A Demonstration of SMDS Customer Network Management Features

Theodore O. Brunner
Bellcore, MRE 2P-252, 445 South St. Morristown NJ, 07960
tob@thumper.bellcore.com
wk 201/829-4678 fax 201/829-2504

Abstract
Switched Multi-megabit Data Service (SMDS) is a high-performance, public data service being offered by the Regional Bell Operating Companies. Targeted toward LAN interconnect, this datagram service offers features to better support LAN needs. These include multicast and unicast addressing, high access speeds, and Customer Network Management (CNM). This SMDS CNM service provides the customer with access to management information collected within the SMDS network. CNM uses the popular Simple Network Management Protocol (SNMP) to provide access to this management information. Based on a managed object paradigm, SNMP presents the managed resource as a set of managed objects, organized into a Management Information Base (MIB) module. SMDS CNM supports five MIB modules. This paper presents a prototype implementation that was developed for demonstration at INTEROP'91, INTEROP '92 Spring and ICA, and used by at least one Bell Telephone Company in field trials of SMDS CNM. The material in this paper focuses on architectural issues in the implementation of an SMDS CNM agent. In particular, the model for information flow through the CNM agent is considered.

1. Introduction
Switched Multi-megabit Data Service (SMDS) is a public, high-performance, packet switching service that is being introduced in US metropolitan areas by the Regional Bell Operating Companies. Long-distance carriers are planning to support inter-exchange SMDS services, and numerous LAN equipment vendors are making products available to support this new service. SMDS provides LAN-like performance over metropolitan and wide areas (HKM). It provides subscriber access at speeds from 1.5 Mbits/sec (DS1) to 45 Mbits/sec (DS3); SONET rates are expected in the future.

SMDS offers a datagram service that is supported by a cell based technology; the IEEE Metropolitan Area Network standard 802.6 [Bell772].

Targeted toward LAN interconnect, SMDS provides a datagram service using Media Access Level frame encapsulation similar to IEEE 802.X. It supports both unicast and multicast addressing. Unicast addressing is provided through SMDS individual addresses, which are assigned to a Subscriber Network Interface (SNI). Multicast addressing is provided through SMDS group addresses which are intended for the multicast protocols commonly used by subscribers (e.g. Address Resolution Protocol (ARP)). Subscribers can define the members of a group address by enumerating the individual addresses included in a group.

SMDS provides both inclusive and exclusive address screening, which allows subscribers to define virtual private networks. Subscribers can populate the screening lists of their own SNIs.

To support generalized routing functionality, SMDS allows subscribers to define multiple SMDS addresses on a single SNI.

Through Customer Network Management (CNM), customers gain access to network management information pertinent to their own SNI, gathered and maintained within the SMDS Network [Bell1062]. This information includes "health" and status of the subscribers access line, counts and error statistics at both the cell and datagram levels as seen by the SMDS Network, and enumeration of subscriber addresses, group address members, and screening table members as used by the SMDS Network. Through CNM they may also make configuration changes to their service features, including the members of a group address and of an address screen [CKT] [Bell-MIB2].

The SMDS CNM service allows subscribers to manage their public network service just as they manage their own private network. As such, its greatest utility comes when it meshes with existing management practice at the customer site [CPT].

This paper describes the SMDS CNM prototype that was developed for and demonstrated at the interoperability trade show INTEROP'91 in San Jose, CA during the week of October 7-11, 1991. It was also demonstrated at INTEROP'92 Spring in Washington DC and at International Communications Association (ICA) in Atlanta, GA during the week of May 18-22, 1992. A companion paper details the whole SMDS demonstration [JLT]. In
addition, it is being used by at least one Bell Telephone Company in field trials of SMDS and SMDS CNM service.

The demonstrations coordinated several functions on a number of machines. An Agent implemented CNM features on an AT&T BNS-2000™ switching platform, thereby filling the role of the Telephone Company Service Provider. Manager stations from several vendors (Cabletron, DEC, HP, IBM, SNMP Research, and SUN) filled the role of customer Network Management Stations. All successfully interoperated with the CNM Agent.

