Definitions of Tests for ATM Management

September 1, 1998

Michael Noto (editor)
Network Equipment Technologies
mike_noto@net.com

Kaj Tesink (editor)
Bell Communications Research
kaj@cc.bellcore.com

1. Status of this Memo

This document is an Internet-Draft. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as 'work in progress.'

To learn the current status of any Internet-Draft, please check the `iid-abstracts.txt` listing contained in the Internet-Drafts Shadow Directories on ftp.ietf.org (US East Coast), nic.nordu.net (Europe), ftp.isi.edu (US West Coast), or munnari.oz.au (Pacific Rim).

Copyright Notice

Copyright (C) The Internet Society (1998). All Rights Reserved.

2. Abstract
This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it describes objects used for managing ATM-based interfaces, devices, networks and services in addition to those defined in the ATM MIB [16], to provide support for the management of ATM Loopback Tests.

3. The SNMP Network Management Framework

The SNMP Management Framework presently consists of five major components:

0 An overall architecture, described in RFC 2271 [1].

0 Mechanisms for describing and naming objects and events for the purpose of management. The first version of this Structure of Management Information (SMI) is called SMIv1 and described in RFC 1155 [2], RFC 1212 [3] and RFC 1215 [4]. The second version, called SMIv2, is described in RFC 1902 [5], RFC 1903 [6] and RFC 1904 [7].

0 Message protocols for transferring management information. The first version of the SNMP message protocol is called SNMPv1 and described in RFC 1157 [8]. A second version of the SNMP message protocol, which is not an Internet standards track protocol, is called SNMPv2c and described in RFC 1901 [9] and RFC 1906 [10]. The third version of the message protocol is called SNMPv3 and described in RFC 1906 [10], RFC 2272 [11] and RFC 2274 [12].

0 Protocol operations for accessing management information. The first set of protocol operations and associated PDU formats is described in RFC 1157 [8]. A second set of protocol operations and associated PDU formats is described in RFC 1905 [13].

0 A set of fundamental applications described in RFC 2273 [14] and the view-based access control mechanism described in RFC 2275 [15].
Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the mechanisms defined in the SMI.

This memo specifies a MIB module that is compliant to the SMIV2. A MIB conforming to the SMIV1 can be produced through the appropriate translations. The resulting translated MIB must be semantically equivalent, except where objects or events are omitted because no translation is possible (e.g., use of Counter64). Some machine readable information in SMIV2 will be converted into textual descriptions in SMIV1 during the translation process. However, this loss of machine readable information is not considered to change the semantics of the MIB.
4. Overview

The purpose of this memo is to provide additional capabilities, not found in the ATM MIB [16], which are needed to manage ATM interfaces. This memo addresses ATM Testing Support and must be used in conjunction with the System/Interface Test MIB [22].

4.1. Background

In addition to the MIB module defined in this memo, other MIB modules are necessary to manage ATM interfaces, links and cross-connects. Examples include MIB II for general system [17] and interface management [18], the DS3 or SONET MIBs for management of SONET and DS3 physical interfaces, and, as appropriate, MIB modules for applications that make use of ATM, such as SMDS and LAN Emulation. These MIB modules are outside the scope of this specification.

This MIB module requires the use of the ATM MIB module defined in [16] and the System/Interface Test MIB module [22].

This memo defines extensions to the ATM MIB in order to support ATM Loopback Tests. An ATM Loopback Test provides the ability to send out a loopback OAM (Operations and Maintenance) cell to verify the existence of connectivity for a particular connection.
4.2. Terminology

The following terms are defined here and used throughout this MIB:

- Virtual Path Link (VPL)
- Virtual Path Connection (VPC)
- Virtual Path Segment (VP Segment)
- Virtual Channel Link (VCL)
- Virtual Channel Connection (VCC)
- Virtual Channel Segment (VC Segment).

Figure 1: Examples of Virtual Path Links, Virtual Path Connection, and Virtual Path Segment
Figure 2: Examples of Virtual Channel Links, Virtual Channel Connection, and Virtual Channel Segment

4.3. Supported Functions

The managed ATM objects are organized as follows:

1. ATM Loopback Testing
2. ATM End-Point Tables

4.3.1. ATM Loopback Testing

The loopback test provides the ability to send out a loopback OAM cell to verify the existence of connectivity for a particular connection. Loopback tests can be performed on either an entire connection (i.e., an end-to-end test), a segment of the connection (i.e., a segment test), a portion of a segment (i.e., a loopback location identifier test), or the network portion of a connection (i.e., a service internal test).

