COMMITTEE T1 CONTRIBUTION

Document No.: T1S1.5/92-165

**********************************************************************************

STANDARDS PROJECT: Broadband Aspects of ISDN

**********************************************************************************

TITLE: A B-ISDN Object-Oriented Call Model

**********************************************************************************

ISSUES ADDRESSED: B-ISDN Signalling Requirements

**********************************************************************************

SOURCE: Southwestern Bell Corporation and Washington University

Matthew Nguyen
Southwestern Bell Technology Resources
550 Maryville Centre Dr., Suite 300
St. Louis, MO 63141

Voice/Fax: (314) 529-7627/7674

Rick Bubenik
Applied Research Laboratory
Washington University, Campus Box 1045
One Brookings Drive
St. Louis, MO 63130
(314) 935-7530/7302

**********************************************************************************

ABSTRACT

An object-oriented call model for broadband communications is presented. This model brings together and encapsulates call, connection and endpoint attributes in an object containment hierarchy. The object hierarchy helps to identify and clarify control and signalling requirements needed for the support of the services being developed in the ANSI T1S1.5 and the CCITT XI/4-3 broadband working groups. To expedite the development of broadband control/signalling software, this contribution proposes that the ANSI T1S1.5 Broadband Network and Services SWG investigate: 1) the benefits of an object-oriented Control Information Base (CIB) containing a set of call control objects, and 2) the standardization of an object-oriented call model and its associated objects.

**********************************************************************************

DATE: May 11–15, 1992

**********************************************************************************

DISTRIBUTION TO: T1S1.5 Technical Subcommittee Working Group Members

**********************************************************************************

## 1 Introduction

For the purposes of network management, CCITT, ISO and the Internet community have designed and incorporated Management Information Bases (MIBs) into networks (and standards), along with associated management models for interacting with these databases. These MIBs contain *managed objects* that are accessed and manipulated by network managers and network management systems. The rationale behind adopting this object-oriented approach has been the *containment, modularity, reusability* and *extensibility* characteristics commonly associated with objects and object-oriented programming.

This contribution proposes that an analogous approach be taken for network control of B-ISDN Release 2 and 3 network services. That is, that an object-oriented Control Information Base (CIB) be established for the definition of call control objects. Associated with this CIB is a call model that describes how the functional elements access and manipulate the objects contained therein. The CIB and call model are consistent with the functional model specified in the Baseline Text For The Harmonizing Signalling Requirements (TD XI/4-37) [4] and act as a complement to this model. They provide a hierarchical structuring of the controlled objects and aid in the design and implementation of services, applications and software control systems. Additionally, they are independent of signalling protocol—many protocols can be used across the UNI and NNI to access and manipulate the objects.

## 2 Discussion

This proposal introduces a set of attribute-based call control objects and describes how the objects can be used to support the multi-party and multimedia services that are being developed in the ANSI T1S1.5 and the CCITT XI/4-3 broadband working groups. In particular, the objects and their associated call model provide a basic communication substrate upon which the broadband services can be layered, meeting the requirements of these services.

A detailed discussion of the call model appears in Appendix A. The model consists of *call*, *connection, endpoint* and *endpoint mapping* objects, with the containment relationships shown in Figure 1.