2. SMDS CNM

The protocol chosen for SMDS CNM is the Simple Network Management Protocol (SNMP) - which is the de facto, open standard protocol for network management in the LAN/WAN environment. It uses the managed object paradigm. Through it, a managed network element is presented to the manager as a set of managed objects, which may be read or written to, much like a virtual database. A collection of managed objects forms a Management Information Base (MIB) module, whose format follows the Structure of Management Information (SMI) [RFC1155] [RFC1212]. MIB modules are supported by an Agent, usually resident on the managed network element, which responds to queries and requests from a Manager, resident on a Network Management Station (NMS). An Agent may represent one or more managed network elements remotely, and is then termed a Proxy Agent. An Agent may respond to one or more Managers, and a Manager may issue queries to one or more Agents.

MIB modules are often organized around a technology, and Agents support as many MIB modules as appropriate - in addition to the core MIB, termed MIB-II [RFC1213].

SNMP supports communications between manager and agent, employing a polling paradigm [RFC157]. The Manager polls the Agent for specific managed objects using a get or get-next function, and writes using the set function. The Agent may also notify the Manager of exceptional events using the asynchronous trap function.

By design, and through ubiquitous implementation, a customer managing a data network using SNMP gains several advantages:

- Access to remote network elements for management purposes.
- Access to multiple (public or private) network elements through a single NMS.
- Access to network elements from different vendors.
- Multiple NMSs with the same network view.

Through SMDS CNM, the customer's NMS can access management information associated with the customer's own SNIs. It also receives fault notification through trap messages from the SMDS CNM Agent. It accesses this information along with management information from the customer's own private network, integrates and presents all of it through a single management station.

Figure 1 shows how customers can use CNM to manage their SMDS network. The CNM Agent resides within the SMDS network, and NMSs gain access to that Agent via their SMDS Network (with out-of-band backup). The CNM Agent supports multiple customers, and represents the whole SMDS Network, quite likely served by multiple SMDS switches.

![Figure 1. Use of CNM](image)

Note that customer routers at the edge of the SMDS Network may support much of the same information as the CNM Agent. Since they support the opposite end of the customer's SNI, they have a complementary view of the customers traffic and transmission link (SNI). This is one of the features of CNM - the values of the managed objects on opposite ends of the SNI can be compared.

2.1. SMDS CNM Information

The SMDS CNM Agent presents SMDS Network behavior to the customer as a collection of managed objects that represent the customer’s SNIs. (These objects do not represent the SMDS switch per se, since its health is not directly related to the management of a particular customer’s SNIs, rather to the Network as a whole.) These managed objects make up most of the following five MIB modules.

- MIB-II, which instruments core systems functions that every SNMP Agent supports [RFC1213].
- DS1 and DS3 MIBs, which instrument the transmission media over which the SMDS subscriber access is carried [RFC1232] [RFC1233].
- SMDS Interface Protocol (SIP)MIB, which instruments the cell and the packet level technology over which SMDS is carried [RFC1304].
- SMDS Subscription MIB, which instruments the special features of SMDS, including group addressing and address screening (BellMIB1).

Some management objects will have only one instance -
e.g. the name of a SMDS CNM Agent - while others will have multiple instances - e.g. the names of interfaces - since there are potentially more than one in a system. In the SNMP Management Framework, objects with multiple instances are organized in a conceptual table. The conceptual rows of the table correspond to the number of instances, the conceptual columns correspond to the managed objects. In the following, many of the objects are organized in tables whose rows correspond to SNIs. But some are organized in tables whose rows correspond to various addressing entities. These tables can be viewed as ordered lists. What follows is an accounting of the managed objects of these MIB modules and the information they contain.

The MIB-II supports objects chosen as the core functionality needed to support an Internet implementation. Some objects are not needed for the management of SNIs - those instrumenting protocol layers above the Media Access Level of SMDS. So only portions of MIB-II are supported for SMDS CNM: the **system group** and the **interfaces group**. The **system group** describes the system as a whole - in this case, the SMDS CNM Agent - and includes its name, location, a contact, and its Up Time. The **interfaces group** describes the interfaces - in this case, the SNIs - and includes their speed and type. It also includes an object that describes the general health of the SNI: its Operational Status. General purpose unsolicited alarms (traps) are sent to customer NMSs. The link-gone-down and link-back-up traps are sent on a change in the state of health of an SNI.

The DS1 and DS3 Interface MIBs support objects in several categories: (1) **Configuration**, which includes the transmission line type, zero code and loopback; (2) **Health**, which includes an indication of any outstanding alarms on the SNI - red and yellow alarms; (3) **Error statistics**, which contains the largest portion of objects and tables in this MIB. The number of seconds of service that have errors in excess of various thresholds are recorded in objects, including errored seconds, errored framing seconds, severely errored seconds, etc. These objects keep count during 15 minute intervals. A history is kept of those 15 minute intervals that grows up to 24 hours deep.

