ATM PNNI INTERFACING
ISSUES WITH MPLS
NETWORKING

By

RAJPAMAL R. PETHURAJ

Bachelor of Engineering

Bharathiar University

Coimbatore, India

1999

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
August, 2001

ATM PNNI INTERFACING

ISSUES WITH MPLS

NETWORKING

Thesis Approved:

Thesis Adviser

Dean of the Graduate College

PREFACE

This Thesis addresses the interfacing issues of ATM PNNI with MPLS.

ATM (Asynchronous Transfer Mode) network supports a variety of services and ensures the Quality of Services requested and network performance. At the same time, the use of Internet Protocol (IP) has been increasing in an exponential manner. This led to problems like insufficient bandwidth and congestion. Carrying IP connectionless services over connection-oriented ATM networks was a solution to the above problems of IP. IP over ATM can be supported by many services like IPOA (Classical IP Over ATM), LANE (LAN Emulation), IP switch, tag switch and MPLS.

MPLS (Multi Protocol Label Switching) Network is such a multiple layer switching technology, that it uses the QoS properties of ATM, supports multiple protocols and can be implemented on several platforms. MPLS is a very good standard for improving the performance, reliability and scalability of IP networks by engineering deterministic route selection and providing a mechanism for traffic management.

Also, as the need to link multivendor private and public ATM grows, so does the need to configure and run those easily. This is where Private Network-to-Network Interface comes in. It is a routing protocol for ATM that allows for the exchange of routing information between ATM switches. It allows switches in a network to determine the best route to establish a connection. But, the difference between PNNI and other routing protocols is that PNNI is a source-routing protocol, rather than a hop-by-hop protocol. It allows for a greater scaling of ATM enterprises and automates the routing table process.

In this thesis, two options are provided for interfacing ATM PNNI networks to MPLS and the advantage of ATM PNNI in providing restoration, in case of link or QoS on demand failure, in comparison with ATM networks is discussed.

ACKNOWLEDGMENTS

I wish to express my deep and sincere thanks and gratitude to my advisor Dr. Jong-Moon Chung for his supervision, support, critical suggestions, and inspiration without whom this thesis would not have been possible. . My appreciation and thanks are also due to my committee members, Dr. R. K. Yarlagadda and Dr. Michael A. Soderstrand for their invaluable support, assistance, encouragement and guidance throughout my Master's program here at the Oklahoma State University.

I would like to thank the Advanced Communication Systems Engineering Laboratories (ACSEL) at the Oklahoma State University for supporting resources. I would like to thank my group members for their contribution. I would also like to thank other members of the ACSEL laboratories for their recommendations and support and my friends who made my thesis work an enjoyable and pleasant one.

Finally, I would like to thank my parents and my sister for their support and encouragement.

TABLE OF CONTENTS

Chapter Page

Chapter | Page

LIST OF FIGURES

CHAPTER I

INTRODUCTION

This chapter gives a general overview of MPLS and ATM PNNI signaling and routing protocol.

1.1 A brief overview of PNNI

PNNI (Private Network-to-Network Interface) is a routing protocol for distributing network topology information among switches and clusters of switches and a signaling protocol for defining message flows to establish point-to-point or point-to-multipoint connections. PNNI routing protocol is a dynamic routing protocol that enables the ATM (Asynchronous Transfer Mode) switches to learn about other ATM switches in the network dynamically so that any link or switch failures can be detected and the calls rerouted. The network can have any number of parallel/redundant paths without users worrying about loops in the network because an ATM switch can use the dynamically learned network topology to generate a loop-free source route. PNNI allows for interoperability of switches from different vendors and designs that span multiple organizations.

PNNI uses source routing across an ATM network. Its dynamic nature means that link failures of changing path characteristics affect the route of call requests. Rerouting can occur while simultaneously accounting for bandwidth, QoS and other aspects.

1.2 A brief overview of Q.2931 specifications

Q.2931 is a signaling specification used in Broadband Integrated Services Digital network (B-ISDN) for call and connection control. It is used in ATM for the establishment of Private Virtual Connections (PVCs). Q.2931 is a sender based signaling protocol and it allows for multiparty calls.

Q.2931 is a recommendation that specifies the procedures for establishing, maintaining and clearing of network connections at B-ISDN user network interface.

1.3 A brief overview of MPLS

Both large and small Internet Service providers constantly face the challenges of adopting their networks to support rapid growth and the demand for more reliable and differentiated services. Multi-Protocol Label Switching (MPLS) offers simpler mechanisms for packet-oriented traffic engineering and multiservice functionality and greater scalability.

MPLS integrates the control of IP routing with the simplicity of Layer 2 switching. It offers solutions for the problems of complexity of network operation, scalability of currently deployed IP over ATM model and promotes multivendor interoperatability.

1.4 Thesis outline

In this thesis, Chapter 2 provides an overview of PNNI routing and signaling specifications. Chapter 3 presents an overview of Q.2931 specifications. Chapter 4 provides an overview of MPLS. Chapter 5 discusses the procedures for VCID/VPID notification over ATM for LDP is discussed. Chapter 6 proposes two options for interoperating ATM PNNI with MPLS networks and the advantage of ATM PNNI signaling concerning path restoration is also presented. Finally, the thesis is concluded in Chapter 7.

CHAPTER II

LITERATURE REVIEW

2.1 PNNI

PNNI stands for either Private Network Node Interface or Private Network-to-Network Interface. PNNI is a routing protocol for ATM and it allows routing information to be exchanged between ATM switches. PNNI protocol can be used between private ATM switches and between groups of private ATM switches. PNNI includes two categories of protocols:

A protocol is defined for distributing topology information between switches and this information can be used for computing paths through the network. A hierarchy mechanism ensures that this protocol scales well for large worldwide ATM networks. PNNI topology and routing is based on the famous link state routing technique.

Another protocol is defined for signaling which are message flows used to establish point-point and point-to-multipoint connections across the ATM network. This protocol has mechanisms to support source routing, crankback and alternate routing of call setup in case of connection setup failure.

Here, we see a description of the PNNI routing protocol and signaling procedures over the PNNI interface.

2.1.1.PNNI Routing

The functions of the PNNI routing protocol include

- Discovery of neighbors and link status

- Synchronization of topology databases
- Flooding of PTSEs
- Election of PGLs
- Summarization of topology state information
- Construction of the routing hierarchy

2.1.1.1 Physical Network

PNNI routing is concerned with a network of lowest-level nodes. The data is passed between end systems through these lowest-level nodes. To determine the route, end systems are identified by the 19 most significant octets of ATM End System Addresses. These end systems are points of origin and termination of connections.

2.1.1.2 Lowest Hierarchical Level

If the PNNI protocol supports only a flat representation of networks, then it is necessary for each lowest-level node to maintain the entire topology of the network which leads to the problem of enormous overhead for larger networks. This overhead is reduced by the use of PNNI routing hierarchy.

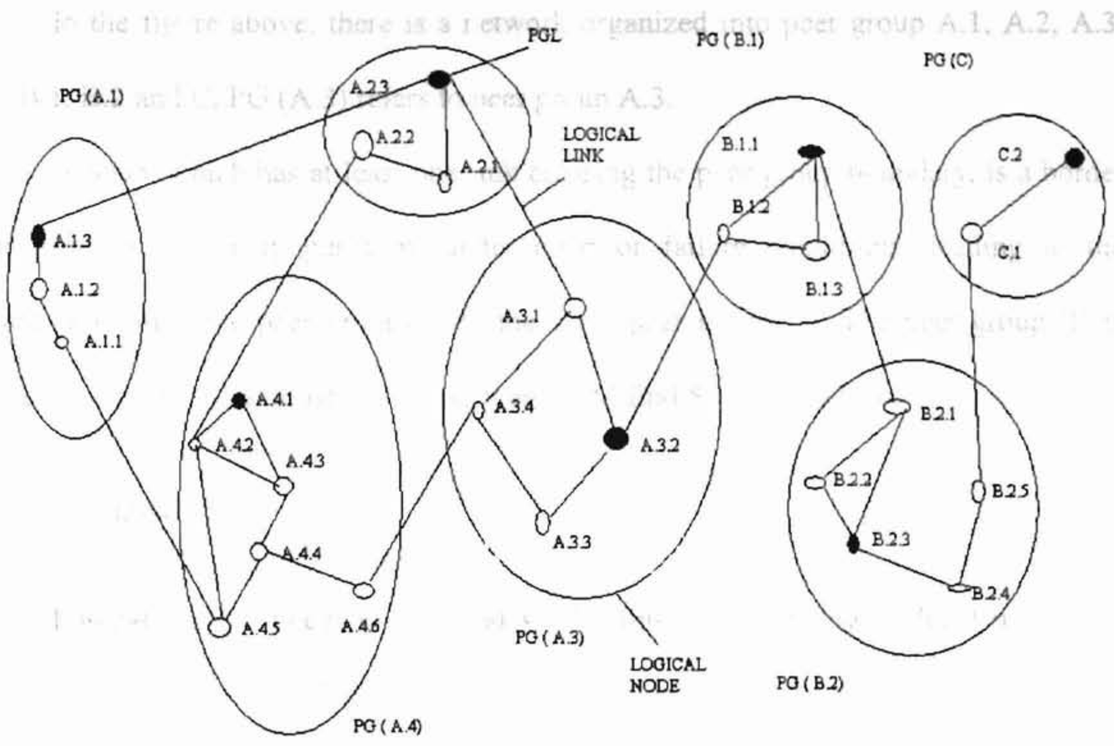

FIG .2-1 Partially configured PNNI hierarchy showing lowest level nodes.

2.1.1.2.1 Peer Groups

The lowest level nodes are organized into peer groups and this is where the PNNI hierarchy begins. A logical node in such a level is a lowest-level node. A collection of logical nodes is a peer group (PG). These logical nodes exchange information with other members of the group and this helps all members to maintain an identical view of the group.

Logical node Ids identifies these logical nodes. Peer group Ids are specified during the time of configuration and these Ids help in the identification of a peer group. Peer group Ids are exchanged by neighboring nodes in Hello Packets. The nodes belong to the same peer group if they have the same peer group ID and to different peer groups if not.

In the figure above, there is a network organized into peer group A.1, A.2, A.3, A.4, B.1, B.2 and C. PG (A.3) refers to peer group A.3.

A node, which has at least one link crossing the peer group boundary, is a border node. Peer groups may partition, under error or failure conditions, leading to the formation of multiple peer groups with the same peer group ID. The peer group ID is defined as a prefix of at most 13 octets on an ATM End System Address.

2.1.1.2.2 Logical links

Logical links connect logical nodes. The links between lowest level nodes can either be VPC or a physical link. Logical links inside a peer group are horizontal links and links connecting peer groups are outside links. As soon as a logical link becomes operational, the attached nodes initiate exchange of information by means of a well-known VCC which is used as a PNNI routing control channel. The hello packets sent by the nodes specifies the ATM end system address, node ID and its port ID for the link. Thus, two neighboring nodes learn about each other through PNNI hello protocol and this protocol runs as long as the link is operational.

2.1.1.2.3 Information exchange in PNNI

The nodes determine their local state information by exchanging Hello Packets as stated above. This state information is then bundled in PNNI topology state elements (PTSEs) and these are reliably flooded through the entire peer group.

The nodes have a topology database consisting of a collection of all PTSEs received and this information can be used to compute a route from a given node to any address reachable in or through that domain.

2.1.1.2.3.1 Nodal Information

The nodal information refers to the PTSE generated by a node that describes the node's identity, information to elect the peer group leader and to establish the PNNI hierarchy.

2.1.1.2.3.2 Topology State Information

Topology state parameters are link state parameters that describe the characteristics of logical links and nodal state parameters that describe the characteristics of nodes. PTSEs contain these topology state parameters also.

These parameters can be classified as either attributes or metrics. An attribute is considered individually when making routing decisions while a metric is a parameter whose effect is cumulative along a path. Security nodal attribute that can cause a path to be refused is an example of an attribute while delay metrics that add up along a path is an example of a metric.

2.1.1.2.3.3 Reachability information

The addresses and address prefixes that describe the destinations to which calls can be routed constitutes reachability information. These are advertised by PTSEs. Internal and external reachability information are two types of reachability information distinguished according to the source. Information derived from other protocol exchanges outside the PNNI routing domain is external reachability information while internal reachability refers to the local knowledge of reachability within the PNNI routing domain.

2.1.1.2.3.4 Initial Topology Database Exchange

Neighboring nodes in the same peer group, after the exchange of Hellos, synchronize their topology databases. This is the exchange of information between neighbor nodes that results in the two nodes having identical topology databases.

During topology database synchronization, the nodes initially exchange PTSE header information. If a node receives PTSE header information that advertises a more recent PTSE version than the one it has or one that it does not have yet, the node issues a request for the advertised PTSE and updates its topology database with the subsequently received PTSE. A link is advertised by means of PTSE transmission only after the database synchronization between the respective neighboring nodes has been completed. This ensures that the link state parameters are distributed to all topology databases in the peer group containing the link.

2.1.1.2.3.5 Flooding

The reliable hop-by-hop propagation of PTSEs throughout a peer group is flooding. It is the advertising mechanism in PNNI that ensures that each node in a peer group maintains an identical topology database.

Flooding occurs in the following manner. PTSEs are encapsulated in PNNI topology state packets (PTSPs) for transmission. The node examines the component PTSEs of a PTSP when it receives one. Each PTSE is acknowledged by encapsulating information from its PTSE header within an Acknowledgement Packet and this is sent back to the sending neighbor. If the PTSE is new or of more recent origin than the current copy in the node, it is installed in the topology database and flooded to all neighbor nodes except to the one from which it the PTSE was received.

Each node issues PTSPs with PTSEs containing update information and thus flooding is an ongoing activity. The PTSEs contained in topology databases are subject to aging and are removed after a predefined duration if they are not refreshed by new incoming PTSEs. The node that originally originated the particular PTSE can only reoriginate the PTSE.

2.1.1.2.4 Peer Group Leader

A peer group is represented in the next hierarchical level by a single node called a logical group node and a node called the Peer Group Leader (PGL) executes the functions needed to perform this. There can be a maximum of one active PGL per peer group.

A peer group leader election (PGLE) process determines the node that becomes the PGL. The node with highest leadership priority in a peer group is the PGL of that peer group. This election process is a continuously running protocol. If a node becomes active with a leadership priority higher than that of the present PGL, the election process makes the newly activated node to be the PGL. If a PGL is removed, the node with the next highest leadership priority becomes PGL. If multiple nodes have the same leadership priority, then the node with highest node ID becomes PGL.

A peer group does not require a peer group leader for its internal operations. A peer group can achieve full connectivity without a PGL.

2.1.1.3 One Hierarchical level up

Here, the representation of peer groups in the next hierarchical level is described.

2.1.1.3.1 Logical Group Nodes

A logical group node is an abstraction of a peer group for representing that peer group in the next PNNI routing hierarchy level. The functions of a logical group node are aggregating and summarizing information about its child peer group and flooding that information into its own peer group. It also passes information received from its peer group to the PGL of its child group for flooding. A node ID that contains the peer group ID of the peer group that the node is representing identifies a logical group node.

A peer group ID that must be shorter in length than its child peer group Ids identifies a parent peer group. The length of a peer group ID indicates the level of that peer group within the PNNI hierarchy. This level is referred as the level indicator. A peer group with an ID of length 'n' bits may have a parent group whose ID can be anywhere from 0 to n-1 bits in length. Also, a peer group with an ID of length 'm' bits may have a child peer group that has an identifier ranging anywhere from m+1 to 104 bits in length.

2.1.1.3.2 Feeding Information

2.1.1.3.2.1 Up the Hierarchy

A logical group node represents an underlying peer group. The PGL of this underlying group has complete topology state information from all nodes in the peer group and thus the PGL has all the required information to instantiate the logical group node. This is the upward flow of information by the PGL to the logical group node it instantiates. Reachability and topology aggregation are included in this upward flow of information.

2.1.1.3.2.2 Down the hierarchy

Feeding information down the hierarchy helps the nodes in the lower level peer to route to all destinations reachable via the PNNI routing domain. Each logical group node feeds information down to its underlying peer group. This consists of all PTSEs it originates or receives through flooding from other members of the LGN's peer group. The PGL floods each PTSE across that peer group. Therefore, every node in a peer group has a view of the higher levels into which it is being aggregated.

Thus, we see that the PTSEs flow down into and through child peer groups and horizontally through a peer group.

2.1.1.3.3 Uplinks

Border nodes are neighbor nodes that belong to different peer groups. The links between these border nodes are called outside links. The only PNNI protocol flows across these links are the Hello protocol and there is no database exchange across these protocols.

Border nodes while exchanging Hello protocols across outside links include information about their respective higher-level peer groups and the logical group nodes representing them in these peer groups. The border nodes are able to determine the lowest level peer group common to both border nodes.

The nodes are thus able to know the complete topology within its peer group and the complete topology of the higher level parent peer group. The border nodes advertise links to the higher nodes to determine which border nodes have connectivity to which higher level nodes. The node at the other end of the uplink, which is the upnode, is always a neighboring peer of one of its ancient nodes.