The loopback test makes use of the Test Table defined in [22]. For a given interface, a loopback test can be invoked by obtaining ownership of a test and then by setting the value of testType equal to one of the ATM Loopback Test Types defined in Section 5. See procedures in [22] for using the Test Table.
After invoking a loopback test, the object testResult can be read to determine the outcome of the loopback test (e.g., ‘success(2)’ if the loopback cell made it back to the originator of the test or ‘failed(7)’ if the loopback cell did not make it back).

The following types of loopback tests are defined:

- End-to-end Loopback Test
- Segment Loopback Test
- Loopback Test Using Loopback Location Identifier
- Network Loopback Test.

1) End-to-end Loopback Test

The end-to-end loopback (LB) is self-explanatory. For a VP test, the cell is sent on the given VP, via VCI=4 specified in [20]. For a VC test, the LB cell is sent on the VC under test, with the PTI (Payload Type Indicator) set to 5 as specified in [20]. Figure 3 illustrates the end-to-end loopback test.

![Figure 3: End-to-end Loopback Test](image)

2) Segment Loopback Test

The segment LB test is explained in ITU-T I.610[21]. For a VP segment test, the LB cell is sent on the VP under test via VCI=3 as specified in [20], and the Loopback Location ID field is set to all 1’s. For a VC segment test, the LB cell is sent on the VC under test, with the PTI set to 4 as specified in [21].

See Section 5 for more details on how to use the End-to-end Loopback Test.
[20], and the Loopback Location ID field is set to all 1’s.

This test involves a LB cell being inserted at a pre-defined segment end-point, and looped back at the corresponding segment end-point encountered. The pair of segment end-points define a segment (which is used for the segment loopback test). A VP/VC connection can have multiple segments, but multiple segments cannot overlap.

A UNI interface is by definition defined as a segment end-point (hence a UNI would be considered a segment). A segment can also define:
- a B-ICI
- a public carrier’s ’piece’ of the connection
- a private network’s ‘piece’ of the connection.

In order to support this functionality, the VP/VC link termination needs to be able to be defined as a segment. This can be done using either the atmVplSegmentEndPoint or atmVclSegmentEndPoint object depending on whether it is for a VPC or VCC. A segment loopback test is illustrated in Figure 4.

```
          _______    _______    _______    _______    ____
|Host|    |       |    |       |    |       |    |Host|
|____|----|SwitchA|----|SwitchB|----|SwitchC|----|____|
|_______|    |_______|    |_______|
|<---------------------->|  Segment
```

Figure 4: Segment Loopback Test

Section 5 describes the use of the ATM Segment Loopback Tests.

3) Loopback Test Using Loopback Location Identifier

This loopback test is a special type of 2) where the Loopback Location ID field is set to a value that corresponds to a specific node in a given network (Note that the format of this field is not standardized, that is, the value is significant only within an administrative domain). In this case, the device initiating the LB test inserts the appropriate Loop Back Location ID. When the LB cell reaches the corresponding
device, that device recognizes the Loopback Location ID as its own, and loops it back. This test is useful for performing fault sectionalization without having to provision segment end-points. An additional object, the atmLoopbackID, is defined to determine the loopback point. Figure 5 shows a loopback test using a location identifier. Note that the loopback test using location identifier can be used to perform a loopback test over a portion of a defined segment. See Figure 5.

```
|               | ____________      | ____________      | ____________      | ____________      | ____________      |
| Host          | | SwitchA |       |       | SwitchB |       |       | SwitchC |       | Host      |
```

|<------------------------> Portion of Segment that Loopback test is performed on

|<--------------------------->| Segment

Figure 5: Loopback Test Using Location Identifier

See Section 5 for more details.

4) Network Loopback Test

This is a loopback test that the manager requests an agent in a network to perform over the internal portion of a designated connection. The Network then initiates the internal network loopback test by inserting an OAM loopback cell at one of the end-points of the internal network portion of the connection. When the loopback cell reaches the other end-point of the internal Network, the cell is looped back. This test is useful for verifying connectivity through a particular network. Figure 6 illustrates the Network loopback test.
There are two ATM End-point tables: the ATM VP End-point Table and the ATM VC End-point Table. The ATM VP End-point Table augments the atmVplTable and defines the atmVplEndptSegmentEndPoint object to represent whether or not a specified VPL is a segment end-point. Similarly for Virtual Channels, the ATM VC End-point Table and the atmVclEndptSegmentEndPoint object are used to represent whether or not a specified VCL is a segment end-point.
5. Definitions