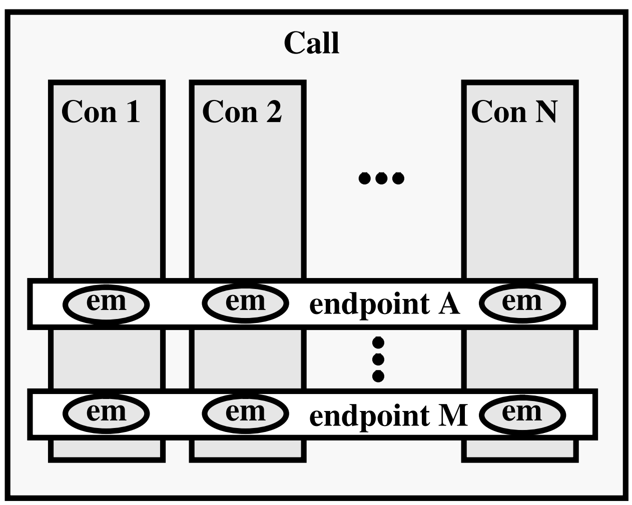


**Figure** 1. Call Control Object Containment Hierarchy.

*Call objects* contain attributes (or characteristics) that apply to the entire call. These include *type, accessibility, monitoring, modifiability,* a collection of *connection objects* and a collection of *endpoint objects*. Each connection object contains attributes that apply only to the associated connection. These include *type, bandwidth, default Rx/Tx permission* and a collection of *endpoint mapping objects*. Each endpoint mapping object describes how one endpoint is able to access the associated connection. Its attributes are *override Rx/Tx permission, Rx/Tx mapping, VPI/VCI labels* and *notification* (of connection bandwidth changes). The endpoint mapping objects associated with a particular endpoint are conceptually grouped into an endpoint object (as shown in the figure above).

The envisioned use of these objects is as follows. Every network node (or signalling transfer point) maintains a Control Information Base (CIB), Figure 2. All network nodes supporting a given call store the associated call object

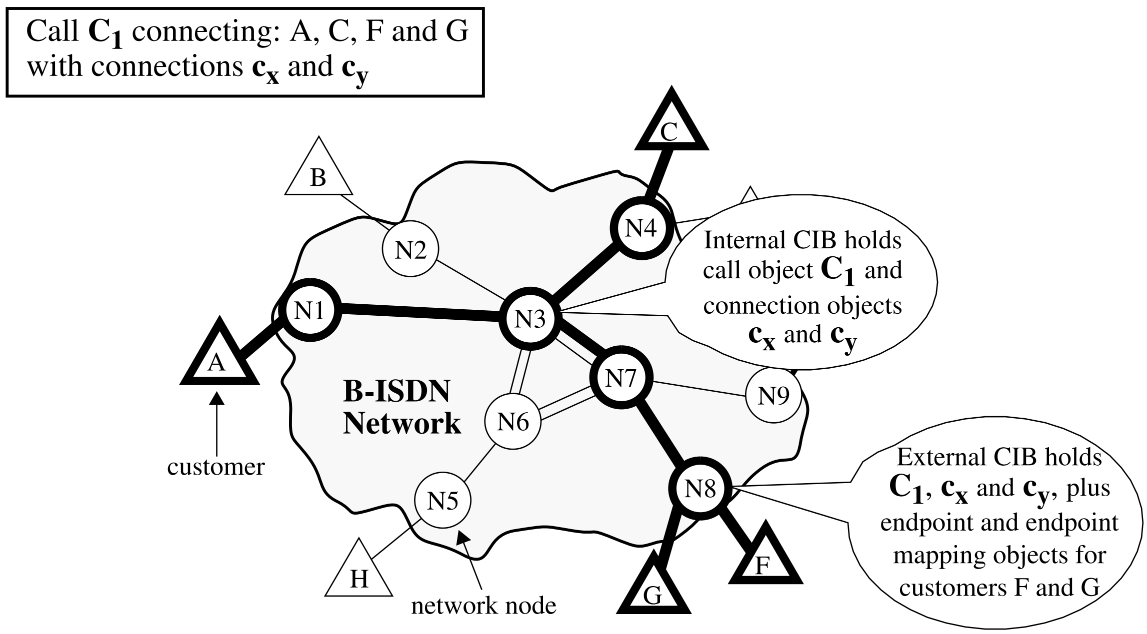


**Figure** 2. Call State Storage in Control Information Bases.

and its connection objects in the CIB. Endpoint objects and endpoint mapping objects are not maintained at every node but rather distributed to the access nodes where the endpoints connect to the network. Thus, for a given call, an access node maintains the call object, its connection objects and the subset of endpoint and endpoint mapping objects corresponding to the endpoints whose UNIs are terminated at this node. Client equipment likewise maintains some state information for each call, the exact nature of which is for further study. Signalling protocols are used to create, modify and delete calls and connections, which results in the creation, modification and deletion of associated call control objects (contained in the CIBs). Management protocols can be used as well for reading call state and remedying anomalous situations.

This call model supports the B-ISDN Release 2 and 3 service and signalling requirements, including: 1) separate invocation of the service components of a call (e.g., audio and video), 2) multi-party, multi-connection and/or multimedia communications, 3) separation of call and bearer controls, 4) on-demand change of the communication configuration of a call, and/or change of attributes of connections/endpoints, independently, 5) point-to-point, multi-point and broadcast communication configurations, 6) symmetric and asymmetric connections, 7) third-party call/connection establishment, and 8) connection grouping and/or synchronization. In addition, the object-oriented nature of the call model provides for easy extensibility to support new services and requirements.