In some instance of Hello states, a border node must include an Uplink information attribute (ULIA) information group in the Hello packets. This ULIA includes a complete set of information about the reverse direction resources to be advertised in the uplink advertisement that the receiving node generates. The nodal hierarchy list provides the border nodes with the information necessary for them to determine their common higher level peer groups and to identify the higher level nodes to which the border nodes will declare connectivity.

Border nodes advertise their uplinks in PTSEs flooded in their respective peer groups that help all nodes in the peer group to update their topology databases with the uplinks.

2.1.1.3.4 PNNI Routing Control Channels

Neighboring PNNI nodes use a routing control channel to exchange PNNI routing information. The neighboring nodes at their lowest level of the PNNI routing hierarchy use a reserved VCC for their routing control channel. Logical group nodes use an SVCC for the routing control channel. The SVCC is established from the information derived from the uplink advertisements in the peer group represented by the logical peer group.

2.1.1.3.5 Horizontal links

The logical links between nodes in the same peer group are called horizontal links. Hellos are sent to the neighbor LGN over the SVCC-based RCC to exchange port Ids and status for horizontal links. Horizontal links are advertised only after a successful exchange of Hellos and completion of database synchronization has occurred between neighboring nodes over the RCC. PTSEs describing the new link can be flooded within the peer group containing the link and downwards to the child peer groups.

2.1.1.3.6 Topology aggregation

Topology aggregation is the reduction of nodal as well as link information to achieve scaling in a large network.

2.1.1.3.6.1 Link aggregation

The representation of some set of links between the same two peer groups by a single logical link is called link aggregation. A logical group node examines all of the uplink advertisements from its child peer group to a specific node. The uplinks to the same upnode with the same aggregation token are aggregated to a single link. This link could either be a horizontal link or an induced uplink.

2.1.1.3.6.2 Nodal aggregation

The process of representing a child peer group by a logical group node in its parent peer group is nodal aggregation. This aggregation is represented by a complex node representation in the parent peer group.

2.1.1.4 Completing the PNNI Routing Hierarchy

2.1.1.4.1 Progressing to the highest level peer group

Ever higher levels of peer groups are created until the entire network is encompassed in a single highest level peer group. This hierarchical structure is very flexible.

2.1.1.4.2 Recursion in the hierarchy

The creation of a PNNI routing hierarchy can be viewed as the recursive generation of peer group, beginning with a network of lowest-level nodes and ending with a single top-level peer group encompassing the entire PNNI routing domain. The association of the peer group Ids with the logical group nodes by configuration determines the hierarchical structure. The highest level peer group dose not need a PGL as there is no parent peer group to represent it.

2.1.1.5 Address summarization

Address summarization helps in the scaling in large networks by reducing the amount of addressing information that needs to be distributed in a PNNI network. It uses a single reachable address prefix to represent a collection of end systems and node addresses that begin with the given prefix. These reachable address prefixes can be either summary addresses or foreign addresses.

An address prefix that is either explicitly configured at that node or takes on some default value is a summary address associated with that node. An address that does not match any of the node's summary addresses is a foreign address. A native address is an address that matches one of the node's summary addresses.

2.1.1.5.1 Address scooping

The scope denotes a level in the PNNI routing hierarchy and it is the highest level at which an address can be advertised or summarized. A scope is associated with the reachability information advertised by a logical node. If a scope of an address indicates a level, lower than the level of the node, the address is not advertised by the node and vice versa. The address to be summarized with the highest scope determines the scope of the summary address while summarizing addresses.

2.1.1.6 Path selection

PNNI must select paths carefully as ATM is a connection-oriented networking technology. This implies that a path selected by PNNI will remain in use for a potentially long period of time.

When setting up a call ATM allows a user to specify Quality of Service (QoS) and bandwidth parameter values than an ATM network must be able to guarantee for that call. The selection of a path and the setup of connection state at each point along that path are the two operations in call establishment. Selection of the path is done so that the path chosen appears to be capable of supporting the QoS and bandwidth requested. Call setup at each node confirms that the resources requested are available.

Source routing and hop-by-hop routing are the two basic routing techniques used in networking. In the former, the originating source selects the path to the destination and other systems on the path obey the source's routing instruction. In the latter, each system independently selects the next hop for that path.

Hop-by-hop routing has some disadvantages when compared with source routing when used in connection-oriented networks like ATM.

The first is the creation of routing loops. These loops can be caused either due to inconsistency in routing decisions or inconsistency in routing databases among the switches. Another disadvantage is that hop-by-hop routing replicates the cost of the path selection at each system

In source routing, the source selects the path based on its local knowledge of the network topology. As only one database is involved, loops are not possible and paths will not be inefficient due to database inconsistency. In addition, the algorithm need not be the same in every system and the cost of executing it is incurred only once.

PNNI uses source routing for all connection setup requests to avoid the disadvantages of hop-by-hop routing. The first node in a peer group selects the entire path across that peer group and this is encoded as a Designated Transit List (DTL). The DTL may also specify the logical links used between nodes in addition to all nodes used in transit. If a node along a path is unable to follow the DTL due to lack of resources, the node must refuse the request and crank it back to the node that created the DTL.

The originating switch selects a hierarchically complete source route as PNNI allows for multi-level hierarchical routing. The use of source routing implies that only one node is involved in the actual selection of the path across any one peer group by any one specific connection setup request. Thus, each node may use whatever path it feels appropriate given the capabilities of the node and constraints of the call. In addition, path selection algorithm used for one level of the PNNI hierarchy can be different from the one used for a different level. Thus, PNNI does not specify any single algorithm for path selection.

2.1.1.6.1 Generic connection admission control (CAC)

A path is computed for each connection request, based on the information stored in the node's topology and the connection's service category and QoS requirements. A path is chosen only if it meets the QoS as well as the connection's end-to-end traffic requirements. During connection setup, each switching system along the chosen path performs CAC to ensure that the connection can be accommodated without jeopardizing QoS guarantees to the existing connection. If a connection is accepted by a switching system, its ability to accept future connections may change. This triggers the origination of new PTSE instances describing the node's updated resource availability.

A generic CAC is needed in the path selection process to determine if a link or node is likely to have enough resources available to support the proposed connection. A generic CAC must include a link or node is the switching system is likely to accept the proposed connection and exclude a link or node if the system is likely to reject the new connection.

Each switching system is free to use any CAC but is required to advertise a set of topology state parameters carrying some information that can be used by the generic CAC to make its prediction.

2.2 PNNI Signaling Protocol Model

The following figure depicts the relationship between the signaling procedures and the underlying services. The signaling layer contains two entities: the PNNI Call Control which serves the upper layers for functions such as resource allocation and routing and the PNNI Protocol Control which provides services to the PNNI Call Control and processes the actual signaling finite state machines using symmetric procedures.

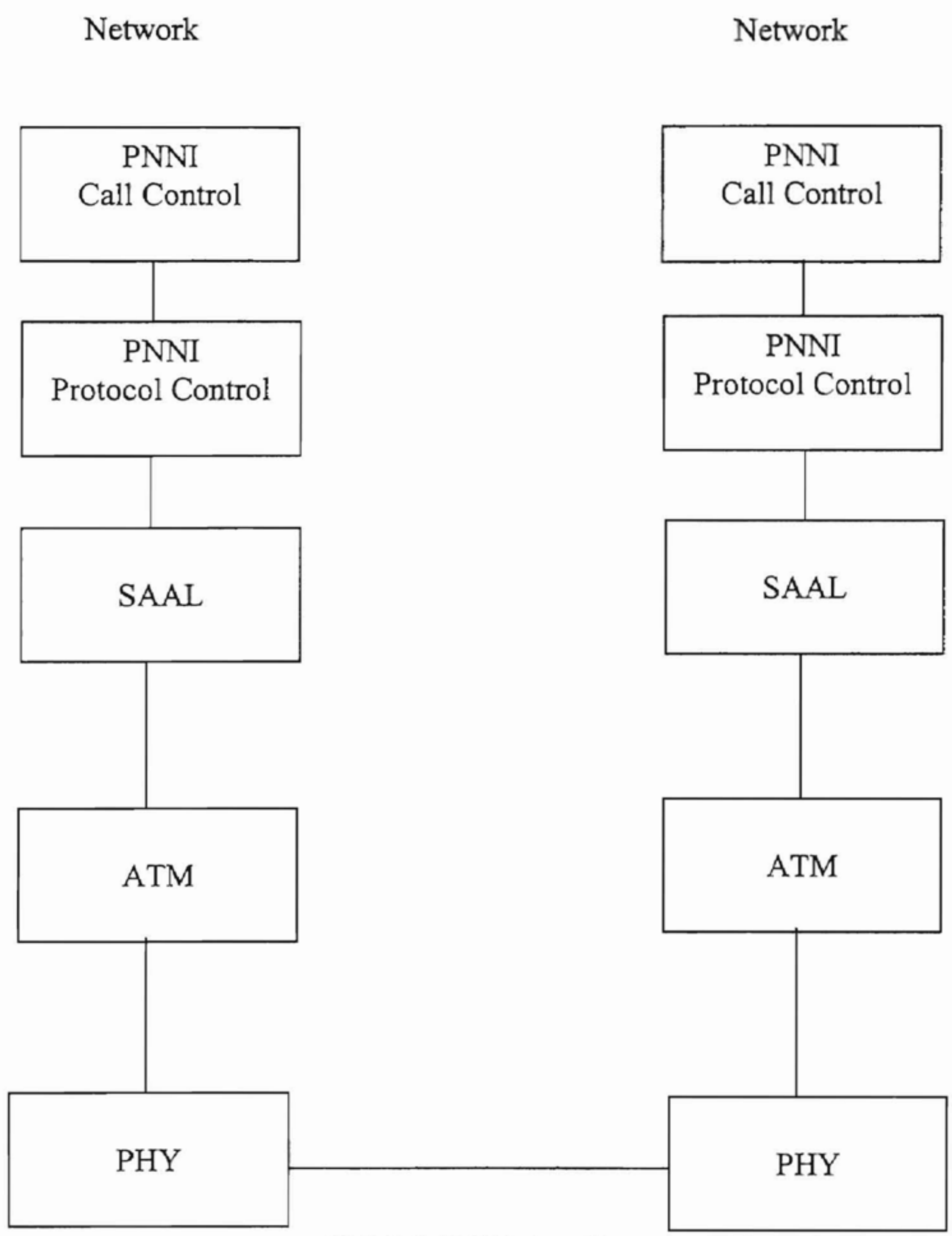

FIG.2-2 PNNI signaling protocol stack

2.2.1 PNNI Signaling

PNNI signaling is based on a subset of UNI 4.0 signaling. It does not support some UNI 4.0 signaling features but adds new features which pertain to the use of PNNI routing for dynamic call setup. PNNI signaling makes use of the information obtained by PNNI routing. It uses the route computations derived from the reachability, connectivity and resource information dynamically maintained by PNNI routing. These routes are computed as needed from the node's view of the current topology. The main concepts of PNNI signaling are to complete source routing across each level of hierarchy and reroute a signaling message around the failed path by using crankback messages. In order to meet the above requirements, PNNI signaling uses the following additional features beyond those defined for UNI signaling. They are Designated Transit Lists (DTLs), crankback routing, alternate routing and associated signaling.

2.2.2 Associated Signaling

Associated signaling is used for PNNI operation over virtual paths. If no virtual path connections (VPCs) are configured for use as logical links on the interface, all the virtual paths on the physical interface are controlled by the signaling channel on VPI=0. PNNI also supports signaling and routing over multiple VPCs to multiple destinations through a single physical interface where each VPC configured for use as a logical link has a signaling channel associated with it. Only the associated signaling channel of that particular VPC controls virtual channels within these VPCs.

2.2.3 Designated Transit Lists

While processing a call, PNNI signaling may request a route from PNNI routing and These routes are specified using Designated Transit Lists (DTLs). A DTL is a complete path across a peer group which consists of a sequence of node Ids and optionally port Ids required to traverse the peer group. The source node or an entry border node to a peer group provides the DTL. A hierarchically complete source route represents a route across a PNNI routing domain that includes each hierarchical routing level between the current level and the lowest visible level in which the source and destination are reachable. This is expressed as a sequence of DTLs ordered from lowest to highest peer group level and organized as a stack following the structure of last in first out. The DTL that corresponds to the lowest level peer group is the DTL at the top of the stack.

2.2.4 Crankback

A node uses the currently available information about resources and connectivity to create a DTL. However, that information may be inaccurate for a number of reasons. The reasons maybe due to hierarchical aggregation and changes in resource availability due to additional calls placed since the information was produced. Because of this, a call being processed according to the DTL may be blocked along its specified route. Crankback and alternate routing is a mechanism for adapting to this situation in place of clearing the call back to the source. When a call cannot be processed according to the DTL, the call is cranked back to the creator of that DTL indicating the problem. This node may choose an alternate path to progress the call or may further crankback the call. An alternate path, if chosen, must obey all received higher-level DTLs and must try to avoid the blocked nodes or links. The crankback process is repeated until either an alternate path is found or the source node of the complete end-to-end path is reached.

2.2.5 Soft PVPCs and PVCCs

A PVPC or PVCC is a permanent virtual path connection or virtual channel connection and is established administratively rather than on demand. A soft PVPC or

PVCC is one where the establishment within the network is one by signaling. The network management system provisions one end of the soft PVPC or PVCC with the address identifying the egress interface from the network and the calling end has the responsibility for establishing, releasing and also reestablishing the call if needed.

2.1.3 An example of connection setup and crankback

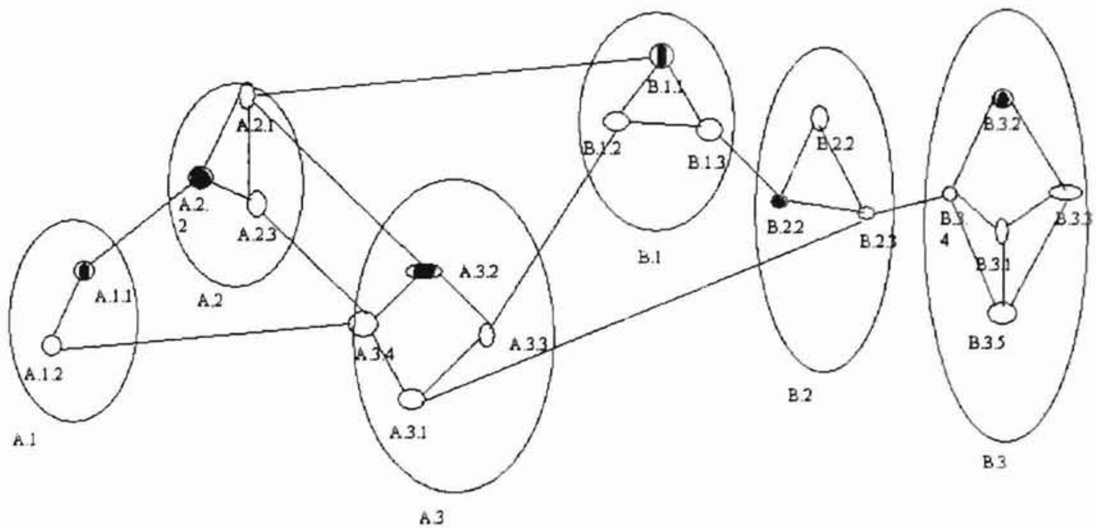

FIG 2-3. A sample network at the lowest level

Assume a connection to be setup from an end system A.1.2.x attached to A.1.2 to one attached to B.3.3.y attached to B.3.3. A.1.2.x sends a setup request to A.1.2. A.1.2 sees that the destination can be reached through node B.

Examining the topology, A.1.2 finds the following 3 paths

1. A.1.2, A.1.1, A.2, A.3, B

2. A.1.2, A.1.1, A.2, B

3. A.1.2, A.3, B

Let us assume links between A.3.3 and B.1.2 are blocked and also that A.1.2 chooses the first path-A.1.2, A.1.1, A.2, A.3, B

Three DTLs are built in a stack.

DTL: [A.1.2, A.1.1], pointer-2

DTL: [A.1, A.2, A.3], pointer-1

DTL: [A, B], pointer-1

Each DTL list nodes that the call setup needs to visit at a given hierarchical level. The current transit pointer following each DTL specifies which element in the list is currently being visited at that level, except the top DTL, which is advanced by one to indicate which node will be visited next.

