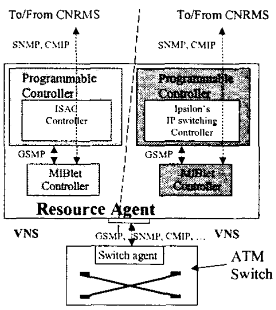

Figure 5: Virtual Network Switches for IP switching.

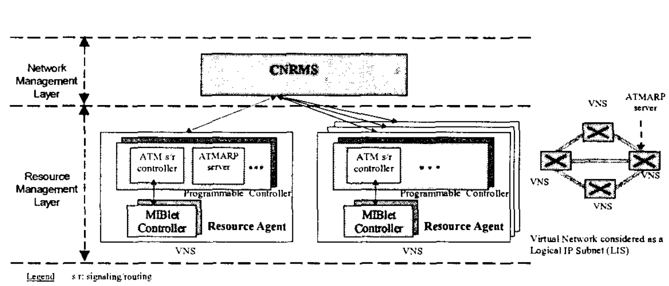

Figure 6: Virtual Network for Classical IP over ATM.

4.2 Virtual Network for IP over ATM

To configure a virtual network for IP over ATM, e.g. Classical IP over ATM or LANE, an ATM signaling/routing controller is required for each Programmable Controller in the VN. As a result, the VN can operate as a conventional ATM network, and the routing can be done using ATM Forum's private network-network interface (PNNI). For IP "layered over" ATM, the components used in IP over ATM technique are downloaded to the appropriate Programmable Controllers in the VN. The ATM signaling/routing controller requests its MIBlet about not only the VPI/VCI ranges, but also the bandwidth information, because the PNNI protocol used by the controller requires the exchanges of QoS information. Also, the ATM signaling/routing controller only exchanges routing information with other controllers involved in the virtual network. Furthermore, the controllers of different VNs use different VPI/VCI spaces to carry out signaling and routing. This allows separation amongst signaling/routing packets of different VNs.

Figure 6 illustrates an example of operating a VN as a Logical IP Subnet (LIS) using Classical IP over ATM model. In this example, the role of the CNRMS is to download not only the ATM signaling/routing controllers, but also the server component for Classical IP over ATM (i.e. ATMARP server). After a virtual network is set up by the provider VNRMS, the CNRMS first downloads the ATM signaling/routing controllers to the Programmable Controllers of all the VNSs involved in the VN. Then, the CNRMS selects one of the VNSs in the virtual network to act as the ATMARP server. After a proper location is selected, the ATMARP server software is downloaded to the Programmable Controller of the selected VNS. To resolve the addresses of nodes within the LIS, all ATMARP clients within the LIS are configured with the ATM address of the ATMARP server. The client may then use the server to request the mapping between IP and ATM addresses of any node in the LIS. In fact, the customer may choose not to implement the server component on the ATM switch, which is in the domain of the VN service provider. Instead, the customer may implement the server component in its own device attached to the ATM virtual network. In this case, the management/control system is only responsible for providing a conventional ATM virtual network and the VN users will configure the IP over ATM virtual network on their own.

5 Partitioning of NAL Testbed Network

This section introduces our testbed network to demonstrate the applicability of the VN concept. We are currently building a testbed to construct three VNs over an ATM network. As illustrated in Figure 7, each VN is to be utilized for a distinct purpose. One is to serve an Integrated Services IP (IS-IP) network, another is to serve an ATM network, and the other is to serve a Best Effort IP (BE-IP) network. ISAC [4] developed at the University of Toronto is used as the signaling/routing controller for the IS-IP virtual network in our testbed. The ISAC controller is responsible for routing and RSVP signaling. It also performs parameter mapping which maps the QoS parameters from IS-IP model to appropriate ATM services. The signaling/routing controller used for the ATM virtual network is the ATM signaling/routing controller, which we developed according to ATM signaling standard. The BE-IP VN is required for the operation of ISAC controller because ISAC switch controllers employ RSVP as a signaling protocol and RSVP delivers control messages through BE-IP services. For the time being, BE-IP control mechanisms and services provided by the NAL ATM switch are used for the control of the BE-IP VN. In other words, we are exercising programmability and resource allocation flexibility for IS-IP VN and ATM VN only.