The SIP MIB instruments the three levels of the SMDS protocol: SIP Level 3 or datagram level, into which the service encapsulates customer frames; SIP Level 2 or cell level, which carries the 48 byte data + 5 byte header cells; and a portion of SIP Level 1, which is the Physical Layer Convergence Protocol (PLCP) for either DS1 or DS3, and maps cells onto the transmission media. Datagram level objects include counts (sent and received, group and individual addressed) and error statistics (errored datagram, unrecognized or invalid address). Cell level objects also include counts (sent and received) and error statistics (checksum, payload, sequence, MID errors). PLCP level objects include health indicators (red/yellow alarm) and error counts (e.g. severely errored framing seconds). Objects that instrument IP functionality are not supported by SMDS CNM, but would be supported by a router.

The SMDS subscription MIB contains various configuration information, primarily concerning addressing. Some of the addressing objects are writable, allowing the customer to make changes to their own SMDS network portion. Subscription parameter objects include access class, maximum simultaneous number of datagrams in or out of the network. Error statistics objects monitor various violations. The subscriber addresses table lists the unicast SMDS addresses assigned to a particular SNI. The individual address screening table lists the unicast addresses that will be screened at the SNI (either inclusively or exclusively). The group address screening table lists the multicast addresses that will be screened at the SNI (either inclusively or exclusively). The group address table lists the unicast members of a multicast group address. The member group address table lists all the multicast group addresses in which a particular unicast individual address has membership. The subscription MIB lists a number of SMDS specific traps that indicate a customer requested change to the addressing tables has taken effect.

### 2.2. Customer NMS

The customer’s NMS accesses and presents objects taken from the CNM MIB modules. When subscribed, an NMS is supplied with machine readable, standard descriptions of the MIBs supported by the SMDS CNM Agent. The NMS automatically incorporates these MIBs, extracting the syntax of the managed objects, through a MIB compiler. Then, the NMS receives traps and polls the agent for various MIB objects, employing trap directed polling to support a remote debugging paradigm. It presents these values to the human operator using whatever style user interface is particular to the NMS vendor. Since the purpose of CNM is to manage a customer’s use of the SMDS Network, the customer’s NMS may organize its presentation of information around important management applications. Possibilities include:

- Monitoring SNI health as a graphical display.
- Plotting traffic flow across an SNI.
- Displaying various address tables.

For an example picture of the NMS’s graphical interface,
A map displaying the health of a network, with green icons for the up and red for the down state, is a popular and instantly understandable application. This sort of application queries for summary objects that indicate the overall health of the SNI. Commonly, the object queried is the operational status of the SNI. The appropriate SNI polling frequency for the application should be several tens of seconds (see below).

A plot of cumulative traffic across an SNI is another popular application. It shows SMDS traffic history at a glance. This sort of application queries for specific counts, often the datagrams or cells across the SNI. The application polls its SNI with a frequency of a minute or so. The polling frequency determines the level of detail in the plot.

SNMP provides for the retrieval of the objects in a table in a stepwise fashion; the table may be returned one row at a time. Displaying the addressing tables or lists, by stepping through them is an obvious application. But it is infrequently used: first when the customer learns the system, and later for debugging purposes. For instance, vanishing datagrams may suggest an address assignment or address screening error. At such a time the customer may display those lists searching for a problem.

Most of the CNM MIB objects are related to a particular SNI, but a few are network-wide and may be displayed separately. For instance, the relatively static information held in the systems group of MIB II, identifying the CNM Agent, may be separated from the SNI specific information.

The behavior of these applications need not and should not follow the highest polling rates that can be supported in a LAN environment, where the bandwidth for a query every second is available. The querying frequencies should be more leisurely, with the highest frequency polling on the order of tens of seconds. This is completely adequate for the goal of the application, and consistent with the tradition of management in the WAN environment, where bandwidth is more limited. Moreover, the SMDS CNM Agent has been designed to update its CNM information on that time scale. The next section will discuss performance issues.