## 3 Related Proposals

An object-oriented call model was presented by Bellcore at the March ‘92 meeting of the T1S1.5 Network and Services SWG [2]. That model proposed a collection of twelve objects and an abstract framework, which could be used to support broadband communications and services. The current proposal complements the Bellcore call model via the introduction of a core set of three call control objects, with well-defined attributes, and a concrete framework for object access and storage within a CIB. Services can be layered directly over the core call model described herein, possibly presenting a more sophisticated model (such as that proposed by Bellcore) to the application layer.

## 4 Recommendations

It is recommended that the ANSI T1S1.5 Network and Services SWG investigate: 1) the benefits of an object-oriented Control Information Base (CIB) containing a set of call control objects, and 2) the standardization of a such an object-oriented call model and associated objects similar to that proposed here.

## **Appendix**: Detailed Object-Oriented Call Model

## A.1 Introduction

This appendix presents a call model for B-ISDN Releases 2 and 3 that meets the requirements stipulated in the B-ISDN Services Baseline Document [1]. This call model: separates call, connection and endpoint attributes for separate control of each, allows multiple connections to be grouped into a single call, supports multipoint communication configurations that change during the life of the call, and allows most other call, connection and endpoint attributes (such as number of participants and connection bandwidth) to be altered dynamically during the life of the call and its associated connections. These capabilities lend themselves well to many of the B-ISDN services, including point-to-point video, videotelephony, NTSC and HDTV video distribution, multimedia services and LAN (customer premise equipment, CPE) interconnect.

This appendix is organized as follows. Section A.2 reviews several key services requirements from the B-ISDN Services Baseline Document [1]. Section A.3 presents the call model and describes the various call, connection and endpoint attributes in detail. Section A.4 summarizes the contents of this appendix and presents concluding remarks.

## A.2 Service Requirements

The B-ISDN Services Baseline Document [1] describes the requirements of several future B-ISDN services. Many of the novel and key requirements are presented in this section as a motivation for the call model features that appear in Section A.3. Most of these requirements come from the Multimedia Services section of the B-ISDN Services Baseline Document, as these services list the most demanding requirements, generally subsuming the requirements of the other services.

- Support for separate invocation of the service components of a call (such as audio and video). These service components will be carried on separate connections. Hence, the call model must group multiple connections and must allow the connections to be accessed (for example, added, dropped and modified) independently.
- Support for separation of call and connection control, with potentially different customers (or CPEs) responsible for the control of the call and the individual connections of the call. All parties need to be able to add and delete connections. Additionally, the designation of the customer (or CPE) in control of calls and connections should be transferable.
- Calls should be viewed as complex, dynamically interacting multipoint communications. This necessitates "on demand" addition and deletion of services, parties and connections within a call, and requires that connection attributes be renegotiable. For example, dynamic variation of service quality, connection capabilities and connection bandwidth are required.
- Support for point-to-point, point-to-multipoint and broadcast communication configurations, and "implementation independence" to allow flexibility in the way in which services are implemented.
- Support for multicast switching systems to complement or replace existing bridges.
- Synchronization of connections (for example, of audio and video components of a call).
- Interworking between different service modes (for example, NTSC to HDTV code conversion).
- Support for virtual path connections and virtual channel connections to the customer.
- Symmetric and asymmetric calls.
- Support for connections in which the originator is not an endpoint in the connection.

When paired with an appropriate signalling protocol, the call model presented here supports all of these service requirements. Note, however, that the model itself does not dictate a particular signalling protocol—many protocols may be able to implement the model [3][5][6].

## A.3 Call Model

A *Call* is a grouping of one or more related *connections*, where each connection is a communication channel over which two or more B-ISDN customers exchange information. Calls have global attributes, which apply to the entire call and to all connections in the call. Connections have attributes as well, which apply only to each individual connection and which can vary from one connection to another within the same call. Customers participating in a call are referred to as *endpoints* of the call. Associated with each endpoint is a collection of *endpoint mappings* (one for each connection), where attributes of the endpoint mapping determine how the associated endpoint is able to access the corresponding connection. Calls are dynamic in that the number of endpoints in the call and the call's attributes can vary during the life of the call. Connections and endpoint mappings are dynamic as well, where connections can come and go, connection attributes can change and endpoint mappings can be altered during the life of a connection.