Before forwarding the call setup A.1.2 stores the content of the SETUP message. A.1.2 forward the call setup to neighbor A.1.1. A.1.1 examines the top DTL and notices the DTL pointing to it's own Node ID. It looks for the next entry in the top DTL, but finds it exhausted. Looking at the next DTL, next destination is A.2. A.1.1 starts looking how to get to A.2 and finds an immediate neighbor in A.2, so it removes the top DTL and advances the current transit pointer in the next DTL:

DTL: [A.1, A.2, A.3] pointer-2

DTL: [A, B] pointer-1

A.1.1 did not make any routing and so does not store a copy of the SETUP message and forwards it to A.2

A.2.2 looks at the top DTL and sees that the current destination is A.2. As A.2.2 is in A.2, it looks at the next entry in the DTL and starts routing to A.3. Analyzing, the topology, A.2.2 fins the path is through A.2.3 and so pushes that DTL onto the list:

DTL: [A.2.2, A.2.3] pointer-2

DTL: [A.1, A.2, A.3] pointer-2

DTL: [A, B] pointer-1

As A.2.2 added a DTL to the stack, it stores the contents of the revised SETUP message before sending it to A.2.3

A2.3 determines that the top DTL target has been reached and then the DTL is exhausted. It notices that the next destination is a neighbor, so A.2.3 removes the top DTL and advances the current transit pointer in the next DTL:

DTL: [A.1, A.2, A.3], pointer-3

DTL: [A, B] pointer-1

The setup arrives at A.3.4. As it is in A.3, the target has been reached and leaves the target as B. A.3.4 builds a route to B and pushes a new DTL to the list.

DTL; [A.3.4, A.3.2, A.3.3] pointer-2

DTL: [A.1, A.2, A.3] pointer-3

DTL: [A, B] POINTER-1

A.3.2 receives the call setup and advances the current transit pointer and forwards it to A.3.3. A.3.3 then decides to forward it to his neighbor in B after removing the top two DTLs, as both have been exhausted.

DTL: [A, B] pointer-2

At this point, the call setup is blocked as A.3.3 finds out (either by CAC or from RELEASE messages) that the link is blocked. The crankback procedures now begin, with A.3.3 sending back out the preceding interface a RELEASE message indicating that the call was blocked between A.3.3 and B.

This message is received at A.3.3, which did not create any DTLS for this call, so crankback proceeds further. After tearing down the succeeding virtual channel, the RELEASE message is sent out the preceding interface in the direction of A.3.4.

A.3.4 specified a DTL for this call and so attempts alternate routing. It finds two alternate paths, one through A.3.1 and to B and other through A.3.2, A.3.3, and A.3.1 and to B. However, the link between A.3.1 and B is blocked and so, A.3.4 decides to continue the crankback procedure. A.3.34 sends another RELEASE message back through the preceding interface.

The RELEASE message is received at A.2.3 and this did not create any DTLs and so crankback proceeds further.

The message arrives at A.2.2 and this sends it to A.1.1.

A.1.1 did not create any DTLs for this call and crankback it further to A.1.2

A.1.2 receives the Crankback message and attempts an alternate route. It deices to try the path A.1.2, A.1.1, A.2.B forming:

DTL: [A.1.2, A.1.1] pointer-2

DTL: [A.1, A.2] pointer-1

DTL: [A, B] pointer-1

A.1.2 forwards the call setup to A.1.1, which removes the top DTL from the stack, advances the current transit pointer in the next DTL and forwards the call to A.2.2.

A.2.2 finds a path through A.2.1 to B, pushes a new DTL onto the stack and forwards it to A.2.1:

DTL: [A.2.2, A.2.1] pointer-2

DTL: [A.1, A.2] Pointer-2

DTL: [A, B] pointer-1

A.2.1 receives the setup and notes that B is a neighbor. He removes the top two DTLs from the stack and advances the current pointer in the next DTL and forwards the setup to B.1.1

DTL: [A, B] pointer-2

B.1.1 receives the setup and sees that the current DTL destination has been reached. It builds a new source route to the destination and pushes on two new DTLs:

DTL: [B.1.1, B.1.3] pointer-2

DTL: [B.1, B.2, B.3] pointer-1

DTL: [A, B] pointer-2

B.1.1 forwards the setup to B.1.3, which removes the top DTL, advances the DTL pointer and hands it to neighbor B.2. B.2.2 notices that he is the current target and finds a path through B.2 to B.3 and pushes that on the list:

DTL: [B.2.2, B.2.1, B.2.3] pointer-2

DTL: [B.1, B.2, B.3] pointer-2

DTL: [A, B] pointer-2

B.2.2 then forwards the setup to B.2.1

B.2.1 looks at the tip DTL, advances it and forwards the setup to B.2.3

B.2.3 removes the DTL, advances the pointer and forwards it to his neighbor in B.3

B.3.4 build a new DTL, giving

DTL: [B.3.4, B.3.1, B.3.3] pointer-2

DTL: [B.1, B.2, B.3] pointer-3

DTL: [A, B] pointer-2

B.3.1 receives it and it looks at the top DTL, advances the current transit pointer, and forwards the setup to B.3.3

B.3.3 determines that it is the DTL terminator for the DTL as

1. All DTLs are the end
2. B.3.3 is a lowest level node
3. B.3.3 has reachability to the destination

B.3.3 then removes the DTL stack from the SETUP message and forwards the message across the UNI to the destination.

CHAPTER I11

LITERATURE REVIEW

3.1 Recommendation Q.2931

This chapter talks about the procedures for establishing, maintaining and clearing of network connections at the Broadband Integrated Services Digital Network (B-ISDN) user-network interface. It specifies the essential features, procedures and messages needed for call and connection control.

Recommendation Q.2931 specifies the layer 3 call/connection states, messages, information elements, timers and procedures used for the control of B-ISDN point-to-point on-demand calls on virtual channels. These connections are established in real-time using signaling procedures. These demand connections can remain active for an arbitrary amount of time but are not automatically re-established after a network failure.

The procedures specified by this Recommendation are applied at the interface between a B-ISDN terminal equipment and B-ISDN public network and at the interface between a B-ISDN customer network and a B-ISDN public network.

3.2 Capabilities supported by this Recommendation

3.2.1 Protocol support for basic signaling functions

The signaling protocol supports the following basic functions at the UNI interface:

- Call/Connection Set-up

This aspect supports the establishment of a call or connection between different parties. It includes Call/Connection Request and Call/Connection Answer.

- Call/Connection Request

This protocol function allows an originating party to request the establishment of a call or connection to a certain destination. Here, the originating party may provide information related to the call or connection.

- Call/Connection Answer

This function allows the destination party to respond to an incoming call or connection request. The destination party may include information relating to the call or connection.

- Call/Connection Clearing

This function allows any party involved in a call or connection to initiate its removal from an already established call or connection. It also allows a destination party to reject its inclusion in a call or connection.

- Reason for clearing

This function allows the clearing party to indicate the cause for initiating its removal from a call or connection.

- Out-of-Band Signaling

This function specifies that call or connection control information will use a channel different from the channels used for exchanging information between end-parties. It means that a specific VPCI/VCI value will be used for the call or connection control signaling channel.

3.2.2 Support of Class A, Class C and Class X

Class A service is a connection-oriented, constant bit rate ATM transport service and has end-to-end timing requirements. The user chooses the desired bandwidth and appropriate QoS in the SETUP message to establish a Class A connection. This service may require stringent cell loss, cell delay and cell delay variation performances.

Class C service is a connection-oriented variable bit rate ATM transport service. The user chooses the desired bandwidth and QoS with appropriate information elements in a SETUP message to establish a Class C connection. This service has no end-to-end timing requirements.

Class X service is a connection-oriented ATM transport service where the AAL, traffic type and timing requirements are user defined. The user chooses only the desired bandwidth and QoS with appropriate information elements in a SETUP message to establish a Class X connection.

3.2.3 VPCI/VCI support

This Recommendation supports the VPCI as the way of identifying the virtual path across the UNI, with a condition that there is a one-to-one mapping between VPCI and VPI.

3.2.4 Support of error recovery

The error recovery capabilities supported by this Recommendation include:

1. Detailed error handling procedures, including means for one signaling entity to inform its peer when it has encountered a no-fatal error.
2. Procedures for recover from signaling AAL reset and failure
3. Mechanisms for signaling entities to exchange state information for calls and interfaces and to recover gracefully if there is a disagreement.
4. Capability to force calls, VCCs and interfaces to an idle state either due to result of server errors or manual intervention.
5. Cause and diagnostic information for fault resolution provided with call clearing, non-fatal errors and recovery from errors affecting the whole interface.
6. Mechanisms to recover from loss of individual messages.

3.2.5 Signaling interworking with Narrowband-ISDN (N-ISDN)

This recommendation supports interworking with N-ISDN. Signaling is also specified to support N-ISDN services in a B-ISDN environment.

3.3 Overview of call/connection control

Here, the call or connection states that individual calls may have are defined. As several B-ISDN calls or connections may exist simultaneously at a user-network interface and each call or connection may be in a different state, the state of the interface cannot be unambiguously defined.

Here, "incoming" and "outgoing" are used to describe the B-ISDN call as viewed by the user side of the interface.

3.3.1 B-ISDN call/connection states

The call or connection control states for B-ISDN calls are defined here.

3.3.1.1 Call/ connection states at the user side of the interface

3.3.1.1.1 Null:

No call exists.

3.1.1.2 Call initiated:

This state exists for an outgoing call when the user requests call establishment from the network.

3.3.1.1.3 Outgoing call proceeding:

This state exists for an outgoing call when the user has received acknowledgements that the network has received all call information necessary to effect call establishment.

3.3.1.1.4 Call delivered:

This state exists for an outgoing call when the calling user has received an indication that remote user alerting has been initiated.

3.3.1.1.5 Call present

This state exists for an incoming call when the user has received a call establishment request but has not yet responded.

3.3.1.1.6 Call received

This state exists for an incoming call when the user has indicated alerting but has not yet answered.

3.3.1.1.7 Connect request

This state exists for an incoming call when the user has answered the call and is waiting to be awarded the call.

3.3.1.1.8 Incoming call proceeding

This state exists for an incoming call when the user has sent acknowledgements that the user has received all call information necessary to effect call establishment.

3.3.1.1.9 Active

This state exists for an incoming call when the user has received an acknowledgement from the network that the user has been awarded the call. This state exists for an outgoing call when the user has received an indication that the remote user has answered the call.

3.3.1.1.10 Release request

This state exists when the user has requested the network to clear the end-to-end connection and is waiting for a response.

3.3.1.1.11 Release indication

This state exists when the user has received an invitation to disconnect as the network has disconnected the end-to-end connection.

3.3.1.2 Call/connection states at the network side of the interface

3.3.1.2.2 Null

No call exists in this state.

3.3.1.2.2 Call initiated

This state exists for an outgoing call when the network has received a call establishment request but has not responded.

3.3.1.2.3 Outgoing call proceeding

This state exists for an outgoing call when the network has sent acknowledgements that the network has received all call information necessary to effect call establishment.

3.3.1.2.4 Call delivered

This state exists for an outgoing call when the network has indicated that remote user alerting has been initiated.

3.3.1.2.5 Call present

This state exists for an incoming call when the network has sent a call establishment request but not received a satisfactory response.

3.3.1.2.6 Call received

This state exists for an incoming call when the network has received an indication that the user is alerting but has not received an answer yet.

3.3.1.2.7 Connect request

This state exists for an incoming call when the network has received an answer but has not yet awarded the call.

3.3.1.2.8 Incoming call proceeding

This state exists for an incoming call when the network has received acknowledgements that the user has received all call information necessary to effect call establishment.

3.3.1.2.9 Active

This state exists for an incoming call when the network has awarded the call to the called user. This state exists for an outgoing call when the network has indicated that the remote user has answered the call.

3.3.1.2.10 Release request

This state exists when the network has received a request from the user to clear the end-to-end connection if present.

3.3.1.2.11 Release indication

This state exists when the network has disconnected the end-to-end connection and has sent an invitation to disconnect the user-network connection.

3.3.2 Additional B-ISDN call/connection states relating to interworking requirements

3.3.2.1 Call/connection at the user side of the interface

3.3.2.1.1 Overlap sending

This state exists for an outgoing call when the user has received acknowledgement of the call establishment request that permits the user to send additional call information to the network in overlap mode.

3.3.2.1.2 Overlap receiving

This state exists for an incoming call when the user has acknowledged the call establishment request from the network and is prepared to receive additional call information in overlap mode.

3.3.2.2 Call/connection states at the network side of the interface

3.3.2.2.1 Overlap sending

This state exists for an outgoing call when the network has acknowledged the call establishment request and is prepared to receive additional call information in overlap mode.

3.3.2.2.2 Overlap receiving

This state exists for an incoming call when the network has received acknowledgement request that permits the network to send additional call information in the overlap mode.

3.4 Message functional definitions

The functional definition of each message is provided in this section.

3.4.1 Messages for B-ISDN call and connection control

3.4.1.1 Call establishment messages

3.4.1.1.1 Alerting

This message is sent by the called user to the network and by the network to the calling user to indicate that the called user alerting has been initiated.

3.4.1.1.2 Call proceeding

This message is sent by the called user to the network or by the network to the calling user to indicate that the requested call establishment has been initiated and no more call establishment information will be accepted.

3.4.1.1.3 Connect

This message is sent by the called user to the network and by the network to the calling user to indicate call acceptance by the called user.

3.4.1.1.4 Connect acknowledge

This message is sent by the network to the called user to indicate the user has been awarded the call. It is also sent by the calling user to the network to allow symmetrical call control procedures.

3.4.1.1.5 Set-up

This message is sent by the calling user to the network and by the network to the called user to initiate B-ISDN call and connection establishment.

3.4.1.2 Call clearing messages

3.4.1.2.1 Release

This message is sent by the user to request the network to clear the end-to-end connection or is sent by the network to indicate that the end-to-end connection is cleared and that the receiving equipment releases the connection identifier and prepare to release its call reference value after sending RELEASE COMPLETE.

3.4.1.2.2 Release complete

This message is sent by the user or the network to indicate that the equipment sending the message has released it's call reference and, if appropriate, the connection identifier. The receiving equipment shall release its call reference.

3.4.1.3 Miscellaneous messages

3.4.1.3.1 Status

This message is sent by the user or the network in response to a STATUS enquiry message or at any time to report certain error conditions.

3.4.1.3.2 Status enquiry

This message is sent by the user or the network at any time to solicit a STATUS message from the peer layer 3 entity. Sending a STATUS message in response to a STATUS ENQUIRY message is mandatory.

3.4.1.3.3 Notify

This message is sent by the user or the network to indicate information pertaining to a call/connection.

3.4.2 Messages used with the global call reference

3.4.2.1 Restart

This message is sent by the user or by the network to request the recipient to restart, that is, to return to an idle condition, the indicated the virtual channel, all virtual channels in the indicated virtual path connection, or all virtual channels controlled by the signaling virtual channel.

3.4.2.1 Restart acknowledge

This message is sent to acknowledge the receipt of a RESTART message and to indicate that the requested restart is complete.

3.5. B-ISDN call/connection procedures

This section describes the general procedures for call or connection control in B-ISDN. They apply only to the point-to-point access configuration. For point-to-point access configuration VCI=5 is used as the signaling channel. The procedures are used to establish B-ISDN connections over a signaling virtual channel that has already been established. Subsequent connections controlled by the same signaling virtual channel are distinguished through different call reference values.

The following general criteria must be satisfied to establish call or connection.

- Basic service support
- VC availability
- Physical and virtual network resources availability to provide QoS requested
- End system resource availability to provide QoS requested
- End-to-end compatibility

The call states referred to in this section cover the states perceived by the network or by the user or states common to both user and network.

3.5.1 Call/connection establishment at the originating interface

Before the invocation of the following procedures, an assured mode signaling AAL connection must be established between the user and the network. All layer 3 messages shall be sent to the signaling AAL using a AAL-DATA request primitive.

Establishment of signaling AAL connections is initiated by transferring an AAL_ESTABLISH-request primitive to the signaling AAL. On receipt of an AAL_ESTABLISH-confirm or AAL_ESTABLISH-indication primitive from the SAAL access signaling procedures may begin.

3.5.1.1 Call/connection request

The calling party initiates the call establishment by transferring a SETUP message on the assigned signaling virtual channel across the interface and starts timer T303. Following this action, the call is considered by the calling party to be in the Call Initiated state. The message shall always contain a call reference and a dummy call reference shall not be used as a call reference. The ATM traffic descriptor, Broadband bearer capability and QoS parameter information elements are mandatory in the SETUP message. The message may also contain all or part of the call information necessary for call establishment depending on whether *en bloc* or overlap procedures are being used.

If en bloc sending is used, the SETUP message contains all the information required by the network to process the call.

If the user before the first expiry of timer T303 receives no response to the SETUP message, the message shall be retransmitted and timer T303 restarted. If the user has not received any response to the SETUP message after the final expiry of timer T303, the user shall clear the call internally.

3.5.1.2 Connection identifier (VPCI/VCI) allocation/selection – Origination

Two cases of signaling exist

1. Associated signaling

The layer 3 signaling entity exclusively controls the VCs in the VPC that carries its signaling VC.

2. Non-associated signaling

The layer 3 signaling entity controls the VCs in the VPC that carries its signaling VC and may control VCs in other VPCs.

The network and user shall support the non-associated signaling procedures and may as an option support the associated signaling procedures. When the network receives a connection identifier information element with the VP-associated signaling field coded with a value not supported by the network, the call shall be rejected with cause "VPCI/VCI assignment failure".