3. SMDS CNM Proxy and Agent

An SMDS CNM Agent is implemented in two parts: an Agent that resides at the SMDS switch (on the Supervisory System (SS) processor which supports the Supervisory Functions (SF)) and a Proxy that resides on a separate system [Bell1062]. In the case of the demonstration SMDS CNM Agent presented here, the interactions between Proxy and Agent are as follow. The Proxy receives a query from the customer NMS, determines where the information resides, and either responds itself or relays the query to the Agent. The information flow is as follows. The customer NMS queries the Proxy. If the information resides locally on the Proxy, it returns a response. If the information resides on the Agent, the Proxy sends a remote query to the Agent. The Agent gathers the information and returns a response to the Proxy, which, in turn, returns a response to the customer NMS. Figure 3 identifies these components.

![Figure 3. Proxy and Agent](image)

There are a number of reasons why an architecture that includes both a Proxy and an Agent is attractive.

- Single point of contact. The Proxy acts as a single point of contact for customer CNM, regardless of the number or type of SMDS switches deployed in the SMDS network. The customer is shielded from changes in the network's topology. Also the SMDS Network Operations Staff can centralize their CNM operations at one proxy.
- Protocol separation. It is possible with a Proxy and Agent, to support one protocol for the customer and a different one for SMDS switch management. This decouples the evolution of network management technology appropriate to the LAN/WAN environment from that appropriate to the public network switching environment. Though the two technologies coincided in SNMP for this implementation, they need not in the future.
- Access Control. The Proxy can present to a customer information exclusively about that customer's SNIs.
- Security. The proxy can authenticate the customer query, thus protecting the agent and the switch from unauthorized queries. From a security prospective, the proxy provides the "firewall". For more on how
SNMP implements security; see [RFC1351] [RFC1352] and [RFC1353]. As an additional mea-
sure of security, it is also possible to secure the rou-
ter through which the customer NMS gains access to
the proxy.

- **Flow control.** The proxy multiplexes queries from
  the various managers and queues them for delivery
to the agent. It may exercise control over the number
of outstanding manager requests delivered to the
agent. If it receives too many requests, it may drop
some at its own input queue, before those requests
claim any processing resources at the agent. The ob-
servation assumes that the agent's processing power
is to be conserved, which was certainly true for this
implementation.

- **Conservation of processing resources at the agent.** It
  is possible to off-load responsibility for certain static
MIB objects, from the agent to the proxy.

- **Customer service functions.** There are reasons that
don't have to do with CNM, but with customer ser-
vice functions needed to operate a network - func-
tions performed by a network Operations System
(OS) which is distinct from the switch. For instance,
these functions come into use when service is con-
nected for a new customer, or when a new SNI is
added to the customer's SMDS network. These
functions are not a part of switch management and
should not reside on the switch. Furthermore the OS
may not be developed by the same vendor as the
switch and its SS. But these functions do interact
with the configuration information CNM maintains.
A separate Proxy and Agent fit more easily into this
environment.

### 3.1. **CNM Architecture Issues**

There are several architectural issues that helped shape
the overall design of the demonstration SMDS CNM
Agent. A novel feature of the Proxy / Agent division, is
that the Proxy uses SNMP to communicate with the
Agent. Thus SNMP is used as a management protocol,
within the SMDS internal network, and directly to
the SMDS switch. This design choice allowed the rapid
implementation of a robust Proxy plus Agent.

The SMDS CNM Agent is more complex than many other
agents implemented under the SNMP management
framework. The SMDS CNM Agent represents a unified
switching system supporting multiple SNIs to the cus-
tomer NMS as a single entity. In reality, each switch con-
sists of many independent systems. The status of each
system, and communications between them, is sub-
merged beneath the unified view, and must not interfere
with it. Quite a number of computer systems and UNIX
processes work together to support this unified view.

The SMDS CNM Agent is responsible for multiple cus-
tomer NMSs. There is a great potential for the Proxy and,
in particular, the Agent to be overrun with manager re-
quests.

The SMDS CNM Agent fulfills different expectations of
network management information availability than does
the switch's SS. The SS supports the internal network
management defined in [Bell774], while the Proxy sup-
ports CNM defined in [Bell1062]. As a result the CNM
Agent must negotiate differences in MIB object seman-
tics from [Bell774] to [Bell1062] through translation. It
must also negotiate differences in query latency through
caching.

### 3.2. **Design Trade-Off**

The managed objects associated with traffic statistics are
normally kept by that subsystem of the switch that inti-
nately deals with the traffic - the SNI subsystem. For in-
fstance, a count of the cells passing the SNI is normally
kept by the hardware and software that segments data-
grams into cells and reassembles them. Through network
management, the count kept by the SNI subsystem is
supplied to a customer management system.