ATMTEST-MIB DEFINITIONS ::= BEGIN

IMPORTS
  MODULE-IDENTITY, OBJECT-IDENTITY,
  OBJECT-TYPE, mib-2, experimental
  FROM SNMPv2-SMI
  MODULE-COMPLIANCE, OBJECT-GROUP
  FROM SNMPv2-CONF
  atmVplEntry, atmVclEntry
  FROM ATM-MIB;

atmTESTMIB MODULE-IDENTITY
  LAST-UPDATED "9809011200Z"
  ORGANIZATION "IETF AToMMIB Working Group"
  CONTACT-INFO
    Michael Noto
    Postal: Network Equipment Technologies
    800 Saginaw Drive RM 21.1.111
    Redwood City, CA 94063
    Tel: +1 415 569-7134
    E-mail: noto@net.com

    Kaj Tesink
    Postal: Bell Communications Research
    331 Newman Springs Road
    Red Bank, NJ 07701
    US
    Tel: +1 732 758 5254
    Fax: +1 732 758 4177
    E-mail: kaj@cc.bellcore.com"

DESCRIPTION
  "This MIB Module provides
   ATM Loopback Tests and supporting objects
   that must be supported by ATM devices
   providing ATM Loopback Tests."
  ::= { experimental XX }

-- ******** NOTE TO THE RFC EDITOR / IANA ************
-- Before this module is put on the standards track
-- * replace { experimental XX } with { mib-2 YY }
-- * remove experimental from the IMPORTS clause
-- * assign YY by IANA.
atmTESTMIBObjects OBJECT IDENTIFIER ::= {atmTESTMIB 1}

-- This ATMTEST-MIB Module consists of the following:
--   (1) ATM Loopback Testing
--   (2) ATM End-Point Tables
-- ************************************************
-- (1) ATM Loopback Testing
-- This concerns information for interfaces
-- supporting ATM Loopback Tests and includes:
-- 1. ATM Loopback Objects
-- 2. List of ATM Loopback Test Types

atmLoopbackTestGroup OBJECT IDENTIFIER ::= {
atmTESTMIBObjects 1
}

-- 1. ATM Loopback Objects
-- The following objects are defined for use in
-- performing ATM Loopback Tests.

atmLoopbackID OBJECT-TYPE
SYNTAX OCTET STRING(SIZE(16))
MAX-ACCESS read-write
STATUS current
DESCRIPTION "This identifier is used to identify this local
ATM device. The value of this object can be used
by other ATM devices to identify this local ATM
device as the device that is being requested to
loopback the OAM Loopback cell. The default for
this field is all 1’s, which would indicate a
segment OAM Loopback Test. Location Identifiers of
less than 16 octets are left justified, and padded
with all 0’s."
DEFVAL { 'ffffffffffffffffffffffffffffffff'H }
 ::= { atmLoopbackTestGroup 1 }

-- 2. List of ATM Loopback Test Types
-- The following loopback test types are defined:
-- atmLoopbackVpE2e
-- atmLoopbackVcE2e
-- atmLoopbackVpSegment
-- atmLoopbackVcSegment
-- atmLoopbackVpLocationID
-- atmLoopbackVcLocationID
-- atmLoopbackVpServiceInternal
-- atmLoopbackVcServiceInternal
atmLoopbackTestTypes OBJECT IDENTIFIER ::= {
    atmLoopbackTestGroup 4 }

atmLoopbackVpE2e OBJECT-IDENTITY
STATUS       current
DESCRIPTION
  "This is an end-to-end loopback test performed on a
  designated VP (Virtual Path). To perform this test
  an end-to-end loopback OAM cell is inserted at one
  of the end-points of the designated VP connection
  (e.g., at a host) via VCI=4 (the VCI value for VP
  OAM end-to-end cells), travels to the other end-
  point of the VP connection, and then loops back to
  the originating end-point on the designated VP.
  Success is achieved if the loopback OAM cell
  returns to the originating end-point within 5
  seconds, otherwise, the test fails.

  The manager-station performs a loopback test by
  making use of the testTable defined in [22]. In
  order to run this test the object testType in the
  testTable shall be set to atmLoopbackVpE2e, and the
  object testTarget points to the row in the
  atmVplTable in [16] corresponding to the VP
  designated for the test.

  Before starting a test, a manager-station must
  first obtain 'ownership' of the entry in the
  testTable for the interface to be tested (follow
  procedure defined in [22]). Once the manager-
  station obtains ownership, a loopback test for a
  given interface can be invoked by first setting up
  parameters necessary for the loopback test (e.g.,
  set the testTarget), and then setting the value of
  testType in the testTable equal to
  'atmLoopbackVpE2e'. The testRowStatus is used to
  invoke the atmLoopbackVpE2e test on the VP with the
  VPI corresponding to the testTarget.