Calls and connections are created, modified and destroyed when CPEs send B-ISDN signalling messages to the network. The dynamic nature of the call model presented here requires, of any B-ISDN signalling protocol supporting this model, the ability to dynamically alter call state. That is, protocol messages must exist to allow the call state and the number of endpoints to be altered during the life of the call. These capabilities are required, in general, for the support of B-ISDN Release 2 and 3 services.

## A.3.1 Calls

The call is the primary object manipulated by customers. When a customer creates a call or adds himself to an ongoing call, he becomes an *endpoint* in the call and gains access to all connections of the call. The creator of a call is referred to has the *owner* and has special privileges, as described below. Calls have a number of parameters that determine how customers may access the call, described in the following subsections and summarized in Table 1.

**TABLE 1. Call Parameters**

| Parameter | Description |
|---|---|
| **Type** | Multipoint or point-to-point |
| **Accessibility** | Freedom (or lack thereof) with which customers can join the call |
| **Monitoring** | Level of notification of customer joins and drops |
| **Modifiability** | Whether a nonowner customer may add connections |
| **Connection List** | Current connections in call |
| **Endpoint List** | Current endpoints in call (distributed) |

### A.3.1.1 Call Type

The call type designates whether the call is *multipoint* or restricted to *point-to-point*:

$$\textbf{Type} \in \{\textbf{MULTIPOINT}, \textbf{POINT-TO-POINT}\}$$

A multipoint call can contain connections that are either multipoint or point-to-point, whereas a point-to-point call can only contain point-to-point connections. While the call model supports multicast calls as the general object, some calls are intrinsically point-to-point and, if designated as such, can be serviced more efficiently by the network (for example, by routing the call over internal trunks). The call model provides for arbitrary multipoint configurations, including 1-to-N and M-to-N connectivity (where the more general M-to-N connectivity can be viewed as a supplementary service).

Point-to-point connections are clearly required to support point-to-point videotelephony and traditional services. One-to-N multipoint connections are required to support broadcast NTSC and HDTV video distribution services. While the connectivity provided by general M-to-N connections can be emulated via a collection of point-to-point and 1-to-N connections, some applications can be facilitated more easily using M-to-N connections. Additionally, M-to-N connections allow the bandwidth requirements of several different flows to be aggregated over a single connection (useful, for example, in LAN interconnect). Therefore, this call model does not preclude M-to-N connectivity.

### A.3.1.2 Call Accessibility

The accessibility parameter is a 1-tuple that describes how customers are allowed to join the call:

$$\textbf{Accessibility} \in \{\textbf{OPEN, CLOSED}, \textbf{CUSTOMER\_VERIFY}, \textbf{NETWORK\_VERIFY}\}$$

If the call is OPEN, any customer in the network may freely join the call without the owner's permission. A CLOSED call restricts the call such that only the owner may add customers. If the call has the CUSTOMER_VERIFY accessibility, then any customer may attempt to join the call, but the operation will be verified with the owner before it is allowed to complete. If the call has the NETWORK_VERIFY accessibility, then the network verifies the join for the owner by consulting a network maintained database.

The OPEN accessibility is useful for commercial broadcast video distribution, where any customer can attach to the video feed without restriction. The CLOSED accessibility is useful for private communications, such as point-to-point and conference calls, where all outside parties must be invited by the call owner. The CUSTOMER_VERIFY accessibility is useful for relatively small fanout broadcast video, where the service provider checks to see if the customer can access the service and maintains the billing statistics. This mode can support university course offerings and similar services. The NETWORK_VERIFY accessibility is useful for subscription video distribution services, such as cable TV and pay-for-view, where only subscribing customers can access the service. With pay-for-view, the network always maintains connection time statistics for use in billing.