3.5.1.2.1 Associated signaling

In this signaling procedure, the user requests a VC in the VPC carrying the signaling VC. The VPC carrying the signaling VC is implicitly indicated.

In the connection identifier information element, the VP-associated signaling field is coded as "VP-associated signaling" and one of the following values is indicated in the preferred/exclusive field:

1. Exclusive VPCI; any VCI
2. Exclusive VPCI; exclusive VCI

In case (1), the network selects any available VCI within the VPC carrying the VC.

In case (2), if the indicated VCI within the VPC carrying the signaling VC is available, the network selects it for the call.

The selected VCI is indicated in the connection identifier information element in the first message returned by the network in response to the SETUP message. The VP-associated signaling field is coded as "VP-associated signaling". The preferred/exclusive field is coded as "exclusive VPCI; exclusive VCI".

In case (1), if no VCI is available, a RELEASE COMPLETE message with cause "no VPCI/VCI available" is sent by the network.

In case (2), if the indicated VCI is not available, a RELEASE COMPLETE message with cause "requested VPCI/VCI not available" is sent by the network.

3.5.1.2.2 Non-associated signaling

In the user's request for a VC in the SETUP message, the user shall indicate one of the following:

1. Exclusive VPCI; any VCI
2. Exclusive VPCI; exclusive VCI
3. No indication is included. In this case, no connection identifier information element is included in the SETUP message.

In cases (1) and (2), the VP-associated signaling field is coded as "explicit indication of VPCI' in the connection identifier information element. If the indicated VPCI is available for these cases, the network selects it for the call.

In case (3), the network selects any available VPCI and VCI.

The selected VPCI/VCI is indicated in the connection identifier information element in the first message returned by the network in response to the SETUP message. The VP-associated signaling field is coded as "explicit indication of VPCI". The preferred/exclusive field is coded as "exclusive VPCI; exclusive VCI".

In case (1) and (2), is the specified VPCI is not available, a RELEASE COMPLETE message with cause " requested VPCI/VCI not available" is sent by the network.

In case (1), if no VCI is available, a RELEASE COMPLETE message with cause "no VPCI/VCI available" is sent by the network. In addition, if the VPCI values in the first response message are not the VPCI value indicated by the user, the network sends a RELEASE message with cause "VPCI/VCI assignment failure".

In case (2), if the VCI in the indicated VPCI is not available, a RELEASE COMPLETE message with cause "requested VPCI/VCI not available" is sent by the network. In addition, if the VPCI and VCI values in the first response message are not the VPCI and VCI values indicated by the user, the network sends a RELEASE message with cause "VPCI/VCI assignment failure".

In case (3), if the network is not able to allocate a VCI in any VPCI, a RELEASE COMPLETE message with cause "no VPCI/VCI available" is sent by the network.

3.5.1.2.3 Use of VPCI

The connection identifier element is used in signaling messages to identify the corresponding user information flow. It contains the Virtual Path Connection identifier (VPCI) and the Virtual Channel Identifier (VCI). The VPCI is used instead of the Virtual Path Identifier (VPI) as the virtual path cross connects may be used in the access and multiple interfaces could be controlled by the signaling virtual channel. VPCIs have significance only with regard to a given signaling virtual channel.

If the signaling virtual channel controls only a single interface at the user side, the VPI and VPCI have the same numerical value at the user side and this is shown in the figure below.

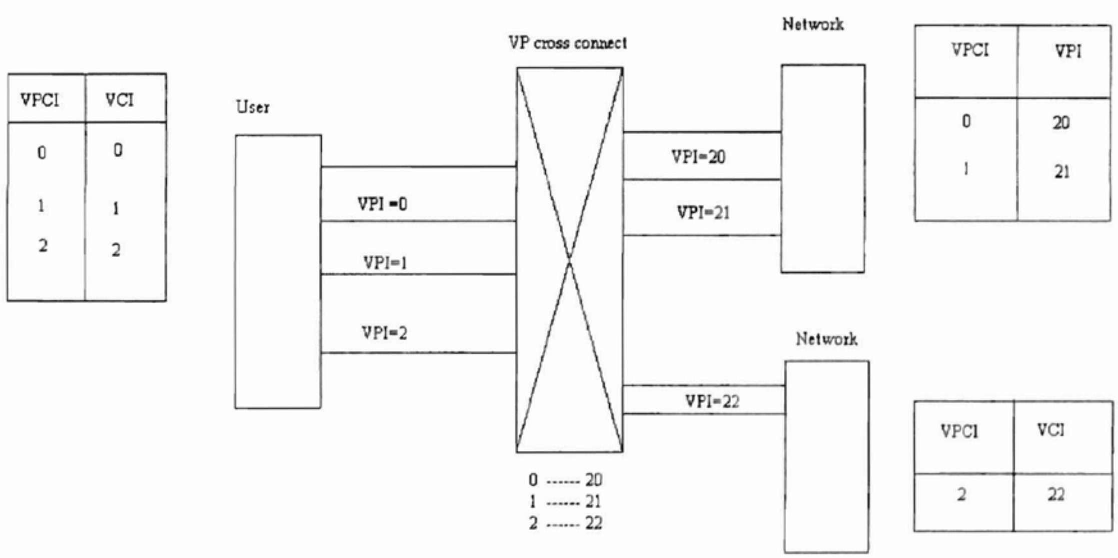

Figure 3-1. Single Interface controlled by SVC

If the signaling channel controls multiple interfaces at the user side, the VPCI corresponds to both the interface and the VPI on the interface.

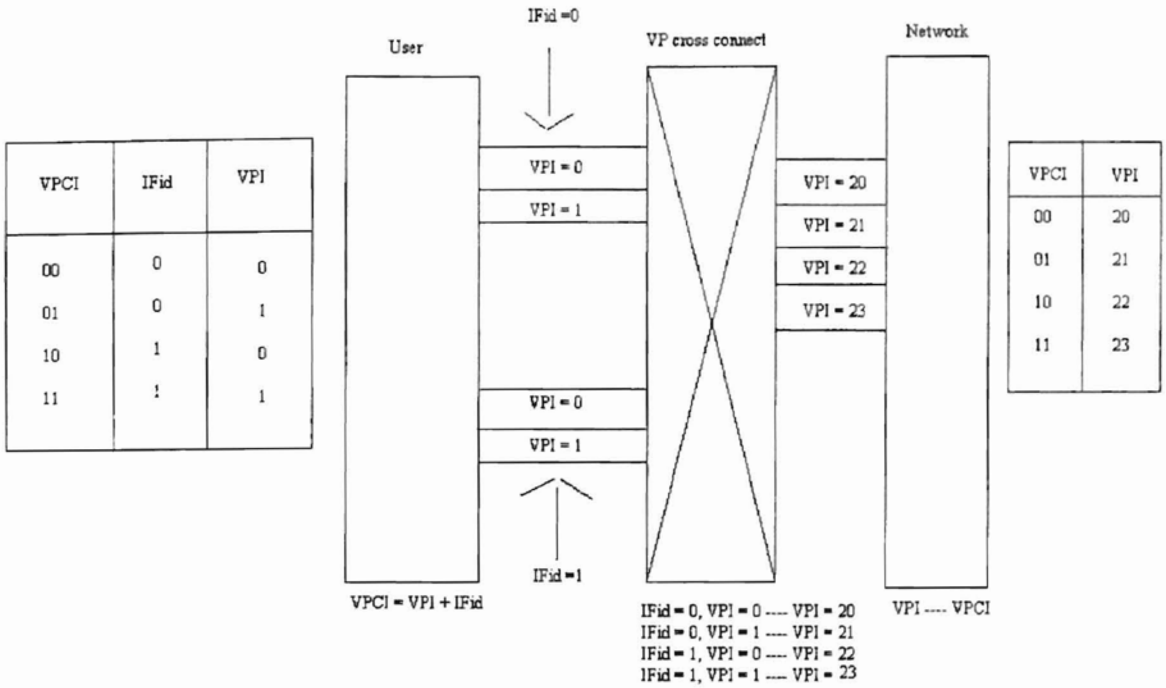

Figure 3-2. Multiple interfaces controlled by SVC

3.5.1.2.4 VCI Range

The range of valid VCI values is

- 0 to 31- This range is not used for on-demand user plane connections
- 32 to 65535 This range is used as identifier of the virtual channel

3.5.1.3 QoS and traffic parameters selection procedures

The user indicates the QoS class in the Quality of Service parameter information element. If the network is able to provide the requested QoS class, the network progresses the call to the called user. If the network is not able to provide the requested QoS class, the network rejects the call, returning a RELEASE COMPLETE message with cause "Quality of Service unavailable".

The user indicates the requested peak cell rate in the ATM user cell rate information element. If the network is able to provide the requested peak cell rate, the network progresses the call to the called user. If the network is not able to provide the requested peak cell rate, the network rejects the call, returning a RELEASE COMPLETE message with cause "user cell rate unavailable".

3.5.1.4 Invalid call/connection control information

If after the receiving the SETUP message, the network determines that the call information received from the user is invalid, then the network shall initiate call clearing with a cause which can be either "unassigned number", or, "no route to destination", or, "number changed", or, "invalid number format".

3.5.1.5 Call/connection proceeding

If the network can determine that access to the requested service is authorized and available, the network sends a CALL PROCEEDING message to the user to acknowledge the SETUP message and to indicate that the call is being processed and enter the Outgoing Call Proceeding state. When the user receives the CALL

PROCEEDING message, the user stops timer T303, starts timer T310 and enters the Outgoing Call Proceeding state.

If the network determines that, a requested service is not authorized or not available, the network initiates call clearing with a cause that can be either "bearer capability not authorized", or, "bearer capability not presently available", or, "service or option not available", or, "bearer service not implemented".

If the user has received a CALL PROCEEDING message, but does not receive an ALERTING, CONNECT, or RELEASE message before the expiration of timer T310, then the user initiates clearing procedures towards the network with cause "recovery on timer expiry".

3.5.1.6 Call/connection confirmation indication

After receiving an indication that user alerting has been initiated at the called address, the network sends an ALERTING message across the user-network interface of the calling address and enter the Call Delivered state. When the user receives the ALERTING message, the user may begin an internally generated alerting indication and stop timer T310 and enter the Call Delivered state.

3.5.1.7 Call/connection acceptance

After receiving an indication that the call has been accepted, the network sends a CONNECT message across the user-network interface to the calling user and enter the active state. This message indicates to the calling user that a connection has been established through the network.

On reception of the CONNECT message, the calling user stops timer T310 (if running), stops any user generated alerting indication, attaches to the user plane virtual channel if not already done and sends a CONNECT ACKNOWLEDGE messages and enters the Active state.

At this point, an end-to-end connection is established.

3.5.1.8 Call/connection rejection

After receiving an indication that the network or the called user is unable to accept the call, the network initiates clearing at the originating user-network interface using the cause provided by the terminating network or called user.

3.5.2 Call/connection establishment at the destination interface

3.5.2.1 Incoming call/connection request

The network indicates the arrival of a call at the user-network interface by transferring a SETUP message across the interface. This message always contains a call reference. The network starts timer T303 and enters the Call Present state. The network sends this message only if resources for the call are available, else the call is cleared towards the user with the cause "resources unavailable, unspecified".

If en bloc receiving is used, the SETUP message contains all the information required by the called user to process the call. In this case, the SETUP message may contain Broadband sending complete information element.

On reception of a SETUP message, the user enters the Call Present state.

If the SETUP message includes Broadband sending complete information element, en bloc receiving procedures will be followed.

If timer T301 expires for the first time, the network retransmits the SETUP message and restart timer T303. If timer T303 expires for the second time, the network follows the procedure given in 4.2.5.4

3.5.2.2 Address and compatibility check

3.5.2.2.1 Address check

If an address check is not possible because of the address information not being included, the user performs compatibility checking.

3.5.2.2.2 Compatibility check

3.5.2.2.2.1 Principles

The user performs compatibility checking based on the compatibility information received in the SETUP message. In B-ISDN, there are two categories of compatibility information:

- Broadband category 1 compatibility information is provided for both the network and the user to determine the attributes of the ATM connection. The called user always checks this information. If the compatibility check fails, the user is incompatible. The broadband category 1 compatibility information is:
    - Broadband bearer capability information

- End-to-end transit delay information
- ATM traffic descriptor
- Quality of Service parameter
- QAM traffic descriptor

- Broadband category 2 compatibility information is provided for the called user. The called user always checks this information. If the compatibility check fails, the user is incompatible. The broadband category 2 compatibility information is:

- ATM adaptation layer parameter information
- Optional broadband low layer information
- Optional broadband high layer information

3.5.2.2.2.2 Point-to-point call/connection offering

A user receiving a SETUP message performs compatibility checking before responding to that SETUP message. An incompatible user responds with a RELEASE COMPLETE message with cause "incompatible destination" and enters the Null state.

3.5.2.3 Connection identifier (VPCI/VCI) allocation/selection-Destination

Two cases of signaling exist

1. Associated signaling

   The layer 3 signaling entity exclusively controls the VCs in the VPC that carries its signaling VC.

2. Non-associated signaling

The layer 3 signaling entity controls the VCs in the VPC that carries its signaling VC and may control VCs in other VPCs.

The user shall support the non-associated signaling procedures and may as an option support the associated signaling procedures. When the network receives a connection identifier information element with the VP-associated signaling field coded with a value not supported by the network, the call shall be rejected with cause "VPCI/VCI assignment failure".

3.5.2.3.1 Associated signaling

In this signaling procedure, the network indicates a VC in the VPC carrying the signaling VC. The VPC carrying the signaling VC is implicitly indicated.

In the connection identifier information element, the VP-associated signaling field is coded as "VP-associated signaling" and one of the following values is indicated in the preferred/exclusive field:

1. Exclusive VPCI; any VCI
2. Exclusive VPCI; exclusive VCI.

In case (1), the user selects any available VCI within the VPC carrying the VC. The selected VCI is indicated in the connection identifier information element in the first message returned by the user in response to the SETUP message. The VP-associated signaling field is coded as "VP-associated signaling". The preferred/exclusive field is coded as "exclusive VPCI; exclusive VCI".

In case (2), if the indicated VCI within the VPC carrying the signaling VC is available, the user selects it for the call. If the Connection identifier information element is not present in the first response message, the connection identifier in the SETUP message shall be assumed.

In case (1), if no VCI is available, a RELEASE COMPLETE message with cause "no VPCI/VCI available" is sent by the user.

In case (2), if the indicated VCI is not available, a RELEASE COMPLETE message with cause "requested VPCI/VCI not available" is sent by the user.

3.5.2.3.2 Non-associated signaling

For non-associated signaling, the network indicates one of the following in the SETUP message:

1. Exclusive VPCI; any VCI
2. Exclusive VPCI; exclusive VCI
3. No indication is included.

In cases (1) and (2), if the indicated VPCI is available, the user selects it for the call. In case (1), the user selects any available VCI in the VPCI. In case (2), if the indicated VCI is available within the VPCI, the user selects it for the call. In case (3), the user selects any available VPCI and VCI.

In cases (1) and (3), the selected VPCI/VCI value is indicated in the Connection identifier information element returned by the user in response to the SETUP message. The VP-associated signaling field is coded as "explicit indication of VPCI". The preferred/exclusive field is coded as "exclusive VPCI; exclusive VCI".

In case (2), if the Connection identifier information element is not present in the first response message, the connection identifier in the SETUP message shall be assumed.

In case (1) and (2), if the specified VPCI is not available, a RELEASE COMPLETE message with cause "requested VPCI/VCI not available" is sent by the user.

In case (1), if no VCI is available, a RELEASE COMPLETE message with cause "no VPCI/VCI available" is sent by the user.

In case (2), if the VCI in the indicated VPCI is not available, a RELEASE COMPLETE message with cause "requested VPCI/VCI not available" is sent by the user.

In case (3), if the user is not able to allocate a VCI in any VPCI, a RELEASE COMPLETE message with cause "no VPCI/VCI available" is sent by the user.

In case (1), if the VPCI value in the first response message is not the VPCI value indicated by the network, a RELEASE COMPLETE message with cause "VPCI/VCI assignment failure " is sent to the user.

In case (2), if the VPCI and VCI values in the first response message are not the VPCI and VCI values indicated by the network. A RELEASE message with cause "VPCI/VCI assignment failure" is sent to the user.

3.5.2.4 QoS and traffic parameter selection procedures

The network indicates the QoS class in the Quality of Service parameter information element. If the user is not able to provide the requested QoS class, the user rejects the call, returning a RELEASE COMPLETE message with cause "Quality of Service unavailable".

The cumulative end-to-end transit delay is indicated in the end-to-end transit delay information element. If the user is not able to accept this delay, the user rejects the call, returning a RELEASE COMPLETE message with cause "Quality of Service unavailable".

The network indicates the peak cell rate in the ATM traffic descriptor information element. If the user is not able to provide the indicated peak cell rate, the user rejects the call, returning a RELEASE COMPLETE message with cause "resources unavailable, unspecified".