  After invoking a loopback test, wait for the test
  completion by polling for the object testResult. A
  value of 'inProgress(3)' will result if the test is
  still in progress. Once the test is completed, the
  object testResult will have a value of 'success(2)'"
if the loopback OAM cell returned to the originator of the test within 5 seconds, if not, a value of ‘failed(7)’ will result. If the ATM system does not support this type of loopback test, then a value of ‘notSupported(4)’ will be provided. Other possible values for the testResult object are ‘unableToRun(5)’ and ‘aborted(6)’.

::= { atmLoopbackTestTypes 1 }

atmLoopbackVcE2e OBJECT-IDENTITY
  STATUS current
  DESCRIPTION
    "This is an end-to-end loopback test performed on a designated VC (Virtual Channel). To perform this test an end-to-end loopback OAM cell is inserted at one of the end-points of the designated VC connection (e.g., at a host) via PTI=5 (the PTI value used for VC OAM end-to-end cells), travels to the other end-point of the VC connection, and then loops back to the originating end-point on the designated VC. Success is achieved if the loopback OAM cell returns to the originating end-point within 5 seconds, otherwise, the test fails.

    The manager-station performs a loopback test by making use of the testTable defined in [22]. In order to run this test the object testType in the testTable shall be set to atmLoopbackVcE2e, and the object testTarget points to the row in the atmVclTable in [16] corresponding to the VC designated for the test.

    Before starting a test, a manager-station must first obtain ‘ownership’ of the entry in the testTable for the interface to be tested (follow procedure defined in [22]). Once the manager-station obtains ownership, a loopback test for a given interface can be invoked by first setting up parameters necessary for the loopback test (e.g., set the testTarget), and then setting the value of testType in the testTable equal to ‘atmLoopbackVcE2e’. The testRowStatus is used to invoke the atmLoopbackVcE2e test on the VC with the VPI/VCI corresponding to the testTarget."
After invoking a loopback test, wait for the test completion by polling for the object testResult. A value of ‘inProgress(3)’ will result if the test is still in progress. Once the test is completed, the object testResult will have a value of ‘success(2)’ if the loopback OAM cell returned to the originator of the test within 5 seconds, if not, a value of ‘failed(7)’ will result. If the ATM system does not support this type of loopback test, then a value of ‘notSupported(4)’ will be provided. Other possible values for the testResult object are ‘unableToRun(5)’ and ‘aborted(6)’.

```plaintext
::= { atmLoopbackTestTypes 2 }
```

```

atmLoopbackVpSegment OBJECT-IDENTITY
  STATUS current
  DESCRIPTION
    "This is a loopback test performed on a designated segment of a VP (Virtual Path). To perform this test a segment OAM cell is inserted at one of the segment end-points of the designated VP connection (e.g., at a host) via VCI=3 (the VCI used for VP OAM segment cells), travels across the segment on the designated VP to the device pre-configured as the corresponding segment end-point, and then loops back to the originating segment end-point on the designated VP. Success is achieved if the loopback OAM cell returns to the originating end-point within 5 seconds, otherwise, the test fails.

In order to use the atmLoopbackVpSegment test, a segment must be defined by setting up segment end-points using the atmVplEndptSegmentEndPoint object from the atmVplEndptTable. The atmVplEndptSegmentEndPoint is set to ‘isaVpSegmentEndPoint(1)’ for each segment end-point. Note that this object is by default set to ‘isaVpSegmentEndPoint(1)’ if the atmVplTable supports one end of a UNI. In such a case, a UNI VP loopback test would be achieved when the atmLoopbackVpSegment test was initiated over the UNI.

The manager-station performs a loopback test by making use of the testTable defined in [22]. In
```
order to run this test the object testType in the testTable shall be set to atmLoopbackVpE2e, and the object testTarget points to the row in the atmVplTable in [16] corresponding to the VP designated for the test.

Before starting a test, a manager-station must first obtain 'ownership' of the entry in the testTable for the interface to be tested (follow procedure defined in [22]). Once the manager-station obtains ownership, a loopback test for a given interface can be invoked by first setting up parameters necessary for the loopback test (e.g., set the testTarget), and then setting the value of testType in the testTable equal to 'atmLoopbackVpSegment'. The testRowStatus is used to invoke the atmLoopbackVpSegment test on the VP with the VPI corresponding to the testTarget.