#### A.3.1.3 Call Monitoring

The monitoring parameter is a 3-tuple designating whether endpoints of the call are notified when new endpoints join or drop out of the call:

$$\textbf{Monitor} = \langle\, \textbf{owner} \in \{\textbf{ON}, \textbf{OFF}\},\ \textbf{transmitters} \in \{\textbf{ON}, \textbf{OFF}\},\ \textbf{all} \in \{\textbf{ON}, \textbf{OFF}\} \,\rangle$$

If the *owner* field is set to ON, all endpoint joins and drops are reported to the owner. Likewise, if the *transmitters* field is set to ON, all transmitters are notified when these changes occur, and if the *all* field is set to ON, all participants in the call are notified.

Monitoring is desirable in dynamic multipoint calls, primarily so that transmitters can become aware of new receivers listening to their transmissions. Also, it allows the owner to keep track of client joins and drops for use in billing.

#### A.3.1.4 Call Modifiability

The modifiability parameter controls whether non-owners have the permission to add connections to the call:

$$\textbf{Modifiability} \in \{\textbf{ON}, \textbf{OFF}\}$$

If modifiability is ON and a non-owner customer adds a connection to the call, this customer becomes the owner of the new connection and may be billed for it.

A modifiability setting of ON is useful in multiparty conference calls, where each endpoint adds a one-to-all connection for its feed when joining the call. A customer can add several feeds corresponding, for example, to audio and video, and perhaps different video encoding schemes.

#### A.3.1.5 Call's Connection List and Endpoint List

The connection list contains the connections of the call and the endpoint list contains the endpoints of the call, as described in the following sections. Both lists are dynamic since connections and endpoints can be added to or removed from the call at any time, and since the attributes of both can change over time. The endpoint list is distributed to the access nodes and is not necessarily completely contained at any one location within the network.

### A.3.2 Connections

The connection is the primary information carrying component of a call. The four parameters associated with connections are described in the following subsections and summarized in Table 2. These parameters are *global* attributes applying to the entire connection and all endpoints in the connection.

**TABLE 2. Connection Parameters**

| Parameter | Description |
|---|---|
| **Type** | Three tuple consisting of VP/VC, dynamic/static bandwidth and quality of service |
| **Bandwidth** | The bandwidth specification for the connection. |
| **Rx/Tx Permission** | Default restrictions on the receive/transmit access of endpoints |

TABLE 2. Connection Parameters (Continued)

| Parameter | Description |
|---|---|
| **Endpoint Maps** | List of endpoint mappings into this connection for all endpoints in call (distributed) |

#### A.3.2.1 Connection Type

The connection type is a 3-tuple:

$$\textbf{ConnType} = < \textbf{channel\_type} \in \{\textbf{VP}, \textbf{VC}\},\ \textbf{BW\_type} \in \{\textbf{STATIC}, \textbf{DYNAMIC}\},\ \textbf{QOS} \in \{\textbf{for further study}\} >$$

The first option specifies the type of channel that the connection requires and must be one of two values: *virtual path* (VP) or *virtual channel* (VC). With a VP connection, clients set the VPI field of the ATM header when sending cells. The network then uses the VPI for routing, possibly remapping this field at every switching node within the network, until the cells reach their destinations. With VC connections, clients set the VCI and the VPI when sending cells and the network uses both the VCI and VPI for routing. For VP connections, the VCI is preserved by the network—whatever value the sending client places in this field is delivered to the destination client(s) and is available for use by the clients, for example, as a multiplexing field.

The second type option specifies whether the connection is DYNAMIC or STATIC. If a connection is STATIC then the bandwidth is fixed throughout the life of the call, which makes the connection more predictable and allows for more efficient use of network resources (such as routing over internal trunks). DYNAMIC connections can have their bandwidth modified during the life of the call.

The last type option specifies the desired *quality-of-service* (QOS). Possible values for this parameter are **for further study**.

#### A.3.2.2 Connection Bandwidth Specification

The precise specification of connection bandwidth is **for further study**.