3.5.2.5 Call/connection confirmation

3.5.2.5.1 Response to en bloc SETUP or completion of overlap receiving

When the user determines that sufficient call set-up information has been received and compatibility requirements have been satisfied, the user responds with either a CALL PROCEEDING, ALERTING or CONNECT message and enters the Incoming Call Proceeding, Call Received or Connect Request state respectively. An incompatible user shall respond by sending a RELEASE COMPLETE message and enters the Null state.

A busy user that satisfies the compatibility requirements indicated in the SETUP message responds with a RELEASE COMPLETE message with cause "user busy". If the user wishes to refuse the call, a RELEASE COMPLETE message is sent with cause "ca;; rejected' and the user returns to Null state.

3.5.2.5.2 Receipt of CALL PROCEEDING and ALERTING

On reception of the CALL PROCEEDING message from a user, the network stops timer T303, starts timer T310 and enters the Incoming Call Proceeding state.

On reception of the ALERTING message from a user, the network stops timers T303 or T310, starts timer T301, enters the Call Received state and sends a corresponding ALERTING message to the calling user.

3.5.2.5.3 Called user clearing during incoming call establishment

If a RELEASE COMPLETE or RELEASE message is received before a CONNECT message has been received, the network stops timer T303, timer T310 or timer T301, clears the call to the called user and clears the call to the calling user with the cause received in the RELEASE COMPLETE or RELEASE message.

3.5.2.5.4 Call failure

If the network does not receive any response to the retransmitted SETUP message before the expiration of timer T303, the network enters the Null state and initiates clearing procedures towards the calling user with cause "no user responding".

If the network has received a CALL PROCEEDING message, but does not receive an ALERTING, CONNECT or RELEASE message prior to the expiration of timer T310, the network initiates clearing procedures towards the calling user with cause "no user responding" and initiates clearing procedures towards the called user with cause 'recovery on timer expiry".

If the network has received an ALERTING message, but does not receive a CONNECT or RELEASE message prior to the expiration of timer T301, then the network initiates clearing procedures towards the calling user with cause "no answer from user" and initiates clearing procedures towards the called user with cause "recovery on timer expiry".

3.5.2.6 Call/connection acceptance

A user indicates the acceptance of an incoming call by sending a CONNECT message to the network. It then starts timer T313 and enters the Connect Request state.

3.5.2.7 Active indication

On reception of the CONNECT message, the network stops timers T301,T303 and T310, enters the Connect Request State, sends a CONNECT ACKNOWLEDGE message to the user, initiates procedures to send a CONNECT message towards the calling user and enters the Active state.

The CONNECT ACKNOWLEDGE message indicates completion of the connection establishment procedures. There is no guarantee of end-to-end connection until a CONNECT message is received at the calling user. On reception of the CONNECT ACKNOWLEDGE message, the called user stops timer T313, attaches to the user virtual plane and enters the Active state.

3.5.3 Call/connection clearing

The following terms are used in this section to describe clearing procedures:

- A VC channel is connected when the VC is part of a B-ISDN virtual connection

- A VC is disconnected when the VC is no longer part of a B-ISDN virtual connection, but is not yet available for use in a new virtual connection
- A VC is released when the VC is not part of a B-ISDN virtual connection and is available for use in a new virtual connection.

3.5.3.1 Exception conditions

Under normal conditions, call clearing is usually initiated when the user or network sends a RELEASE message. The only exception is in response to a SETUP message, the user or network rejects a call by responding with a RELEASE COMPLETE message provided no other response has previously been sent releasing the call reference.

3.5.3.2 Clearing initiated by the user

The user initiates clearing by sending a RELEASE message, starts timer T308, disconnects the virtual channel and enters the Release Request state.

The network enters the Release Request state upon receipt of a RELEASE message. This then prompts the network to disconnect the virtual channel, and to initiate procedures for clearing the network connection to the remote user. Once the virtual channel used for the call has been disconnected, the network sends a RELEASE COMPLETE message to the user, releases its call reference and virtual channel and enters the Null state.

On reception of the RELEASE COMPLETE message, the user stops timer T308, releases the virtual channel and call reference and returns to the Null state.

If timer T308 expires for the first time, the user retransmits a RELEASE message to the network, resets timer T308 and remains in the Release Request state. Also, the user may indicate a second Cause information element with cause "recover on timer expiry'. If no RELEASE COMPLETE message is received from the network before timer T308 expires a second time, the user places the virtual channel in a maintenance condition, releases call reference and returns to the Null state.

When user initiates normal call/connection clearing, cause "normal clearing" is used in the first clearing message.

3.5.3.3 Clearing initiated by the network

The network initiates clearing by sending a RELEASE message, starts timer T308, disconnects the virtual channel and enters the Release Indication state.

The user enters the Release Indication state upon receipt of a RELEASE message. Once the virtual channel used for the call has been disconnected, the user sends a RELEASE COMPLETE message to the network, releases its call reference and virtual channel and enters the Null state.

On reception of the RELEASE COMPLETE message, the network stops timer T308, releases the virtual channel and call reference and returns to the Null state.

If timer T308 expires for the first time, the network retransmits a RELEASE message to the user, starts timer T308 and remains in the Release Indication state. Also, the network may indicate a second Cause information element with cause "recover on timer expiry'. If no RELEASE COMPLETE message is received from the user before timer T308 expires a second time, the network places the virtual channel in a maintenance condition, releases call reference and returns to the Null state.

3.5.4 Restart procedure

The restart procedure is used to return a virtual channel, all virtual channels in a virtual path or all virtual paths controlled by the signaling virtual channel to the idle condition. It is usually invoked when the other side of the interface does not respond to other call control messages or a failure has occurred.

3.5.4.1 Sending RESTART

3.5.4.1.1 Normal procedure

A RESTART message is sent by the network or user equipment to return virtual channels to the idle condition. The Restart indicator information element shall be present in the RESTART message to indicate whether an "indicated virtual channel", "all user plane virtual channels in the indicated VPC controlled via signaling virtual channel, in which the RESTART message is sent", or "all virtual channels controlled by layer 3 entity" are to be restarted. The Connection identifier information element shall be present in the first two cases to indicate which virtual channel or virtual path is to be returned to the idle condition and the element shall not be present in the last case.

After transmitting the RESTART message, the sender enters the Restart Request state. Starts timer T316, waits for a RESTART ACKNOWLEDGE message. Receipt of a RESTORE ACKNOWLEDGE message stops timer T316 and indicates that the virtual channels and associated resources can be freed for re-use. The Null state is entered after the virtual channel and call reference value are released.

Calls associated with restart user plane virtual channels are cleared towards the remote parties using cause "temporary failure".

3.5.4.1.2 Exceptional procedures

If a RESTART ACKNOWLEDGE message is not received before the expiry of timer T316, one or more RESTART messages may be sent until a RESTART ACKNOWLEDGE message is returned. The number of consecutive unsuccessful restart attempts is two. After this, the originator of RESTART message makes no further attempts and enters the Null state. An indication is provided to the appropriate maintenance entity. The virtual channel is considered to be in an out-of-service condition until maintenance action has been taken.

3.5.4.2 Receipt of RESTART

3.5.4.2.1 Normal procedure

On receiving a RESTART message, the recipient enters the Restart state associated to the global call reference, starts timer T317, initiates the appropriate internal actions to return the specified virtual channels to the idle condition and release all call references associated with the specified virtual channels. After completion of internal clearing, timer T317 is stopped and a RESTART ACKNOWLEDGE message is transmitted to the originator and the Null state is entered. The RESTART

ACKNOWLEDGE message indicates a Restart indicator information element containing the same information as received in the related RESTART message.

Calls associated with restart user plane virtual channels shall be cleared towards the remote parties using cause 'temporary failure".

If the Restart indicator information element is coded as "all virtual channels controlled by layer 3 entity which sends the RESTART message", then all calls on all interfaces associated with the signaling virtual channel shall be cleared.

If semi-permanent connections established by management procedures are implicitly specified, no action shall be taken on these virtual channels, but a RESTART ACKNOWLEDGE message is returned containing appropriate indications.

If semi-permanent connections established by management procedures or reserved VPCI/VCIs are explicitly specified, no action shall be taken on these virtual channels and a STATUS message may be returned, as an option, with cause "identified channel does not exist".

The following entities are released because of the Restart procedures:

- Virtual channels established by Q.2931 Recommendation's procedures
- All resources associated with the released virtual channel

The following entities are not released because of the Restart procedures:

- Permanent connections established by the network
- Management system reserved virtual channels.

3.5.4.2.2 Exceptional procedures

If timer T317 expires before completion of internal clearing, an indication is sent to the maintenance entity and the Null state is entered.

If the Restart indicator information element is coded as, "all virtual channels controlled by layer 3 entity which sends the RESTART message" and a Connection identifier element is included, the Connection identifier information element is treated as described in 3.5.5.8.3

If the Restart indicator information element is coded as, "indicated virtual channel" and the Connection identifier element is not included, then the procedures described in 3.5.5.7.1 are followed

If the Restart indicator information element is coded as "indicated virtual channel" or "all user plane virtual channels in the indicated VPC controlled via signaling virtual channel, in which the RESTART message is sent" and the Connection identifier information element contains an unrecognized VPCI, the procedures in 3.5.5.7.2 are followed.

If the RESTART message is received in the Restart state, the procedures in 3.5.5.4 are followed.

3.5.5 Handling of error conditions

Detailed error handling procedures are implementation dependent and may vary from network to network. Nevertheless, capabilities facilitating the orderly treatment of error conditions are provided in this subsection.

3.5.5.1 Protocol discrimination error

When a message is received with a protocol discriminator coded other than Q.2931 user-network call control message, that message will be ignored.

3.5.5.2 Message too short

When a message received is too short to contain a complete Message length information element, that message will be ignored.

3.5.5.3 Call reference error

3.5.5.3.1 Call reference procedural errors

1. Whenever any message except SETUP, RELEASE COMPLETE, STATUS ENQUIRY or STATUS is received specifying a call reference that is not recognized as relating to an active call or to a call in progress, the receiver initiates call clearing by sending a RELEASE COMPLETE message with cause "invalid call reference value", specifying the call reference in the received message and remains in the Null state.
2. When a RELEASE COMPLETE message is received specifying a call reference that is not recognized as relating to an active call or to a call in progress, no action is taken
3. When a SETUP message is received specifying a call reference that is not recognized as relating to an active call or to a call in progress and with a call reference flag incorrectly set to "1", this message is ignored.

4. When a SETUP message is received specifying a call reference that is recognized as relating to an active call or to a call in progress, this message is ignored

5. When any message except RESTART, RESTART ACKNOWLEDGE or STATUS message is received using the global call reference, no action is taken on this message and a STATUS message using the global call reference with a call state indicating the current state associated with the global call reference and cause "invalid call reference" is returned.

6. When a STATUS message is received specifying a call reference that is not recognized as relating to an active call or to a call in progress, the procedures of 4.5.12 applies.

7. When a STATUS INQUIRY message is received specifying a call reference that is not recognized as relating to an active call or to a call in progress, the procedures of 4.5.11 applies.

8. When a RESTART message is received specifying the global call reference with a call reference flag incorrectly set to "1" or a RESTART ACKNOWLEDGE message is received specifying the global call reference with a call reference flag incorrectly set to "0", no action is taken on this message and a STATUS message with a call state indicating the current state associated with the global call reference and cause "invalid call reference" is returned.

3.5.5.4 Message type or message sequence errors

The error procedures in this subsection apply only if the flag in the message compatibility instruction indicator is set to "message instruction field not significant".

Whenever an unexpected message, except RELEASE, RELEASE COMPLETE or an unrecognized message is received in any state other than the Null state, no state change occurs and a STATUS message is returned with either a cause "message type non-existent or not implemented" or "message not compatible with the call state".

3.5.5.5 Message length error

If the message length indicated in the Message length information element is inconsistent with the length of the message actually received, the message is handled normally as far as possible and, if necessary, the error handling procedures of 4.5.6 are followed.

3.5.5.6 General information element errors

The general information element error procedures may also apply to information in code sets other than 0. Then, the diagnostics in the Cause information element may indicate information elements other than those in code set 0.

3.5.5.6.1 Information element sequence

Variable length information elements may appear in any order within a message. If more than one information element of the same type is included in a message, and the repeated information elements do not immediately follow the preceding occurrence of this information type, the receiving entity ignores subsequent information element of this type.

3.5.5.6.2 Duplicated information elements

If an information element is repeated in a message in which repetition of the information element is not permitted, only the contents of the information element appearing first are handled and all subsequent repetitions of the information element is ignored. When repetition of information elements is permitted and when the limit on repetition of information elements is exceeded, the contents of information element appearing first up to the limit of repetitions are handled and all subsequent repetitions of the information element are ignored.

3.5.5.6.3 Coding standard error

If the user or network receives an information element with the coding standard field indicating a coding standard that is not supported by the receiver, this information element is treated as an information element with a content error

3.5.5.7 Mandatory information element error

3.5.5.7.1 Mandatory information element missing

When a message other than SETUP, RELEASE or RELEASE COMPLETE is received that has one or more mandatory information elements missing, no action is taken on the message and no state change occurs. A STATUS message is returned with cause, "mandatory information element is missing".

"normal, unspecified", was received, with the exception that the RELEASE COMPLETE message sent on the local interface contains cause, "invalid information element contents".

When a RELEASE COMPLETE message is received with invalid content of the Cause information element, it will be assumed that a RELEASE COMPLETE message was received with cause, "normal, unspecified".

Information elements with a length exceeding the maximum length are treated as information elements with content error.

3.5.5.8 Non-mandatory information element error

The error procedures in this subsection apply only if the flag in the instruction field is set to "IE instruction field not significant".

3.5.5.8.1 Unrecognized information element

When a message is received that has one or more unrecognized information elements, then the receiving entity proceeds as follows.

Action is taken on the message and those information elements that are recognized and have valid content. When the received message is other than RELEASE or RELEASE COMPLET, a STATUS message is returned containing one Cause information element. This indicates the call state of the receiver after taking action on the message. The Cause information element contains cause "information element non-existent or not implemented", and the diagnostic field contains the information element identifier for each information element that was unrecognized.

If a clearing message contains one or more unrecognized information elements, the error is reported to the local user in the following manner:

1. When a RELEASE message is received that has one or more unrecognized information elements, RELEASE COMPLETE message with cause, "information element non-existent or not implemented" is returned.
2. When a RELEASE COMPLETE message is received that has one or more unrecognized information elements, no action is taken on the unrecognized information.

3.5.5.8.2 Non-mandatory information element content error

When a message is received that has one or more non-mandatory information elements with invalid content, action is taken on the message and those information elements that are recognized and have valid content. A STATUS message is returned containing one Cause information element. This indicates the call state of the receiver after taking action on the message. The Cause information element contains cause "invalid information element contents", and the diagnostic field contains the information element identifier for each information element that was invalid contents. Information elements with a length exceeding the maximum length are treated as information elements with content error.

3.5.5.8.3 Unexpected recognized information element

When a message is received with a recognized information element that is not defined to be contained in that message, the receiving entity treats the information element as an unrecognized information element and follows the procedures defined in 4.5.8.1

3.5.5.9 Signaling AAL connection reset

Whenever a Q.2931 entity is informed of a spontaneous Signaling ALL reset by means of the AAL-ESTABLISH-indication primitive, the following procedures apply:

1. For call in the clearing phase, no action is taken.
2. For calls in the establishment phase, the calls are maintained.
3. Calls in the Active state are maintained and the entity invokes the status enquiry procedures.

3.5.5.10 Signaling AAL connection release

Whenever a Q.2931 entity is informed by its Signaling ALL connection release by means of the AAL-RELEASE-indication primitive, the following procedures apply:

1. Any calls not in the active state are cleared locally.
2. If there is at least one call in the Active state controlled by the released Signaling AAL connection, timer T309is restarted, if its not running already.

The Q.2931 entity requests Signaling AAL re-establishment by sending an AAL-ESTABLISH-request primitive. Then, timer T309 is stopped and status enquiry procedure is performed to verify the call state of the peer entity per each call/connection.

3.5.5.11 Status enquiry procedure

To check the correctness of a call state at a peer entity, a STATUS ENQUIRY message is sent requesting the call state. In addition, it may be also sent whenever indication is received from the Signaling AAL that a disruption has occurred at the data link layer.

After sending the STATUS ENQURIY message, timer T322 is started in anticipation of receiving a STATUS message. While timer T322 is running, only one outstanding request for call state information shall exist. If a clearing message is received before timer T322 expires, timer T322 is stopped and call clearing is continued.

After reception of a STATUS ENQUIRY message, the receiver responds with a STATUS message, reporting the current call state and cause, "response to STATUS ENQUIRY".

The sending or receipt of the STATUS message will not directly affect the call state of either the sender or receiver. The side having received the STATUS message inspects the Cause information element. If the cause is "response to STATUS ENQUIRY", timer T322 is stopped and the appropriate action taken, based on the information in that STATUS message, relative to the current state of the receiver.