After invoking a loopback test, wait for the test completion by polling for the object testResult. A value of ‘inProgress(3)’ will result if the test is still in progress. Once the test is completed, the object testResult will have a value of ‘success(2)’ if the loopback OAM cell returned to the originator of the test within 5 seconds, if not, a value of ‘failed(7)’ will result. If the ATM system does not support this type of loopback test, then a value of ‘notSupported(4)’ will be provided. Other possible values for the testResult object are ‘unAbleToRun(5)’ and ‘aborted(6)’.

::= { atmLoopbackTestTypes 3 }

atmLoopbackVcSegment OBJECT-IDENTITY
  STATUS current
  DESCRIPTION
  "This is a loopback test performed on a designated segment of a VC (Virtual Channel). To perform this test a segment OAM cell is inserted at one of the segment end-points of the designated VC connection (e.g., at a host) via PTI=4 (the PTI value used for VC OAM segment cells), travels across the segment on the designated VC to the device pre-configured as the corresponding segment end-point, and then loops back to the originating segment end-point on the designated VC connection."
VC. Success is achieved if the loopback OAM cell returns to the originating end-point within 5 seconds, otherwise, the test fails.

In order to use the atmLoopbackVcSegment test, a segment must be defined by setting up segment end-points using the atmVciEndptSegmentEndpoint object from the atmVciEndptTable. The atmVciEndptSegmentEndpoint is set to ‘isaVcSegmentEndpoint(1)’ for each segment end-point. Note that this object is by default set to ‘isaVcSegmentEndpoint(1)’ if the atmVciTable supports one end of a UNI. In such a case, a UNI VC loopback test would be achieved when the atmLoopbackVcSegment test was initiated over the UNI.

The manager-station performs a loopback test by making use of the testTable defined in [22]. In order to run this test the object testType in the testTable shall be set to atmLoopbackVcE2e, and the object testTarget points to the row in the atmVciTable in [16] corresponding to the VC designated for the test.

Before starting a test, a manager-station must first obtain ‘ownership’ of the entry in the testTable for the interface to be tested (follow procedure defined in [22]). Once the manager-station obtains ownership, a loopback test for a given interface can be invoked by first setting up parameters necessary for the loopback test (e.g., set the testTarget), and then setting the value of testType in the testTable equal to ‘atmLoopbackVcSegment’. The testRowStatus is used to invoke the atmLoopbackVcSegment test on the VC with the VPI/VCI corresponding to the testTarget.

After invoking a loopback test, wait for the test completion by polling for the object testResult. A value of ‘inProgress(3)’ will result if the test is still in progress. Once the test is completed, the object testResult will have a value of ‘success(2)’ if the loopback OAM cell returned to the originator of the test within 5 seconds, if not, a value of ‘failed(7)’ will result. If the ATM system does not support this type of loopback test, then a value of ‘notSupported(4)’ will be provided. Other possible
values for the testResult object are ‘unAbleToRun(5)’
and ‘aborted(6)’."
 ::= { atmLoopbackTestTypes 4 }

atmLoopbackVpLocationId OBJECT-IDENTITY
STATUS current
DESCRIPTION
"This is a loopback test performed on a portion of a
designated VP segment. To perform this test a loopback
OAM cell is inserted at a connection point of the
designated VP connection (e.g., the end-point or a
tandem point) with a value inserted in the Location
Identifier ID field of the OAM cell that corresponds to
the ATM device where the cell is to be looped back.
The loopback cell then travels through the VP
connection until it reaches the designated ATM device,
where it is looped back to the loopback cell insertion
point on the designated VP. Success is achieved if the
loopback OAM cell returns to the originating point of
insertion within 5 seconds, otherwise, the test fails.

The manager-station performs a loopback test by making
use of the testTable defined in [22]. In order to run
this test the object testType in the testTable shall be
set to atmLoopbackVpE2e. The object testTarget points
to the row in the atmVplTable in [16] corresponding to
the VP designated for the test. The object testMoreInfo
contains the desired Loopback Location ID.

Before starting a test, a manager-station must first
obtain ‘ownership’ of the entry in the testTable for
the interface to be tested (follow procedure defined in
[22]). Once the manager-station obtains ownership, a
loopback test for a given interface can be invoked by
first setting up parameters necessary for the loopback
test (e.g., set testMoreInfo to AAAABBBBCCCCDDDD and
set the testTarget), and then setting the value of
testType in the testTable equal to
‘atmLoopbackVpSegment’. The testRowStatus is used to
invoke the atmLoopbackVpLocationId on the VP with the
VPI corresponding to the testTarget and looped back at
loopback location ID= AAAABBBBCCCCDDDD.