#### A.3.2.3 Connection Default Rx/Tx Permissions

Connection permissions describe the allowed receive and transmit capability of endpoints on the connection. Each connection has a *default permission*, which can be overridden by the call owner so that individual endpoints can be assigned different permissions. The permission tuple consists of a receive part and a transmit part, as shown below:

$$\textbf{Perm} = <\textbf{Rcv} = < \textbf{perm} \in \{\textbf{ON}, \textbf{HOLD}, \textbf{OFF}\},\ \textbf{changeability} \in \{\textbf{ON}, \textbf{VERIFY}, \textbf{OFF}\}>,\ <\textbf{Trans} = < \textbf{perm} \in \{\textbf{ON}, \textbf{HOLD}, \textbf{OFF}\}\ \textbf{changeability} \in \{\textbf{ON}, \textbf{VERIFY}, \textbf{OFF}\}>>$$

For the receive part, a *perm* of ON indicates that the endpoint can receive on this connection, a *perm* of OFF indicates that the endpoint cannot receive on this connection, and a *perm* of HOLD indicates that the endpoint cannot receive, but that bandwidth should be reserved within the network up to the endpoint's access node (for use when the endpoint's receive permission is to be changed to ON at some later time). The *changeability* field indicates whether an endpoint can deviate from its assigned permissions when mapping the connection in, as described below in Section A.3.3.2. The transmit part is defined analogously to the receive part.

#### A.3.2.4 Connection Endpoint Mapping List

The endpoint mapping list contains all of the endpoint mappings for this connection. This list is dynamic since endpoints can be added to or removed from the call (and thus the connection) at any time, and since the attributes of endpoints can change over time. The endpoint mapping list is distributed to the access nodes and is not necessarily completely contained at any one location within the network

### A.3.3 Endpoint Mappings

The endpoint mapping describes how an endpoint accesses a particular connection. Each endpoint has a mapping object for all connections of the call. The four parameters associated with endpoint mappings are described in the following subsections and summarized in Table 3.

**TABLE 3. Endpoint Mapping Parameters**

| **Parameter** | **Description** |
|---|---|
| **Rx/Tx Permission** | Override restriction on the receive/transmit access for the associated endpoint |
| **Rx/Tx Mapping** | The current receive/transmit access mapping for the associated endpoint |
| **VPI/VCI** | The Virtual Path Identifiers and Virtual Channel Identifiers for this connection |
| **Notification** | Whether the associated endpoint should be notified of bandwidth changes |

#### A.3.3.1 Endpoint Override Rx/Tx Permissions

The default connection Rx/Tx permission, described in Section A.3.2.3, is a *global* connection parameter that applies to all endpoints in the connection, until overridden. Each endpoint also has an *override Rx/Tx permission* for each connection, only applicable to itself. The owner of the connection can change individual endpoint override permissions from that of the default value to give certain endpoints a restricted or expanded set of privileges. This can be done either when adding the endpoint to the connection or at some later time by modifying the endpoint's attributes. If no override is specified, the default connection permission is assigned to the endpoint.

#### A.3.3.2 Endpoint Rx/Tx Mappings

The endpoint Rx/Tx mapping is set by the endpoint and determines how the endpoint accesses the connection. The mapping tuple is defined as follows:

$$\begin{aligned}\textbf{Rx/Tx Mapping} = \langle\langle\, &\textbf{Rcv\_map} \in \{\textbf{ON}, \textbf{HOLD}, \textbf{OFF}\}\rangle,\\ &\langle\, \textbf{Trans\_map} \in \{\textbf{ON}, \textbf{ON\_W\_ECHO}, \textbf{HOLD}, \textbf{OFF}\}\rangle\rangle\end{aligned}$$

A mapping of ON indicates that the endpoint can receive or transmit on this connection and a mapping of OFF or HOLD indicates that the endpoint cannot receive or transmit (where HOLD reserves network resources and OFF does not). The ON_W_ECHO transmit mapping indicates that the transmitter is to receive a copy of all transmissions.

The endpoint Rx/Tx mapping is constrained by the Rx/Tx permission (Sections A.3.2.3 and A.3.3.1). If the permission *changeability* field is set to OFF for the receive (transmit) part, the endpoint's receive (transmit) mapping must equal its permission *perm*. If *changeability* is ON, then the endpoint may choose any mapping and change its mapping during the life of the connection. In this case, the permission is not enforced and is merely viewed as a suggestion to the endpoint. If the *changeability* is VERIFY, the endpoint can alter its mapping, but the owner will first be queried to see if the new mapping is acceptable. There is one exception to this general rule: if a receive or transmit permission *perm* is ON, the endpoint can freely change its mapping to HOLD regardless of the *changeability* setting.