If timer T322 expires and no STATUS message were received, the STATUS ENQUIRY message may be retransmitted one or more times until a response is received. If following the maximum number of retransmissions of the STATUS ENQUIRY message no STATUS message is received, the call is cleared to the local interface with cause "temporary failure".

3.5.5.12 Receiving a STATUS message

On receipt of a STATUS message reporting an incompatible state, the receiving entity shall:

1. Clear the call by sending the appropriate clearing message with cause "message not compatible with call state"; or
2. Take other actions that attempt to recover from a mismatch.

Except for the following rules, the determination of which states are incompatible is left as an implementation option:

1. If a STATUS message indicating any call except the Null state is received in the Null state, the receiving entity sends a RELEASE COMPLETE message with cause, "message not compatible with call state" and remains in Null state.
2. If a STATUS message indicating any call except the Null state is received in the Release Request or Release Indication state, no action is taken.
3. If a STATUS message indicating the Null state is received in any state except the Null state, the receiver releases all resources and moves into Null state.

A status message may be received indicating a compatible call state but containing one of the following causes: mandatory information element missing, or message type non-existing or not implemented, or information element non-existent or not implemented, or invalid information element contents, or message not compatible with call state. In these cases, the actions to be taken are an implementation option.

On receipt of a STATUS message specifying the global call reference and reporting an incompatible state in the Restart Request or Restart start, the receiving Q.2931 entity informs the layer management and takes no further action on this message.

3.5.6 Error procedures with explicit action indication

The procedures of this subsection can be used only if the flag of the message compatibility instruction indicator or information element instruction field is set to "follow explicit instruction".

3.5.6.1 Unexpected or unrecognized message type

If an unexpected or unrecognized message type is received in any state other than the Null state, the following procedures are applied.

If the action indicator bits of the instruction field of a Message type information element are set to "clear all" in any state other than the Release Request and Release Indication state, the call is cleared with cause "message type non-existent or not implemented", or "message not compatible with call state". When in the Release Request or in the Release Indication state, the receiver takes no action and remains in the same state.

If the action indicator bits of the instruction field of a Message type information element are set to "discard and ignore", the message is ignored.

If the action indicator bits of the instruction field of a Message type information element are set to "discard and status", no action is taken on the message but a STATUS message is sent with a CAUSE information element "message type non-existent or not implemented" or "message not compatible with call state".

If the action indicator bits of the instruction field of a Message type information element are set to an undefined value, the receiver handles the message as if the message action indicator bits had been set to "discard and report status".

3.5.6.2 Information element errors

When a message other than a RELEASE or RELEASE COMPLETE message is received that has one or more unexpected information elements, unrecognized elements or information elements with unrecognized contents, the receiving entity examines the information element action indicator and follows the following procedures.

When a RELEASE message is received with one or more information elements in error, a RELEASE COMPLETE message with cause "information element non-existent or not implemented" or "invalid information element contents" is returned.

When a RELEASE COMPLETE message is received with one or more information elements in error, no action is taken on the information elements in error.

If more than one information element is received in error, only one response is given. The response is according to the handling of the action indicator filed according to the following order of priority: "clear call", "discard message and report status", "discard message and ignore", "discard information element, proceed and report status", "discard information element and proceed".

1. Action indicator field = clear call

Here, the call is cleared with the Cause information element "information element non-existent or not implemented" or "invalid information element contents".

2. Action indicator field = discard message and report states

The message is ignored and a STATUS message is sent with cause "information element non-existent or not implemented" or "invalid information element contents".

3. Action indicator field = discard message

The message is ignored.

4. Action indicator field = discard information element, proceed and report status

The information element is discarded, the handling of the message proceeds and STATUS message is returned indicating the call state of the receiver after taking action on the message and contains clause "information element non-existent or not implemented" or "invalid information element contents".

5. Action indicator field = discard information element and proceed

The information element is ignored and the message is processed as if the information element was not received. No STATUS message is sent.

6. Action indicator field = undefined value

The receiver handles the information element as if the action indicator filed had been set to "discard information element, proceed and report status".

3.5.7 Notification procedures

The delivery of bearer related notification uses an active call reference of the call or connection the notification is associated with. Here, a call reference is active from the initiation of the call establishment to the initiation of call clearing.

If the delivery of the notification coincides with call/connection establishment or clearing procedures, the notification information can be carried in the associated call control messages. In all other cases, the notification information is delivered in a NOTIFY message. Also, a NOTIFY message may be sent or received by the user or by the network only after the first response to a SETUP message has been sent or received and before clearing of the call reference is initiated.

If the network receives a notification, the network optionally ensures that the contents of the notification are a valid coding and forwards the notification to the other user involved in the call.

No call state change occurs at either side of the interface following the sending or receipt of a NOTIFY message.

CHAPTER IV

LITERATURE REVIEW

4.1 MPLS

Multiprotocol Label Switching (MPLS) is a technique that brings most of the qualities and attributes of switched networks to Internet Protocol (IP) networks. MPLS is a versatile solution that addresses the problems faced by the IP networks of today-speed, scalability, Quality of Service (QoS) management and traffic engineering. MPLS can exist over existing asynchronous transfer mode (ATM) and frame-relay networks. It is an elegant solution to meet the bandwidth-management and service requirements of next generation IP based backbone networks.

4.2. Components of MPLS

MPLS performs the following functions:

- It specifies mechanisms to manage traffic flows of various granularities.
- It remains independent of the layer-2 and layer-3 protocols.
- It provides a means of mapping IP addresses to simple, fixed-length labels.
- It interfaces to existing routing protocols like open shortest path first and resource reservation protocol
- It supports ATM, IP, and frame relay protocols.

Data transmission in MPLS occurs on label switched paths (LSPs). LSPs can be established either before data transmission (control-driven) or upon detection of a certain flow of data (data-driven). LSPs are a sequence of labels at each node along the path from the source to the destination. The labels are distributed using label distribution protocol (LDP) or resource reservation protocol (RSVP) or piggybacked on routing protocols like border gateway protocol (BGP) and OSPF. The labels are encapsulated in each data packet and carried from source to destination. As the fixed-length labels are inserted at the very beginning of the packet or cell and can be used by hardware to switch packets quickly between links, high-speed switching of data is possible.

4.2.1 Label Switched Routers and Label Edge Routers

Label Edge Routers (LERs) and Label Switched Routers (LSRs) are the two types of devices that take part in MPLS protocol mechanisms.

An LSR is a high-speed router device in the core of an MPLS network. It takes part in the establishment of LSPs using the appropriate label signaling protocol and high-speed switching of the data traffic based on the above-established paths.

An LER is a device that operates at the edge of MPLS networks. It supports multiple ports connected to dissimilar networks and forwards this traffic on to the MPLS network after the establishment of LSPs, using label signaling protocols at the ingress and distributing the traffic back to the access networks at the egress. LERs play an important role in the assignment and removal of labels as traffic enters or exists an MPLS network.

4.2.2 Forward Equivalence Class

Forward equivalence class represents a group of packets that share the same requirements for their transport. Each packet in such a group is provided the same treatment on its route to the destination. In MPLS, the assignment of a particular packet to a particular FEC is done once, as the packet enters the network as opposed to conventional IP routing. Each LSR builds a table, called a label information base (LIB) to specify how a packet must be forwarded. This table comprises of FEC-to-label bindings.

4.2.3 Labels and label bindings

A label identifies the path a packet should traverse. It is carried or encapsulated in a Layer-2 header along with the packet. The receiving router determines the next hop by examining the label in the packet. Once a packet has been labeled, its journey through the backbone is based on label switching.

A label is assigned to a packet after it has been classified as a new or existing FEC. These label values are derived from the data link layer lying under. Layer-2 identifiers such as data link connection identifiers (DLCIs) in the case of frame-relay networks or virtual path identifiers (VPIs) or Virtual Channel Identifiers (VCIs) in case of ATM networks can be used as labels.

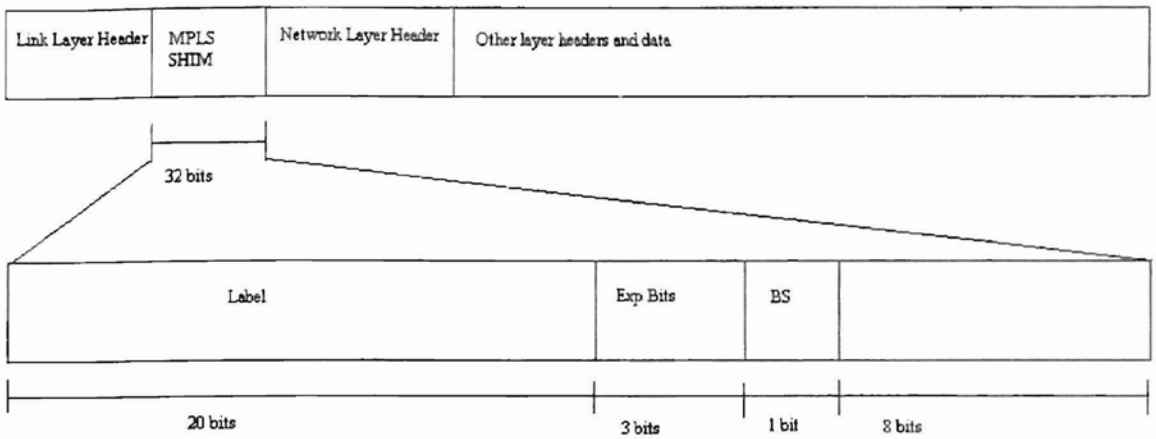

FIG 4-1. Generic MPLS Label format

4.2.4 Label Creation

The methods used in the creation of labels are

-Topology-based method – this method uses normal processing of routing protocols (OSPF and BGP).

-Request-based method – this method uses processing of request-based control traffic (RSVP)

-Traffic-based method – this method uses the reception of a packet to trigger the assignment and distribution of a label.

The first two methods are examples of control-driven label bindings while the third method is an example of data-driven bindings.

4.2.5 Label Distribution

There are various methods of signaling for label distribution. Existing routing protocols, like border gateway protocol (BGP), have been enhanced to piggyback the label information within the contents of the protocol. RSVP has also been extended to support this. A new protocol called Label Distribution Protocol (LDP) has been defined for explicit signaling and management of label space.

The following are the various schemes for label exchange:

-LDP – It maps unicast IP destination address into labels

-RSVP, CR-LDP – They are used for traffic engineering and resource reservation

-Protocol-independent multicast (PIM) – It is used for multicast states label mapping

-BGP – It is used for external labels and finds application in Virtual Private Networks (VPN)

4.2.6 Label-Switched Paths (LSPs)

In an MPLS domain, a path is set up for a given packet to travel based on FEC. This path is set up before the transmission of data. The following two options are provided by MPLS to set up an LSP.

- Hop-by-hop routing – In this method, each LSR independently select the next hop for a given FEC. This is similar to the method currently being used in IP networks.
- Explicit routing – This is similar to source routing. The ingress LSR specifies the list of nodes through which the LSP traverses. Along the path,

resources can be reserved to ensure QoS to the data traffic. This eases traffic engineering throughout the network.

The LSP set up for a given FEC is unidirectional and the return traffic must take another LSP.

4.2.7 Label Spaces

The following labels are used by an LSR for FEC-label bindings:

- Per Platform – The label values are unique across the whole LSR. The labels are allocated from a common pool. Two labels distributed on different interfaces cannot have the same value.
- Per Interface – The label rages are associated with interfaces. Multiple label pools are defined for interfaces and labels for these interfaces are allocated from separate pools. Two labels distributed on different interfaces could be the same.

Stream merging is the merging of the incoming streams of traffic from different interfaces and switching them using a common label if they are traversing the network toward the same final destination. If the underlying transport network is ATM, LSRs can use virtual path or virtual circuit merging. In this case, cell-interleaving problems must be avoided.

4.2.8 Label Retention

If label bindings are received from LSRs, that are not the next hop for a given FEC, the following two modes are defined.

- Conservative – Here, the bindings between a label and an FEC received from LSRs are discarded if they are the next hop for a given FEC. An LSR in this mode requires maintaining fewer nodes and is the recommended mode for ATM-LSRs.
- Liberal – Here, the bindings between a label and an FEC received from LSRs are retained even if they are not the next hop for a given FEC. This mode allows for quicker adaptation to topology changes.

4.2.9 Label Control

The following modes for distribution of labels to neighboring LSRs are defined in MPLS.

- Independent – Here, an LSR recognizes a particular FEC and makes the decision to bind a label to the FEC independently for distributing the binding to its peers.
- Ordered – Here, an LSR binds a label to a particular FEC if and only if it is the egress router or it has received a label binding for the FEC from its next hop LSR. This mode is recommended for ATM-LSRs.

4.2.10 Signaling Mechanisms

Two types of signaling mechanisms are provided in MPLS

- Label Request – In this mechanism, an LSR requests a label from its downstream neighbor so that it can bind to a specific FEC. This can be employed down the chain of LSRs up until the egress LER.

- Label mapping – In this mechanism, a downstream LSR, in response to a label request will send a label to the upstream initiator using the label mapping mechanism.

4.2.11 Label Distribution Protocol (LDP)

This is a new protocol for the distribution of label binding information to LSRs in an MPLS network. The FECs are mapped to labels using LDP, which, in turn, create LSPs. LDP sessions are established between LDP peers in the MPLS network and these peers exchange the following LDP messages.

- Discover messages – These messages announce the presence of an LSR in a network.
- Session messages – These messages establish, maintain and terminate sessions between LDP peers.
- Advertisement messages – These messages create, change and delete label mappings for FECs.
- Notification messages – These messages provide advisory information and signal error information.

4.2.12 Traffic Engineering

The process that enhances the overall network utilization by attempting to create a uniform or differentiated distribution of traffic throughout the network is traffic engineering. This process helps in the avoidance of congestion on any one path. It is possible, in traffic engineering, for two packet data flows, the packets may traverse completely different paths even though their originating and final destination nodes maybe the same. Thus, the less-used network parts can be used and differentiated services can be provided.

Traffic engineering is inherently provided in MPLS using explicitly routed paths. Two possible approaches to supply dynamic traffic engineering and QoS in MPLS are RSVP and CR-LDP.

4.2.13 Constraint-based routing (CR)

CR takes into account various parameters like bandwidth, delay, hop count and QoS. The LSPs established could be CR-LSPs where the constraints can be explicit hops or QoS requirements. QoS requirements specify which links and queuing or scheduling mechanisms are to be employed for the flow. Explicit hops specify which path is to be taken.

CR may select a longer but less utilized path. Nevertheless, as CR increases network utilization, it adds more complexity to route calculations, as the path selected must satisfy the QoS requirements of the LSP. CR can be used in conjunction with MPLS to set up LSPs.

4.3 MPLS Operation

The steps needed for a data packet to travel through an MPLS domain are

1. Label creation and distribution
2. Table creation at each router
3. Label-switched path creation

4. Label insertion/table lookup

5. Packet forwarding

4.3.1 MPLS Actions

1. Label Creation and Distribution: The routers make decision to bind a label to a specific FEC before the flow of any traffic begins and they build their tables. Downstream routers in LDP initiate the distribution of labels and label/FEC binding. The traffic-related characteristics and MPLS capabilities are negotiated using LDP. LDP uses TCP as a reliable and ordered transport protocol.

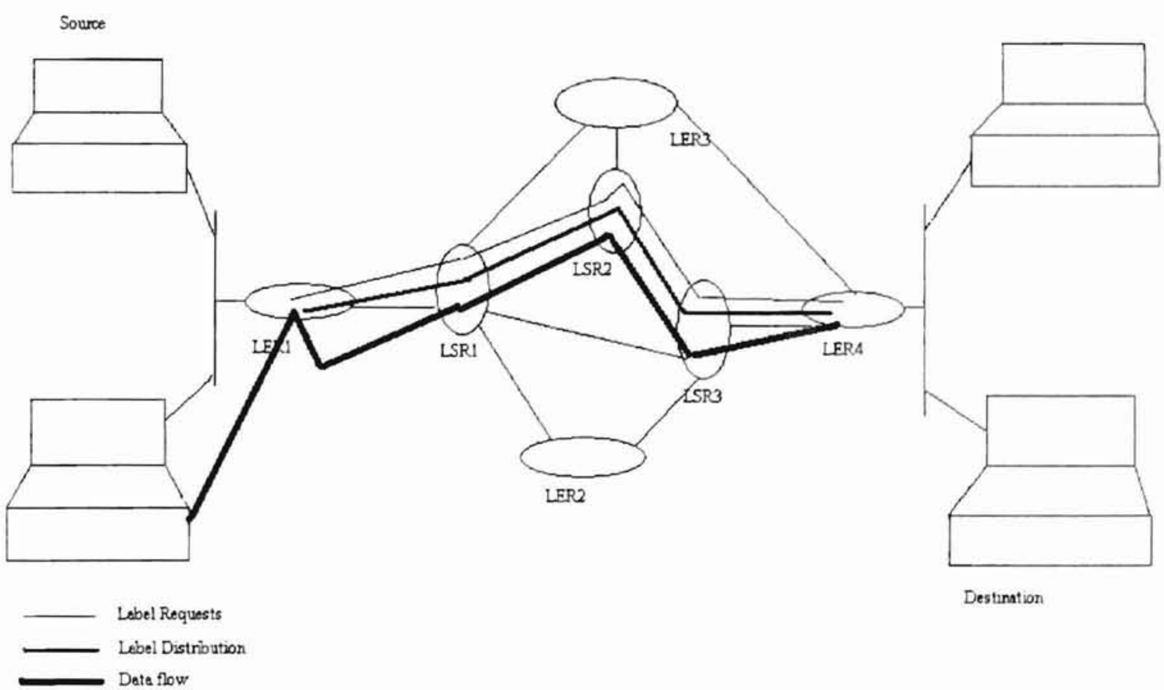

FIG 4-2. LSP Creation and Packet Forwarding through a MPLS Domain

2. Table creation: When an LSR receives label bindings, it creates label information base (LIB). The contents of this table specify the mapping between a label and an FEC.