After invoking a loopback test, wait for the test
completion by polling for the object testResult. A value of ‘inProgress(3)’ will result if the test is still in progress. Once the test is completed, the object testResult will have a value of ‘success(2)’ if the loopback OAM cell returned to the originator of the test within 5 seconds, if not, a value of ‘failed(7)’ will result. If the ATM system does not support this type of loopback test, then a value of ‘notSupported(4)’ will be provided. Other possible values for the testResult object are ‘unAbleToRun(5)’ and ‘aborted(6)’.

::= { atmLoopbackTestTypes 5 }

atmLoopbackVcLocationId  OBJECT-IDENTITY
STATUS current
DESCRIPTION
"This is a loopback test performed on a portion of a designated Vc segment. To perform this test a loopback OAM cell is inserted at a connection point of the designated VC connection (e.g., the end-point or a tandem point) with a value inserted in the Location Identifier ID field of the OAM cell that corresponds to the ATM device where the cell is to be looped back. The loopback cell then travels through the VC connection until it reaches the designated ATM device, where it is looped back to the loopback cell insertion point on the designated VC. Success is achieved if the loopback OAM cell returns to the originating point of insertion within 5 seconds, otherwise, the test fails.

The manager-station performs a loopback test by making use of the testTable defined in [22]. In order to run this test the object testType in the testTable shall be set to atmLoopbackVcE2e. The object testTarget points to the row in the atmVclTable in [16] corresponding to the VC designated for the test. The object testMoreInfo contains the desired Loopback Location ID.

Before starting a test, a manager-station must first obtain ‘ownership’ of the entry in the testTable for the interface to be tested (follow procedure defined in [22]). Once the manager-station obtains ownership, a loopback test for a given interface can be invoked by first setting up parameters necessary for the loopback
test (e.g., set testMoreInfo to AAAABBBBCCCCDDDD and set the testTarget), and then setting the value of testType in the testTable equal to 'atmLoopbackVcSegment.d’. The testRowStatus is used to invoke the atmLoopbackVcLocationId test on the VC with the VPI/VCI corresponding to the testTarget and looped back at loopback location ID= AAAABBBBCCCCDDDD.

After invoking a loopback test, wait for the test completion by polling for the object testResult. A value of ‘inProgress(3)’ will result if the test is still in progress. Once the test is completed, the object testResult will have a value of ‘success(2)’ if the loopback OAM cell returned to the originator of the test within 5 seconds, if not, a value of ‘failed(7)’ will result. If the ATM system does not support this type of loopback test, then a value of ‘notSupported(4)’ will be provided. Other possible values for the testResult object are ‘unAbleToRun(5)’ and ‘aborted(6)’.

::= { atmLoopbackTestTypes 6 }

atmLoopbackVpServiceInternal OBJECT-IDENTITY
STATUS current
DESCRIPTION "This is a loopback test that the manager requests an agent to perform over the managed resource’s internal portion of a designated VP (i.e., between the ingress and egress interfaces of the VP connection). The agent is provided with the Ingress VPI, Egress Interface, and Egress VPI in order to run this internal test. This test may be useful in proxy situations where the proxy agent represents a network. Implementations of this test may be specific to the managed resource. One implementation in a managed network may be as follows, the managed network inserts a segment loopback OAM cell at the network internal segment end-point (corresponding to the ingress connection point) for the designated VP connection. The loopback cell then travels through the network’s portion of the VP connection until it reaches the networks connection point to the egress, where it is looped back to the network’s cell insertion point on the designated VP. Success is achieved if the loopback OAM cell returns to the originating internal network segment end-point"
within 5 seconds, otherwise, the test fails.

The manager-station performs a loopback test by making use of the testTable defined in [22]. In order to run this test the object testType in the testTable shall be set to atmLoopbackVpServiceInternal, and the object testTarget points to the row in the atmVpCrossConnectTable in [16] corresponding to the VP designated for the test.

Before starting a test, a manager-station must first obtain ‘ownership’ of the entry in the testTable for the interface to be tested (follow procedure defined in [22]). Once the manager-station obtains ownership, a loopback test for a given interface can be invoked by first setting up parameters necessary for the loopback test (e.g., set the testTarget), and then setting the value of testType in the testTable equal to ‘atmLoopbackVpServiceInternal’. The testRowStatus is used to invoke the atmLoopbackVpServiceInternal test on the VP crossconnect with the ingress and egress VPI values corresponding to the testTarget.