When supplying management information there are two
models for the information flow. The first, dubbed "on-
demand," follows the pattern outlined above (section
3.0) in describing the Proxy / Agent division. A request
from the customer initiates a request all the way to the
SNI subsystem keeping the information. The second,
dubbed "cache and query," introduces an intermediate
cache. A request from the customer is answered direc-
tly from the cache, Independently, the cache is updated in
some regular polling arrangement.

To compare the models we consider two characteristics
of the information returned to the customer: the latency
of the response to a query, and the currency or timeliness
of the information when it is returned. The "on demand"
model is most attractive because of the currency of the
information collected. The query goes straight to the
source.

It is difficult to build a distributed network management
system that achieves low latency and high currency. The
SNI subsystem collecting traffic counts has better things
to do than answer questions about how many cells it has
processed - it has to process and switch them! Essenti-
ally, transport is more important to that subsystem than
management and will always have prior claim to pro-
cessing resources. So queries may be delayed in unpre-
dictable ways.

The "cache and query" model provides low query re-
response latency but does so by providing less timely in-
formation. Queries that are fielded by a cache can be quickly responded to, but the cache may reflect old information. To update the cache, the subsystem is queried at some acceptable rate. Building to such a model is not hard, but the difficulty is in choosing the query rate: an acceptable rate to the subsystem may seem too slow to the customer. Furthermore, the cache solution seems wasteful. The SNI subsystem designer sees the update rate as too fast if the cached data isn't read (99% of the time), and sees that caching as wasted effort, while the customer sees the update as too slow if the data is polled more frequently than the updates.

So in the first model the response latency tends to vary widely and unpredictably, and in the second model the information that is returned is not timely. There is a trade-off here, between the response latency and the age of the information.

We ultimately chose to implement the "cache and query" solution for several reasons. First, the SNI subsystem responsible for SMDS transport is insulated from the activity of the customer. Regardless of how many queries are issued by the customer, the management work load on the SNI subsystem is predictable. This was particularly important since the customer NMSs were newest generation workstations, and the SS processor and SNI subsystem processor were older microprocessors. Second, for a while during development of the CNM Agent the "on demand" model simply did not work. We could not guarantee that the latency of a query response would not exceed a reasonable customer time-out value.

4. **CNM Agent Design History**

This section presents the design of the demonstration SMDS CNM Agent, and some design history. The demonstration CNM Agent implements a subset of the information features outlined in section 2.1 above. One design decision we made was to implement all of what we saw as the "primary" functionality of SMDS CNM, and delay or omit implementation of "secondary" functionality. In particular, the DS3 MIB is not implemented at all - it being a virtual carbon copy of the DS1 MIB. Parts of the DS1, SIP and SMDS MIBs are also omitted. We believe the core features were not lost.

Write or set functionality is not supported in this CNM Agent. This is a direct result of the SMDS switch supervisory system, and its choice of what to give the agent access to. So the CNM Agent demonstrates monitoring functionality only.

As a smaller point, at all demonstrations, the Proxy and Agent were co-located on the same ethernet as the customer management stations. In an actual deployment scenario the management stations would use the SMDS network to reach the Proxy.

4.1. **Switching Platform**

The demonstration SMDS CNM Agent is implemented on an AT&T SMDS switch platform, called the BNS-2000™. During at least part of the time period covered by this paper, (July 1991 to March 1992) this AT&T switch was still in its development cycle. Various upgrades occurred in its software, and may continue to occur in both hardware and software.

An SNI subsystem - the Access Unit (AU) - provides the switch side termination of the SNI, and supports SMDS datagram and cell level functions. Currently these AUs are located out-board of the BNS-2000™ switch. An in-board version called an Access Interface (AI) - AU on a card - will be supported in the future.

The AT&T Starkeeper™ is a Supervisory System that manages one or more BNS-2000™ switches. It supports a variety of fault, performance and configuration management functions (Supervisory Functions), through a graphical user interface.

4.2. **Agent**

The Agent portion of the CNM Proxy / Agent resides on the SS processor. However, it does not use the SS to gather any management information. Instead a remote procedure call (RPC) library was made available for the purposes of the demonstrations. That library establishes a virtual-circuit to each SNI subsystem, and retrieves tables of management information from them. The Agent uses this RPC library to achieve direct access to the tables of information maintained by the SNI subsystems.