3. Label switched path creation: The mid thick lines in figure 4-2 show the creation of LSPs. The LSPs are created in a direction reverse to the creation of entries in the LIBs.

4. Label insertion/table lookup: The first router (LER1 in figure 4-2) uses the LIB table to find the next hop and request a label for a specific FEC. Subsequent routers just use the label to find the next hop. Once the packet reaches the egress LSR (LER4), the label is removed and the packet is supplied to the destination.

5. Packet Forwarding: The path of a packet is examined with reference to figure2. The path is from the ingress LSR, LER1, to the egress LSR, LER4.

> 1. LER1 may not have any labels for this packet, as it is the first occurrence of this request. In an IP network, it will find the longest address match to find the next hop. Let LSR1 be the next hop.
>
> 2. LER1 will initiate a label request to LSR1.
>
> 3. The request will be propagated through the network as indicated by the least thick lines.
>
> 4. Each intermediary router receives a label from its downstream router starting from LER2 and going upstream until LER1. The mid thick lines using any signaling protocol like, for example, LDP indicate the LSP setup.

5. LER1 inserts the label and forwards the packet to LSR1.

6. Each subsequent LSR examines the label in the received packet and replaces it with the outgoing label and forwards it.

7. When the packet reaches LER4, it removes the label as the packet is departing from an MPLS domain and delivers it to the destination.

8. The thickest lines indicate the actual data path.

4.3.2 Tunneling in MPLS

MPLS can control the entire path without explicitly specifying the intermediate routers. Creating tunnels through the intermediary routers that can span multiple segments does this.

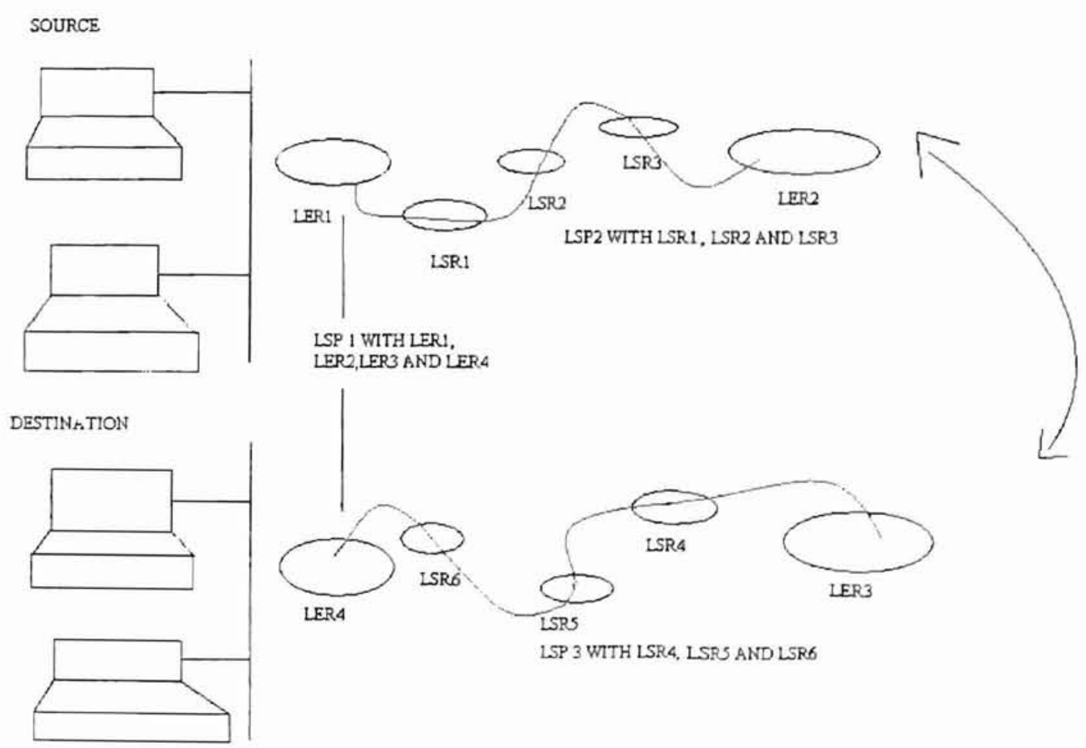

Figure 4-3. Tunneling in MPLS

Consider the figure above. LERs (LER1, LER2, LER3 and LER4) use BGP and create an LSP between them, which is LSP1 as shown in figure4- 3. These LERs use LDP to receive and store labels from the egress LER (LER4) all the way to the ingress router (LER1).

Nevertheless, for LER1 to send its data to LER2, it must go through 3 LERs. Therefore, a separate LSP (LSP2 in figure 4-3) is created between LER1 and LER2 and this spans LSR1, LSR2 and LSR3. This represents a tunnel between the two LSRs. The labels used in this path are different from the labels that the LERs created for LSP1. This is also true for LER3 and LER4 and for LSRs in between them. LSP3 is created for this particular segment. Label stack is used when transporting the packet through two network segments. As a packet travels through LSP1, LSP2 and LSP3 it carries two complete labels at a time.

As the packet exists the first network and is received by LER3, it removes the label for LSP2 and replaces it with LSP3 label, also swapping LSP1 label within the packet with the next hop label. LER4 removes both labels before it sends the packet to the destination.

4.4 Applications of MPLS

MPLS addresses the requirements of the present network backbone by accomplishing the following:

1. MPLS improves packet-forwarding performance in the network

- It enhances packet forwarding through routers using Layer-2 switching functions.

- It is very simple to implement

- It enables routing by switching at wireline speeds and thus increases the performance of the network.

2. MPLS supports QoS and CoS for service differentiation.

   - It uses traffic-engineered path setup.

   - It has provisions for constraint based and explicit path setup

3. MPLS supports network scalability.

4. MPLS integrates ATM and IP in the network.

   - It acts as a bridge to connect access IP and core ATM networks.

   - It can join the two disparate networks by reusing existing router/ATM switch hardware.

5. MPLS build interoperable networks.

   - It achieves synergy between IP and ATM networks.

   - It facilitates IP-over-SONET integration in optical switching.

   - It helps build scalable VPNs which support traffic engineering.

CHAPTER V

VCID Notification over ATM link for LDP

The Asynchronous Transfer Mode label Switching Router (ATM-LSR) is a major application of label switching. The ATM layer labels (VPI and VCI) associated with a VC are rewritten with a new value at every ATM switch node and so they cannot identify a Virtual Circuit (VC) in label mapping messages. Virtual Channel Connection Identifier (VCID) is introduced to solve the above problem as VCID has the value at both ends of a VC. The procedures for the communication of VCID values between neighboring ATM-LSRs are given in this chapter.

5.1 Introduction

The ATM-LSR is one of the major applications of label switching to integrate Layer 2 and Layer 3. In MPLS, streams are referred to classes of packets that have some common characteristic that can be deduced by examination of the layer 3 header in the packets. These streams are bound to layer 2 'labels' and these bindings are conveyed between peer LSRs by means of a Label Distribution Protocol (LDP).

To apply MPLS to ATM links, there must be a way to identify ATM VCs in LDP mapping messages and VCID is used for this purpose.

5.2 VCID Notification procedures

The types of VCs in ATM are:

1. Transparent point-to-point link

2. Virtual Path (VP)

3. Permanent Virtual Circuit (PVC)

4. Switched Virtual Circuit (SVC)

There are two broad categories of VCID Notification procedures; inband and outband and these are based on the connection over which the messages of the VCID notification procedures are forwarded. For inband procedures, the messages are forwarded over the VC to which they refer. However, for outband procedures, the messages are forwarded over some other connection than the VC to which they refer.

The various types of link and the VCID notification procedures used for each are listed below.

1. Transparent point-to-point link: no VCID notification

VCID notification procedure is not needed as the label, (VPI/VCI), is the same at each end of the VC.

2. VP: inband or VPID or no notification

Inband notification or VPID notification:

VCID notification is needed, as the VPI at each end of the VC may not be the same. Inband VCID notification or VPID notification is used in this case.

No notification:

VCID notification is not necessary if a node has only one VP to a neighboring node and the VCI can be used as the VCID.

3. PVC: inband notification

The labels at each end of the VC may not be the same and inband VCID notification is used.

4. SVC: outband notification or outband notification using a small-sized field

Outband notification:

The VCID is carried directly in it if a signaling message has a field that is large enough to carry a VCID value.

Outband notification using a small-sized field:

This procedure is used if a signaling message has a field that is not large enough to carry a VCID value.

Inband notification:

This procedure is used when the signaling message cannot carry user information.

5.3 VCID Notification

5.3.1 Inband Notification Procedures

5.3.1.1 Inband Notification for point-to-point VC

VCID notification is performed by transmitting a control message through the VC newly established for use as a label switched path (LSP). The procedure for VCID notification between two nodes A and B is given below.

1. The node A establishes a VC to the destination node B by signaling or management.
2. The node A selects a VCID value.
3. The node A sends a VCID PROPOSE message containing the VCID value and a message ID through the VC to the node B.

4. The node A establishes an association between the outgoing label (VPI/VCI) for the VC and the VCID value.

5. The node B after receiving the message, establishes an association between the VCID in the message and the incoming label (VPI/VCI) for the VC.

6. The node B sends an ACK message to the node A containing the same VCID and message ID as in the received message.

7. Node A, after receiving the ACK message, checks whether the VCID and the message ID in the message are the same as the registered ones. If they are not, the message is ignored. Else, node A regards that node B has established the association between the VC and VCID.

8. Node A now sends an LDP REQUEST message to the node B containing the message ID used for VCID PROPOSE. When node B receives the LDP REQUEST message, node B regards that the node A has received the ACK correctly. The message exchange using VCID PROPOSE, VCID ACK and LDP REQUEST message constitutes a 3-way handshake. After the completion of 3-way handshake, node B ignores all VCID PROPOSE messages received over the VC. The node B sends an LDP Mapping message containing the VCID value in the label TLV.

5.3.1.2 Inband notification for point-to-multipoint VC

Multicasting is not supported by current LDP specifications. But VCID notification is defined for multicasting. The upstream node assigns the VCID value and notifies the downstream node of that value.

The procedure for establishing the first VC is described.

1. The upstream node assigns a VCID value for the VC. If a VCID has already been assigned to a VC, it is used for VCID.
2. The upstream node sends a message containing the VCID value and a message ID through the VC and this message is transferred to all the leaf nodes.
3. The upstream node establishes an association between the outgoing label for the VC and the VCID value.
4. If the downstream node receiving the message already received the LDP REQUEST message for the VC, it discards the received message. Else, the downstream nodes establish an association between the VCID in the message and the VC from which the message was received.
5. The downstream node sends an ACK message to the upstream node.
6. After receiving the ACK messages, the upstream and downstream nodes share the VCID. The upstream node sends the LDP REQUEST message to make a 3-way handshake.

The Procedure for adding a leaf to the existing point-to-multipoint VC is described.

1. The upstream node adds the downstream node using the ATM signaling.

2. The VCID value already assigned for the VC is used.

3. The upstream node sends a message containing the VCID value and a message ID and this is transferred to all the leaf nodes.

4. If the downstream node receiving the message already received the LDP REQUEST message for the VC, the received message is discarded. Else, the node establishes an association between the VCID in the message and the VC from which the message was received.

5. After the upstream node receives the ACK messages, the upstream and downstream nodes share the VCID. The upstream node sends the LDP REQUEST message to make a 3-way handshake.

5.3.2Outband Notification Procedures

5.3.2.1 Notification using a small-sized field

This message is applied when a VC is established using an ATM signaling message and the message has a field that is not large enough to carry a VCID value. SETUP message used to establish connections between the source and the destination has a 7-bit mandatory field for the user. This is a user specific field in the Layer 3 protocol field in the BLLI IE (Broadband Low Layer Information Information Element). The BLLI value is used as a temporary identifier for a VC during VCID notification procedure. This procedure is the "Outband Notification using a small-sized field".

5.3.2.1.1. Outband Notification using a small-sized field for a point-to-point VC

The upstream LSR assigns a VCID value and the procedure for the notification of VCID is as follows.

1. An upstream LSR establishes a VC to the downstream LSR using ATM signaling and a value that is not currently used for any other VCID notification transaction with this peer is supplied in the BLLI field.

2. The upstream LSR sends the VCID PROPOSE message to notify the downstream LSR of the association between the BLLI and VCID values.

3. The downstream node establishes the association between the VC with the VCID and BLLI value and sends an ACK message to the upstream LSR.

4. After receiving the ACK message, the upstream LSR establishes the association between the VC and VCID and now the VC is ready to be used and BLLI value is ready for reuse.

5. The upstream node sends LDP REQUEST message to the downstream node containing the VCID value in the label TLV of LDP.

5.3.2.1.2 Outband Notification using a small-sized field for a point-to-multipoint VC

This subsection describes procedures for establishing the first VC for a multicast tree adding a new VC leaf to an existing VC tree.

The procedure for establishing the first VC is described.

1. An upstream LSR establishes a VC by the ATM signaling to the downstream LSR with a unique BLLI.
2. The upstream LSR notifies the downstream LSR of a paired BLLI value and VCID using a message exclusive for this purpose.
3. The downstream LSR establishes the association between the VC with the BLLI value and the VCID and sends an ACK message to the upstream LSR.
4. After ACK message is received by the upstream LSR, the VC is ready for use and the BLLI value can be reused for another VC.

The procedure for adding a leaf to the existing point-to-multipoint VC is described.

1. The upstream LSR establishes a VC by ATM signaling to its downstream LSR with the BLLI value used during the first signaling procedure. If another VC is making use of the BLLI at the same time, the upstream node waits for the completion of the signaling procedure that is using this BLLI value.
2. The upstream LSR notifies the downstream LSR of a paired BLLI value and VCID using a message exclusive for this purpose.
3. The downstream LSR establishes the association between the VC with the BLLI value and the VCID and sends an ACK message to the upstream LSR.
4. After ACK message is received by the upstream LSR, the VC is ready for use and the BLLI value can be reused for another VC.

5.3.3 Outband Notification

This method is used when a VC is established using an ATM signaling message having a field large enough to carry a VCID value. The connection is established first between nodes A and B and then VCID notification is done. Node A, then, sends the LDP request message to node B. Node B responds with a LDP mapping message to the node A.

5.4 VPID Notification Procedure

The VCID notification procedures can also be applied to share the same identifier between both ends of a VP. VPID notification procedure is used for this and a distinct procedure is used for each direction of each VP.

The steps involved in VPID notification procedure are given below and these are similar to the VCID notification procedures.

An upstream node sends the VPID PROPOSE message to the downstream node and the downstream node responds with a VPID ACK message. After receiving the ACK message, the upstream node sends the LDP Label Request message and the downstream node sends the LDP Label Mapping message.

Thus, the VCID and VPID notification procedures over ATM for LDP were discussed in this chapter.

CHAPTER IV

RESULTS AND DISCUSSIONS

In Chapter IV, the procedures for VCID Notification over ATM link for LDP were discussed. ATM signaling was used to establish a VC between a source and a destination first in this case and then VCID Notification procedures were sent as a null encapsulation packet through the VC to be used as an LSP. In this chapter, we look into two possibilities for the establishment of a VC by making use of ATM PNNI signaling procedures rather than Q.2931. In addition, the advantage of using ATM PNNI in terms of physical link or QoS demand failure is also discussed.

6.1 Interoperating ATM PNNI for MPLS networks

As discussed in the previous chapter, the VC from the source to the destination is established directly by using ATM signaling procedure Q.2931. Two methods to establish this connection between source and destination by using PNNI routing protocol and interoperating it with MPLS is discussed in this section.

6.1.1 Loose CR-LDP Routing

PNNI protocol uses routing hierarchy to represent the entire network. So, the source node or an entry border node, while constructing a DTL for traversing a peer group, constructs it as abstract nodes in each level rather than specifying the entire path through a peer group. Thus, PNNI routing and signaling can be combined with LDP for VCID notification over ATM link and this procedure is the loose CR-LDP routing procedure. Loose CR-LDP routing supports loose explicit routes by the use of abstract nodes and/or hierarchical explicit routing without the need to use LDP for hop-by-hop LSP setup.