After invoking a loopback test, wait for the test completion by polling for the object testResult. A value of ‘inProgress(3)’ will result if the test is still in progress. Once the test is completed, the object testResult will have a value of ‘success(2)’ if the loopback OAM cell returned to the originator of the test within 5 seconds, if not, a value of ‘failed(7)’ will result. If the ATM system does not support this type of loopback test, then a value of ‘notSupported(4)’ will be provided. Other possible values for the testResult object are ‘unableToRun(5)’ and ‘aborted(6)’.

::= { atmLoopbackTestTypes 7 }

atmLoopbackVcServiceInternal OBJECT-IDENTITY
STATUS current
DESCRIPTION
"This is a loopback test that the manager requests an agent to perform over the managed resource’s internal portion of a designated VC (i.e., between the ingress and egress interfaces of the VC connection). The agent is provided with the Ingress VPI, Ingress VCI, Egress
Interface, Egress VPI, and Egress VCI in order to run this internal test. This test may be useful in proxy situations where the proxy agent represents a network. Implementations of this test may be specific to the managed resource. One implementation in a managed network may be as follows, the managed network inserts a segment loopback OAM cell at the network internal segment end-point (corresponding to the ingress connection point) for the designated VC connection. The loopback cell then travels through the network’s portion of the VC connection until it reaches the network’s connection point to the egress, where it is looped back to the network’s cell insertion point on the designated VC. Success is achieved if the loopback OAM cell returns to the originating internal network segment end-point within 5 seconds, otherwise, the test fails.

The manager-station performs a loopback test by making use of the testTable defined in [22]. In order to run this test the object testType in the testTable shall be set to atmLoopbackVcServiceInternal, and the object testTarget points to the row in the atmVcCrossConnectTable in [16] corresponding to the VC designated for the test.

Before starting a test, a manager-station must first obtain ‘ownership’ of the entry in the testTable for the interface to be tested (follow procedure defined in [22]). Once the manager-station obtains ownership, a loopback test for a given interface can be invoked by first setting up parameters necessary for the loopback test (e.g., set the testTarget), and then setting the value of testType in the testTable equal to ‘atmLoopbackVcServiceInternal’. The testRowStatus is used to invoke the atmLoopbackVcServiceInternal test on the VC crossconnect with the ingress and egress VPI/VCI values corresponding to the testTarget.

After invoking a loopback test, wait for the test completion by polling for the object testResult. A value of ‘inProgress(3)’ will result if the test is still in progress. Once the test is completed, the object testResult will have a value of ‘success(2)’ if the loopback OAM cell returned to the originator of the
test within 5 seconds, if not, a value of ‘failed(7)’ will result. If the ATM system does not support this type of loopback test, then a value of ‘notSupported(4)’ will be provided. Other possible values for the testResult object are ‘unAbleToRun(5)’ and ‘aborted(6)’.

::= { atmLoopbackTestTypes 8 }
-- **************
-- (2) ATM End-Point Tables

-- This concerns information for interfaces
-- supporting ATM Loopback Tests and includes:
-- 1. ATM VP End-Point Table
-- 2. ATM VC End-Point Table

atmEndptGroup OBJECT IDENTIFIER ::= {
atmTESTMIBObjects 2
}

-- 1. ATM VP End-Point Table

atmVplEndptTable OBJECT-TYPE
SYNTAX SEQUENCE OF AtmVplEndptEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION "End-point Information for each VP."
::= { atmEndptGroup 1 }

AtmVplEndptEntry ::= SEQUENCE {
atmVplEndptSegmentEndPoint INTEGER
}

atmVplEndptSegmentEndPoint OBJECT-TYPE
SYNTAX INTEGER {
  isaVplSegmentEndPoint(1),
  notaVplSegmentEndPoint(2)
}
MAX-ACCESS  read-create
STATUS  current
DESCRIPTION
"An indication of whether or not the VP interface has been configured to represent a VPC Segment End-Point. If the corresponding VP Link is a UNI, the value of this object is permanently set to isaVplSegmentEndPoint(1). Otherwise, the default is set to notaVplSegmentEndPoint(2)."
 ::= { atmVplEndptEntry 1 }

-- 2. ATM VC End-Point Table

atmVclEndptTable OBJECT-TYPE
SYNTAX            SEQUENCE OF AtmVclEndptEntry
MAX-ACCESS       not-accessible
STATUS           current
DESCRIPTION
"End-point Information for each VC."
 ::= { atmEndptGroup 2 }

atmVclEndptEntry OBJECT-TYPE
SYNTAX  AtmVclEndptEntry
MAX-ACCESS  not-accessible
STATUS  current
DESCRIPTION
"An entry with end-point information about the ATM VC."
AUGMENTS { atmVclEntry }
 ::= { atmVclEndptTable 1 }