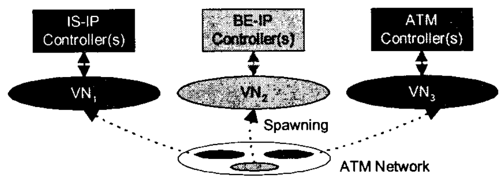

Figure 7. Partitioning of NAL Testbed.

6 Conclusion

In this paper, we have discussed the virtual network (VN) concept as a means for organizing a programmable network. We have introduced the partitioning function of virtual network resources through the provision of MIBlets. Static and dynamic partitioning schemes are combined to support VNs with different QoS requirements, and to allow for flexible resource allocation. To provide customizability and programmability of the VNRM architecture, the Resource Agent hosts programmable controllers which enable the customers to enforce a variety of management/ control mechanisms of their choice. Network control and management functions are implemented by downloading software packages into the Resource Agents.

This paper focused on the provision and customization of virtual networks. The provision of VN is considered as the provision of a group of MIBlets. The paper discussed how the CNRMS interacts with the provider VNRMS to provide and customize the virtual network. To demonstrate the capability of supporting full customization of network control, this paper discussed how the VNRM architecture supports different existing approaches for integration of ATM and IP. The ATM Forum and the Internet Engineering Task Force (IETF) have investigated several approaches that utilize the switching capability of ATM as a component of internetworking. Each approach has its own strength, and different customers may be in favor of different approaches. Through the programmability of the VNRM architecture, different approaches for integration of ATM and IP can be applied in distinct virtual networks on top of a single physical ATM network.

7 References

[1] Andrew Do-Sung Jun, Alberto Leon-Garcia, "Virtual Network Resources Management: A Divide-and-Conquer Approach for the Control of Future Networks," IEEE Global Telecommunications Conference (GlobeCom'98), Sydney, Australia, November 1998.

[2] W.F. Ng, D. S. Jun, H. K. Chow, R. Boutaba, A. Leon-Garcia, "MIBlets: A Practical Approach to Virtual Network Management", Sixth IFIP/IEEE International Symposium on Integrated Network Management (IM'99), Boston, USA, May 1999.

[3] J.E. van der Merwe and I.M. Leslie, "Switchlets and Dynamic Virtual ATM Networks," Fifth IFIP/IEEE International Symposium on Integrated Network Management (IM' 97), pp. 355-368, May 1997.

[4] HungKei Keith Chow and Alberto Leon-Garcia, "Implementation and Performance Evaluation of ISAC: Integrated Service Internet with RSVP over ATM shortCuts," ICC'98, June 1998.

[5] P. Newman, T. Lyon, G. Minshall, "Flow Labelled IP: A Connectionless Approach to ATM," Proceedings of IEEE Infocom, March 1996.

[6] Y. Rekhter, B. Davie, D. Katz, E. Rosen, and G. Swallow, "Cisco System's Tag Switching Architecture Overview," IETF, RFC 2105, February 1997.

[7] Laubach, M., "Classical IP and ARP over ATM", IETF, RFC 2225, April 1998.

[8] B. Ellington, "LAN Emulation over ATM, Version 1.0," ATM Forum, af-lane-0021.000, January 1995.

[9] "GSMP: General Switch Management Protocol", Ipsilon Networks, IETF RFC 1987.

[10] C. M. Adam, A. A. Lazar and M. Nandikesan, "QOS Extensions to GSMP", Technical Report 471-97-05, Center for Telecommunications Research, Columbia University, New York, April 1997.

[11] Amir, E., McCanne, S., and Zhang, H. "An application-level video gateway", Proceedings of ACM Multimedia '95, November 1995.

[12] Ahmed Mehaoua, Youssef Iraqi and Adel Ghlamallah, "A Self-Regulating Congestion Control Scheme Using an Intelligent Multi-Agent Architecture", Proceedings of IFIP/IEEE International Conference on Management of Multimedia Networks and Services (MMNS'97), Montreal, Canada, July 1997.