The procedure for Inband VCID notification between two nodes A and B using PNNI signaling for loose routing CR-DP is given below.

1. Node A establishes a connection to the node B by using the PNNI routing protocol.
2. Node A selects a VCID value.
3. Node A sends a VCID PROPOSE message containing the VCID value and a message ID to node B. Node A establishes the association between the outgoing label (VPI/VCI) for the VC and the VCID value.
4. The node B after receiving the message establishes an association between the VCID in the message and the incoming label (VPI/VCI).
5. The node B sends an ACK message to node A This message contains the same VCID and message ID as in the received message.
6. Node A, after receiving the ACK message, checks if the VCID and the message ID are the same as the registered ones and if not the message is ignored. If node A does not receive the ACK message with the expected message ID and VCID during a given period, the node A resends the VCID PROPOSE message to the node B.
7. After receiving the proper ACK message, node A sends an LDP REQUEST message to node B containing the message ID used for VCID PROPOSE. The

message exchange using VCID PROPOSE, VCID ACK AND LDP REQUEST message constitutes a 3-way handshake.

8. Node B sends a LDP mapping message with the VCID value in the label TLV.

6.1.2 Strict CR-LDP Routing

In this method, the complete route from the source to destination is established using PNNI routing and signaling and then the VCID Notification procedures are sent through LDP. This method is applicable only when flooding to all the nodes is possible and concepts of Peer Group Level (PGL) or hierarchical levels are eliminated. We must make use of an interworking function to separate the CR-LDP Notification functions from the PNNI routing and signaling function. The CR-LDP notification procedures and the PNNI routing and signaling function must function independently.

The procedure for Inband VCID notification between two nodes A and B using PNNI signaling for strict routing CR-DP is given below.

1. Node A establishes a connection to node B using PNNI routing protocol.
2. There must be an interworking function separating the PNNI routing protocol from CR-LDP specifications.
3. Node A selects a VCID value.
4. Node A sends a VCID PROPOSE message containing the VCID value and message ID to node B. Node A establishes the association between the outgoing label (VPI/VCI) for the VC and the VCID value.
5. The node B after receiving the message establishes an association between the VCID in the message and the incoming label (VPI/VCI).

6. The node B sends an ACK message to node A This message contains the same VCID and message ID as in the received message.
7. Node A, after receiving the ACK message, checks if the VCID and message ID are the same as the registered ones and if not the message is ignored. If node A does not receive the ACK message with the expected message ID and VCID during a given period, the node A resends the VCID PROPOSE message to the node B.
8. After receiving the proper ACK message, node A sends an LDP REQUEST message to node B containing the message ID used for VCID PROPOSE. The message exchange using VCID PROPOSE, VCID ACK AND LDP REQUEST message constitutes a 3-way handshake.
9. Node B sends a LDP mapping message with the VCID value in the label TLV.

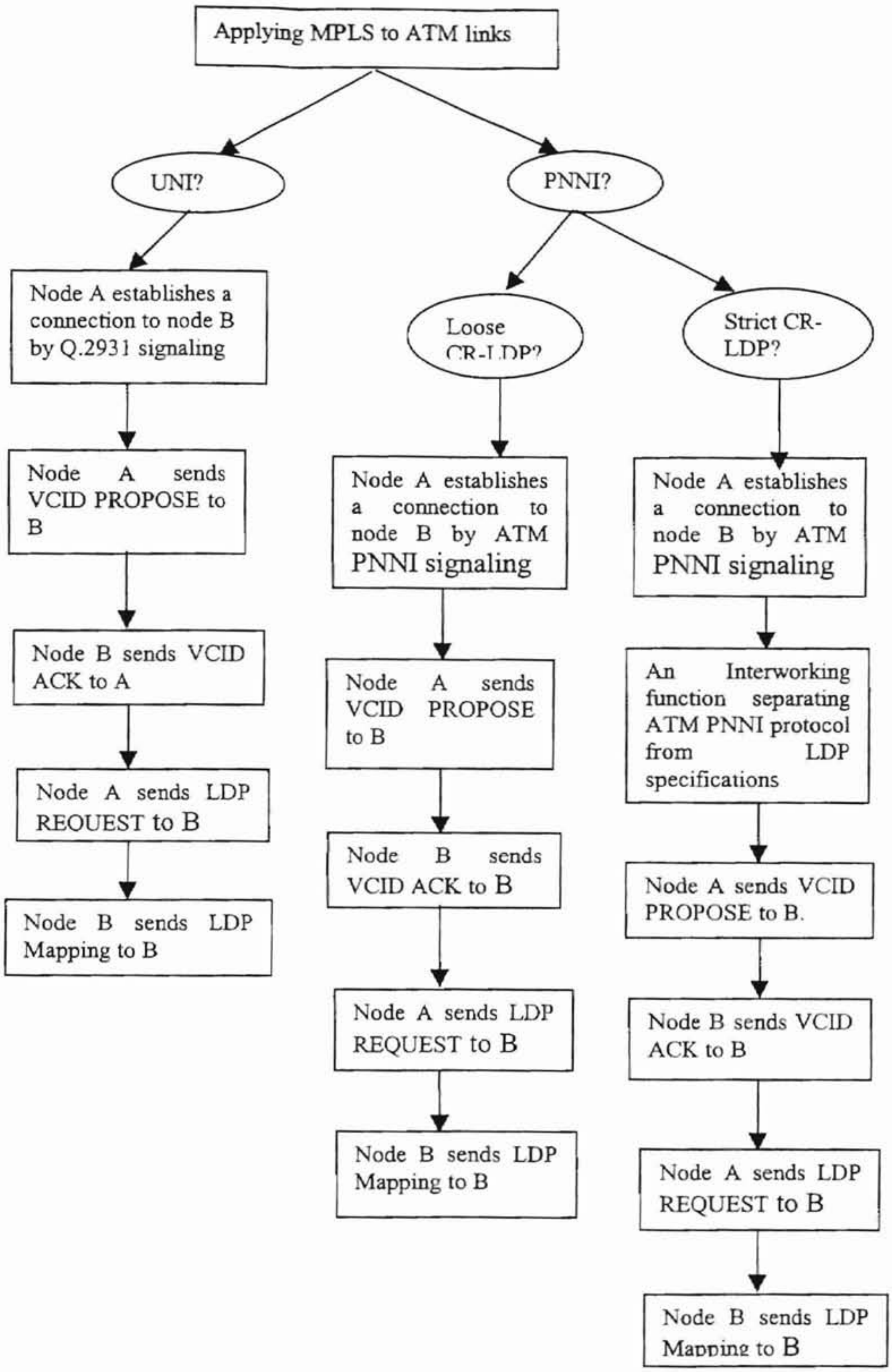

FIG 6-1 Flowchart representing the ATM PNNI-MPLS interoperability

6.2 Reliability and Utilization efficiency

After a connection has been established in an ATM UNI network, there can be physical link failure and QoS demand failure. The physical link failure can be either due to cut in the fiber, laser malfunction, dirty fiber and many other factors. The QoS demand failure can be due to the network not being able to meet the bandwidth and delay requested by the user. MPLS allows for preemptions in the second case.

MPLS supports both high and low priority services. When a high priority service requests for resources like say, bandwidth, and if they are not available, the low priority services are preempted to provide the requested resources to the high priority service. An example of high priority services can be MPEG video. The bit rate made available to them can be either Variable Bit Rate (VBR) or Constant Bit Rate (CBR). The bit rate made available to low priority services can be VBR, Available Bit Rate (ABR), and Unspecified Bit Rate (UBR).

In an ATM UNI network, when there is a failure either due to the physical link or due to the requested services being not met, the time involved to provide path or line restoration is large compared to the restoration that can be achieved by making use of an ATM PNNI network.

In an ATM UNI network when there is a failure, this failure must be flooded throughout the network indicating the failed link and the new resources available. This updated information must also reach the source node and then the source node must try for a new path establishment in such a way that it avoids the failed link or path. This is nothing but source routing where the source tries to find a new path to the destination avoiding the failed path or link. This is time consuming as the updated information must be flooded all the way until it reaches the destination and for the source to find a new path.

The time involved in restoration can be made comparatively less in an ATM PNNI network as it is not necessary for the updated information to be flooded all the way back to the source node to find an alternate path. We discuss below how this restoration time can be made less in an ATM PNNI network compared to an ATM UNI network.

As we know, in an ATM PNNI network the network is organized in a hierarchical manner. There are hierarchical levels in an ATM PNNI network. A collection of lowest level nodes forms a peer group. This peer group is represented in the next hierarchical level as a logical group node. Ever higher levels of peer groups are created until a single highest level peer group represents the entire network. Each peer group has a peer group leader (PGL). The PGL has complete topology state information from all nodes in the peer group node. These PGLs create the Designated Transit Lists (DTLs) that is a complete path across a peer group consisting of a sequence of node Ids. The PGLs also store the DTLs created by them.

If there is a link failure or blocking, the crankback procedures are started back to the origin of the particular DTL that will be the nearest PGL. The crankback message has information about the failed link. The PGL can then update its topology database and try to find an alternate path to the destination so that it avoids the failed path. We see that it is not necessary for the crankback messages to be flooded all the way back to the source node and then for the source node to update its topology database and find an alternate path to the destination. The PGL finds an alternate path and we shall call this type of restoration as peer group restoration. To enable peer group restoration we need to use loose CR-LDP routing concept.

CHAPTER VII

CONCLUSIONS

In chapter 1, a description of the PNNI signaling and routing protocol functions was given. In chapter 2, a detailed description of the Q.2931 signaling procedures used in ATM UNI networks was presented. These two chapters gave us a clear picture of the how ATM PNNI signaling features differ from that of the ATM UNI signaling features. In chapter 3, a general introduction to the structure and working of MPLS networks was presented. In chapter 4, the different types of VCID/VPID notification procedures over ATM for LDP were discussed. This all formed the background research part of this thesis.

In chapter 5, two methods of interoperating ATM PNNI with MPLS networks were discussed. There was a problem of delay in the restoration in case of link or QoS demand failures in ATM-MPLS networks.

The two methods proposed to interoperate ATM PNNI with MPLS were loose CR-LDP routing and strict CR-LDP routing. Also, ATM PNNI routing with its features of hierarchical representation, peer groups and PGL, will be able to provide restoration at a time that is comparatively less than the time needed for an ATM UNI network. This is possible due to the ATM PNNI's additional features of crankback and alternate routing. The failed path information can be crankbacked to the nearest PGL and this PGL can find an alternate route to the destination instead of waiting for the message to reach the source node and for the source node to find an alternate path to the destination. This is the conclusion of the thesis.

CHAPTER V11

REFERENCES

[1] C. Casetti, J. C. De Martin, and M. Meo, "A Framework for the Analysis of Adaptive Voice over IP," former IETF Draft, currently removed from IETF links.

[2] P. Van Mieghem, "A lower bound for the end-to-end delay in networks: Application to Voice over IP," *Proceedings of Globecom '98*, pp. 2508 –2513, 1998.

[3] B. Bollobas, *Random Graphs*. London: Academic Press, 1985.

[4] H. Bruneel and B. G. Kim, *Discrete-Time Models for Communication System Including ATM.* Boston: Kluwer Academic Publishers, 1993.

[5] K. Calvert, M. Doar, and E. Zegura, "Modeling internet topology." *IEEE Commun. Magazine*, pp. 160-163, June 1997.

[6] D. De Vleeschauwer, B. Steyaert, and G. H. Petit, "Performance analysis of voice over packet based networks." submitted to ITC 16, Edingburgh, U.K., June 7-10, 1999.

[7] P. Erdos and A. Renyi, *On random graphs I.* Publ. Math. Debrecen., 6:290-297, 1959.

[8] P. Erdos and A. Renyi, *On the evolution of random graphs.* Magyar Tud. Akad. Mat. Kutato Int. Kozl., 5:17-61, 1960.

[9] E. N. Gilbert, *Enumeration of labeled graphs.* Canad. J. Math., 8:405-411, 1956.

[10] Leonard Kleinrock. *Queueing Systems,* volume 1: Theory. N.Y.: John Wiley and Sons, 1975.

[11] B. I. Shklovskii and A. L. Efros, *Electronic Properties of Doped Semiconductors.* Berlin: Springer Verlag, 1984.

[12] K. Van der Wal, M. Mandjes, and H. Bastiaansen, "Delay Performance analysis of the new internet services with guaranteed qos." *Proceedings of the IEEE*, 85(12), pp. 1947-1957, Dec. 1997.

[13] J. Walrand, *An Introduction to Queueing Networks*. New York: Prentice-Hall, 1988.

[14] B. M. Waxman, "Routing of multipoint connections." *IEEE Journal on Selected Areas in Commun.*, 6(9), pp. 1617-1622, Dec. 1998.

[15] P. Mishra and H. Saran, "Capacity Management and Routing Policies For Voice Over IP Traffic," *IEEE Commun.Mag.*, vol.14, pp. 24-33 ,Mar.-Apr. 2000.

[16] O. Hagsand, K. Hanson, I. Marsh, "Measuring Internet Telephony Quality: Where are we today?" *Proceedings of the IEEE Global Telecommunications Conference*, vol. 3, pp 1838-1842, 1999.

[17] G. Ash, *Dynamic Routing in Telecommunication Networks*, Mc-Graw-Hill, 1998.

[19] Wroclawski., "The Use of RSVP with Integrated Services," RFC 2210, NIC, 1997.

[20] Braden, "Resource Reservation Protocol (RSVP) – Version 1 Functional Specification," RFC 2205, NIC, 1997.

[21] Callan, "A Framework for Multiprotocal Label Switching," IETF Draft, NIC, 1998.

[22] Nichols, "Definition of the Differentiated Services Field (DS field) in the IPV4 and IPV6 Headers," RFC 2474, NIC, 1998.

[23] P. Goyal, "Integration of Call Signaling and Resource Management for IP Telephony," *IEEE Networks*, May 1999.

[24] R. Guerin, "Aggregating RSVP-Based QoS Requests," IETF Draft, NIC, 1999.

[25] H.Liu and P. Mouchtaris, "Voice over IP Signaling: H.323and Beyond," RFC 2474, NIC, 1998.

[26] A. Kankkunen, G. Ash, A. Chiu, J. Hopkins, J. Jeffords, F. Le Faucheur, B. Rosen, D. Stacey, A. Yelundur, and L. Berger, "VoIP over MPLS Framework," IETF Draft, http://www.ietf.org/internet-drafts/draft-kankkunen-vompls-fw-01.txt, Jan. 2001.

[27] "Private Network-Network Interface Specification 1.0 (PNNI 1.0)," The ATM Forum, af-pnni-0055.000, Mar. 1996

[28] "Recommendation Q.2931", ITU-T, Feb. 1995

[29] www.ietf.org/html.charters/mpls-chartet.html

[30] J.-M. Chung, (*Invited Paper*) "Analysis of MPLS Traffic Engineering," *Proceedings of the IEEE Midwest Symposium on Circuits and Systems 2000 conference* (IEEE MWSCAS'00), East Lansing, Michigan, USA, Aug. 8-11, 2000.

[31] J.-M. Chung, (*Invited Paper*) "Wireless Multiprotocol Label Switching," *Proceedings of the 35th Asilomar Conference on Signals, Systems and Computers 2001*, Pacific Grove, California, USA, Nov. 4-7, 2001.

[32] J.-M. Chung, "Report-3: Performance Enhancement and Service Improvement Recommendations for Multiprotocol Label Switching and Multicasting Applications,: Technical Project Report, Williams Communications, June 29, 2001.

[33] J.-M. Chung, E. Marroun, H. Sandhu, and S-C. Kim. "VoIP over MPLS Networking Requirements," *Proceedings of the IEEE International Conference on Networking 2001* (IEEE ICN'01), Colmar, France, July 9-13, 2001.

2

VITA

Rajpamal R. Pethuraj

Candidate for the Degree of

Master of Science

Thesis: ATM PNNI INTERFACING ISSUES WITH MPLS NETWORKING

Major Field: Electrical Engineering

Biographical:

Personal Data: Born in Madurai, India, On November 1, 1977, the son of P. Raveenthiran and Rani Raveenthiran.

Education: Received a Bachelor of Engineering degree in Electronics and Communication Engineering from the Bharathiar University, India in May 1999. Completed the requirements for the Master of Science degree with a major in Electrical Engineering at the Oklahoma State University in August 2001.

Experience: Employed as a Research Assistant by the ACSEL Laboratories at the School of Electrical and Computer Engineering, Oklahoma State University, Aug. 2000 to present.
Employed as a Teaching Assistant in the FORTRAN Lab at the School of Electrical and Computer Engineering, Oklahoma State University, Jan. 2000 to May 2001.

Professional Membership: Student member of the honorary Institute of Electrical and Electronics Engineers, Inc. (IEEE) for the academic years 2000 and 2001.