AtmVclEndptEntry ::= SEQUENCE {
admVclEndptSegmentEndPoint  INTEGER
}

atmVclEndptSegmentEndPoint OBJECT-TYPE
SYNTAX  INTEGER {
  isaVclSegmentEndPoint(1),
  notaVclSegmentEndPoint(2)
}
MAX-ACCESS  read-create
STATUS  current
DESCRIPTION
"An indication of whether or not the VC interface has been configured to represent a VCC Segment End-Point. If the corresponding VC Link is a UNI, the value of this object is permanently set to isaVclSegmentEndPoint(1). Otherwise, the default is set to notaVclSegmentEndPoint(2)."

::= { atmVclEndptEntry 1 }

-- ********************************************************
-- Conformance Information
atmTESTMIBConformance  OBJECT IDENTIFIER ::= {atmTESTMIB 2}
atmTESTMIBGroups       OBJECT IDENTIFIER ::= {atmTESTMIBConformance 1}
atmTESTMIBCompliances   OBJECT IDENTIFIER ::= {atmTESTMIBConformance 2}

-- Compliance Statements
atmTESTMIBCompliance   MODULE-COMPLIANCE
STATUS         current
DESCRIPTION
"The compliance statement for SNMP entities which represent ATM interfaces. The compliance statements are used to determine if a particular group or object applies to hosts, networks/switches, or both."

MODULE  -- this module

MANDATORY-GROUPS  { atmLoopbackGroup }

-- Objects in the ATM Loopback Test Group

OBJECT       atmLoopbackID
MIN-ACCESS  read-only
DESCRIPTION
"Write access is not required. This object is required for ATM systems supporting the atmLoopbackVpLocationID and atmLoopbackVcLocationID tests."
OBJECT           atmVplEndptSegmentEndPoint
MIN-ACCESS       read-only
DESCRIPTION      "Write access is not required. This object is
mandatory for systems that are supporting ATM loopback
tests."

OBJECT           atmVclEndptSegmentEndPoint
MIN-ACCESS       read-only
DESCRIPTION      "Write access is not required. This object is
mandatory for systems that are supporting ATM loopback
tests."

::= { atmTESTMIBCompliances 1 }
6. Acknowledgments

This document is a product of the AToMMIB Working Group. The authors would like to acknowledge Dawn Xie for her valuable suggestions for this memo.
7. References


8. Security Considerations

This MIB contains both objects which allow an administrator to perform tests on ATM interfaces. Tests can only be performed when using these objects through in conjunction with [22]. Unauthorized access to the associated objects could cause a denial of service, or in combination with other (e.g., physical) security breaches, could cause unauthorized connectivity to a device. The tests defined in this document are not service interrupting.

9. Authors’ Addresses

Michael Noto  
Network Equipment Technologies  
800 Saginaw Drive RM 21.1.111  
Redwood City, CA 94063  
Phone +1 415 569-7134  
EMail: mike_noto@net.com

Kaj Tesink  
Bell Communications Research  
331 Newman Springs Road  
P.O. Box 7020  
Red Bank, NJ 07701-7020  
Phone: (732) 758-5254  
EMail: kaj@cc.bellcore.com

10. RFC Editor and IANA Considerations

Prior to publication of this memo as an RFC, the RFC Editor and IANA are requested to make a suitable OBJECT IDENTIFIER assignment and update the following in the MIB:

-- * replace { experimental XX } with { mib-2 YY }  
-- * remove ‘experimental’ from the IMPORTS clause  
-- * assign YY by IANA.  
-- * remove this notice in the MIB
11. Full Copyright Statement

Copyright (C) The Internet Society (1998). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
Table of Contents

1 Status of this Memo ........................................ 1
2 Abstract .................................................. 2
3 The SNMP Network Management Framework ................. 2
4 Overview .................................................. 4
4.1 Background .............................................. 4
4.2 Terminology ............................................ 5
4.3 Supported Functions ..................................... 6
4.3.1 ATM Loopback Testing .............................. 6
4.3.1.1 ATM End-Point Tables .......................... 10
5 Definitions ............................................... 11
6 Acknowledgments .......................................... 29
7 References ............................................... 30
8 Security Considerations .................................... 33
9 Authors’ Addresses ........................................ 33
10 RFC Editor and IANA Considerations ..................... 33
11 Full Copyright Statement .................................. 34

Expires March 1, 1999