The combination of connection/endpoint permissions and mappings allows the service provider to designate service components (connections) as mandatory or optional. For example, if a televideo call has a mandatory audio component and an optional video component, the permissions on the audio component could be set to Rx/Tx both ON with a changeability of OFF and the permissions on the video component could be set to Rx/Tx both ON with a changeability of ON (giving the customer the capability of turning the video component on or off).

#### A.3.3.3 Endpoint VPI/VCI

In order to facilitate communication at the network (hardware) level, each direction of a connection must have a VPI/VCI pair that describes the VPI/VCI values to be placed in the corresponding ATM cells. These values are contained in the VPI/VCI endpoint mapping attribute, as shown below:

$$\begin{aligned}\textbf{VPI/VCI} = \langle\textbf{Rcv} = \langle\, &\textbf{VPI} \in \{\textbf{0} \ldots \textbf{VPI\_LIMIT}, \textbf{UNUSED}\},\\ &\textbf{VCI} \in \{\textbf{0} \ldots \textbf{VCI\_LIMIT}, \textbf{UNUSED}\}\rangle,\\ \langle\textbf{Trans} = \langle\, &\textbf{VPI} \in \{\textbf{0} \ldots \textbf{VPI\_LIMIT}, \textbf{UNUSED}\},\\ &\textbf{VCI} \in \{\textbf{0} \ldots \textbf{VCI\_LIMIT}, \textbf{UNUSED}\}\rangle\rangle\end{aligned}$$

The receive (Rcv) value contains the VPI/VCI pair for connections flowing from the network to the customer, while the transmit (Trans) value contains the pair for the customer-to-network direction. The allowed values for VPI and VCI must be within the defined range (0 to some limit), or can be a distinguished value that indicates the VPI and VCI are unused (for unidirectional or NULL connections that do not flow in one or both directions).

The call model presented here supports both *uni-directional* and *bi-directional* assignment of VPIs and VCIs at the UNI. Uni-directional assignment allows the VPI and VCI to differ for each direction of a *receive/transmit* connection. Bi-directional assignment requires the VPI/VCI values to be the same for both directions.

#### A.3.3.4 Endpoint Notification

The notification flag is set by each endpoint when joining a connection and has one of two values:

$$\textbf{Notification} \in \{\textbf{ON}, \textbf{OFF}\}$$

When notification is ON, the endpoint will be notified of all changes in bandwidth that occur on the connection. If OFF, the endpoint is not notified. This parameter is useful when a endpoint wishes to reconfigure an interface (for example, by adding more buffers) when a bandwidth change occurs.

## A.4 Summary

This appendix has described a call model and object decomposition wherein calls are represented by *call, connection, endpoint* and *endpoint mapping* objects. The call, connection and endpoint attributes have been chosen to allow dynamic modification of the call. For example, call accessibility governs how customers can join the call while it is in progress, and connection/endpoint permissions and mappings govern how individual endpoints are allowed to change their access method for each connection. Additionally, the model allows the modification of the call, connection and endpoint attributes themselves during the life of the call so that, for example, the bandwidth or quality of service can be changed dynamically.

## Acknowledgments

Work on the call model described in this contribution was supported by Southwestern Bell Telephone Company, Southwestern Bell Technology Resources and NEC America.

### References

[1] ANSI T1S1 Technical Sub-Committee. *B-ISDN Services Baseline Document*. T1S1.5/92-006R1, March 1992. Editors: H.M. Ruck and D.J. Kostas.

[2] ANSI T1S1 Technical Sub-Committee. *A Call Model For Long-Term Signalling*. T1S1.5/92-017, January 1992. Authors: H.E. Bussey and S.E. Minzer.

[3] Bell Communications Research. *Broadband ISDN Access Signaling Framework Generic Criteria for Class II Equipment*. Framework Technical Advisory FA-NWT-001111, Issue 1, December 1991.

[4] CCITT, Study Group XI/Working Party XI/4-3. *Baseline Text For The Harmonized Signalling Requirements*. Document XI/4–37, March 1992.

[5] CCITT, Study Group XI/Working Party XI/6. *Draft Text for Q.93B*. Report TD XI/6–37 Revision 1, March 1992.

[6] J.D. DeHart, M.E. Gaddis and R.G. Bubenik. *Connection Management Access Protocol (CMAP) Specification.* Washington University, Department of Computer Science, Technical Report WUCS-92-01, Version 2.1. February 1992.