Most traffic counter objects in the DS1, SIP and SMDS MIBs, are available as one entry each, in various SNI subsystem management tables. So a single customer request could potentially be satisfied by one table request through the RPC library. But the Agent does not pass requests on through to the SNI subsystems. Instead it follows the "cache and query" model of information flow.

There are several reasons for placing the cache in the Agent. It is closest to the RPC library, which is the source of management information to the CNM Proxy / Agent. The constant querying to update the cache does not have to pass between the Proxy and Agent, consuming resources on both. The management tables do not have to be translated into MIB objects, as they would to be carried between Proxy and Agent. That translation would involve overhead of two types: conversion into ASN.1 syntax (the presentation language used by SNMP), and mismatch between SNI subsystem management table values and MIB objects. For instance, the semantics of
the traffic counter objects supported by the SNI subsystems are based on 15 minute intervals - not based on a continuous count since the boot time of the SNMP Agent, as are the counter objects in the SIP and SMDS MIBs. So a translation must be made, and it must be made at the Agent, if SIP and Subscription MIB objects are used between Proxy and Agent.

The Agent is implemented as multiple processes which communicate through shared memory (which holds the cache). There is an SNMP process that responds to queries from the Proxy by accessing the cache. There are multiple updating processes (one per SNI subsystem) that update the cache at regular intervals. For updating, multiple processes were preferred to a single process executing a round robin, since a crashed SNI subsystem could hang the updating for all SNI subsystems. On the other hand, many processes do tie up a significant amount of system resources, both memory and communication sockets.

4.3. Proxy

The Proxy portion of the CNM Proxy / Agent resides on a SUN workstation. It is designed to generate remote queries to obtain the values of objects that it does not support locally. When a customer SNMP query arrives, the Proxy parses the list of objects requested, and formulates a remote query that contains a list of the remotely maintained objects. When the Agent responds to the remote request, the Proxy joins the "remote" to the "local" objects, and responds to the original query. The objects the proxy supports locally are of two kinds: (1) static self-descriptive objects from MIB II and (2) static address tables from the Subscription MIB.

The Proxy is implemented as two processes which communicate through shared memory. The first process implements the proxying function above. The second process waits to receive alarms from the SS detailing faults in the SMDS switch and processes those alarms. Many alarms have no bearing on CNM and are ignored. Others may indicate a loss of signal or a loss of frame on an SNI. Based on the incoming alarm sequence, it both generates traps to the customer manager stations and changes the values of the health objects in MIB II, and the DS1 and SIP MIBs. For instance the Operational Status object may change from up to down for a particular SNI.

5. Conclusions

The function of CNM in the SMDS service has been presented. Design decisions taken in developing a prototype CNM Agent have been presented, and related to trade-offs involved in choosing an information flow model. Two models have been presented, the "cache and query" and the "on demand" models. The former is deemed preferable for SMDS CNM, for several reasons.

5.1. Future Improvements

In a number of regards this CNM Agent could use improvement. This is not surprising because it was developed in two months, by two programmers who started with various software piece parts. The result was not so much production software as "demoware." Nonetheless, during the Trade Shows and field trials of SMDS and CNM, it has behaved quite well.

A list of items for future work would include:

- Access control at the instance level. Customer's access should be limited to his own SNIs.
- Security. (The SNMP Security Model [RFC1351] [RFC1352] [RFC1353] addresses this.)
- Proxy support for multiple SMDS switches.
- More Operations features for memory management (eg. to add an SNI).

6. Acknowledgments

We gratefully thank all those who helped with the SMDS CNM demonstration. The effort of network management station companies, including Cabletron, DEC, HP, IBM, SNMP Research and SUN, and their personnel is appreciated. Thanks go to Sholom Fried and Jung Tjong of NYNEX for assistance with the CNM Agent software. Special thanks go to Dan Yip and Guru Raman of AT&T, and their co-workers, for their extra effort in helping make CNM happen. Special thanks go to Jeff Case of SNMP Research. The effort of Bellcore personnel who helped with the demonstration is gratefully appreciated and thanks go to Cathy So for developing the CNM applications. And finally, special recognition should go to Sherri Hiller, my co-developer, who built the Agent portion of the CNM Proxy / Agent.

7. References


[Bell774] SMDS Operations Technology Network Element Generic Requirements, Bellcore Technical Refer-


