ISO/CCITT and Internet Management Coexistence (IIMC):

Translation of Internet MIBs

to

ISO/CCITT GDMO MIBs

(IIMCIMIBTRANS)

9 October 1992

Lee LaBarre

The MITRE Corporation
Burlington Road
Bedford, MA 01730
cel@mbunix.mitre.org

Status of this Memo

This memo provides information to the network and systems management community. This memo is intended as a contribution to ongoing work in the area of multi-protocol management coexistence and interworking. This memo is part of a package of ISO/CCITT and Internet Management Coexistence (IIMC) drafts; see also [IIMCOMIBTRANS] [IIMCMIB-II] [IIMCPARTY] [IIMCPROXY].

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. Internet Drafts may be updated, replaced, or obsoleted by other documents at any time. It is not appropriate to use Internet Drafts as reference material or to cite them other than as a "working draft" or "work in progress".

Please check the 1id-abstracts.txt listing contained in the internet-drafts Shadow Directories on nic.ddn.mil, nnsc.nsf.net, nic.nordu.net, ftp.nisc.sri.com, munnari.oz.au to learn the current status of any Internet Draft.

Distribution of this memo is unlimited. Comments on this memo should be sent to iimc@thumper.bellcore.com by November 20, 1992.

Abstract

This memo is intended to facilitate the multi-protocol management coexistance and interworking for networks that are managed using the OSI Common Management Information Protocol (CMIP) and networks that are managed using the Simple Network Management Protocol (SNMP). This RFC contains translation and registration procedures that are applicable to translation of Internet MIBs defined according to the Internet Structure of Management Information (SMI) into ISO/CCITT SMI defined MIBs. It also defines management information and SNMP trap to ISO/CCITT event mappings required for ISO/CCITT-Internet proxy.

Table of Contents

Status of this Memo i
Abstract ii
Table of Contents ii
1. Introduction 1
1.1 Background 1
1.2 Overview 2
1.3 Scope 4
1.4 Terms and Conventions 5
2. Registration and Naming Procedures 5
2.1 Registration Procedures 5
2.1.1 Object Classes and Attributes Registration 7
2.1.2 Notifications Registration 7
2.1.3 NAME BINDINGs Registration 8
2.2 Naming Procedures 8
2.2.1 Naming Attribute 8
2.2.2 ISO/CCITT-Internet Proxy Naming Tree 9
2.2.3 Distinguished Names 11
2.3 OID Translation 12
2.3.1 OID/Name Translation: ISO/CCITT to Internet 12
2.3.2 OID/Name Translation: Internet to ISO/CCITT 14
2.4 Inheritance for Object Classes 14
2.5 Reference Labels for Derived Entities 14
3. Internet to ISO/CCITT MIB Translation Procedures 14
3.1 Pre-translation Procedures 15
3.2 GDMO Translation Procedures 16
3.2.1 Translation of Groups 17
3.2.2 Translation of Table Objects 19
3.2.3 Translation of Table Entry Objects 19
3.2.4 Translation of Other OBJECT-TYPES 20
3.2.5 Translation of Traps 23
3.2.6 Translation of Internet Attribute Types 24
3.3 Post-translation Procedures 25
3.3.1 Post-translation of BEHAVIOUR Cause 26
3.3.2 Post-translation of Notifications 26
3.3.3 Creation of NAME BINDING Templates 27

4. ISO/CCITT-Internet Proxy MIB 28
4.1 Object and Attribute Definitions 29
4.2 Name Bindings 34
4.3 Common SNMP Derived Attribute Types 35
4.4 Notifications for SNMP/SNMP-2 Traps 41
4.5 ASN.1 Definitions 44
4.6 ISO/CCITT-Internet Proxy Communications 45
5. Acknowledgements 45
References 47

1. Introduction

The past decade has witnessed the development of enterprise wide networks composed of a multi-vendor environment containing heterogeneous protocol and hardware suites. Organizations have become increasingly dependent on these enterprise networks for their daily operations. This dependence has focussed attention on the need for operation, administration, maintenance, and provisioning (OAM&P) of the multi-vendor enterprise network on an end-to-end basis.

1.1 Background

This memo is part of a package of ISO/CCITT and Internet Management Coexistence (IIMC) drafts. Other memos included in this package are:

- Translation of ISO/CCITT GDMO MIBs to Internet MIBs (Newnan) [IIMCOMIBTRANS]

- Translation of Internet MIB-II (RFC1213) to ISO/CCITT GDMO MIB (LaBarre) [IIMCMIB-II]

- Translation of Internet Party MIB (RFC1353) to ISO/CCITT GDMO MIB (LaBarre) [IIMCPARTY]

- ISO/CCITT to Internet Management Proxy (Chang) [IIMCPROXY]

These memos together comprise a package aimed at integrating ISO/CCITT-based and Internet-based management systems. These memos are offered as input to coexistence and interworking efforts underway throughout the industry,including organizations such as:

- IETF OSI Internet Management (OIM),
- Network Management Forum Technology Convergence Team,
- X/Open Systems Management (SysMan),
- OIW Network Management Special Interest Group (NMSIG), and
- OSF Management Special Interest Group (MANSIG).

This work was initiated, in part, by NM Forum efforts to translate RFC 1214 for use with OMNIPoint 1 implementations. Through this effort, it became obvious that end-to-end management requires an integrated, unified view of the managed network, despite differences in management protocol and information structure. Integrated management can be facilitated by the development of "proxy" mechanisms which translate between functionally equivalent service, protocol, and SMI differences to create this unified view. MIB translation procedures can be used to support proxy management, as well as to take advantage of existing MIB definition and avoid duplication of effort. In this way, commercial investment in both ISO/CCITT and Internet-based

management technologies can be preserved through deployment of common methods and tools which support integration.

This overall strategy was outlined in a joint publication developed by the NM Forum and X/Open entitled "ISO/CCITT and Internet Management: Coexistence and Interworking Strategy" [NMFMC92]. The memos included in the IIMC package are intended as detailed specifications which implement several of the methodologies identified in this strategy.

1.2 Overview

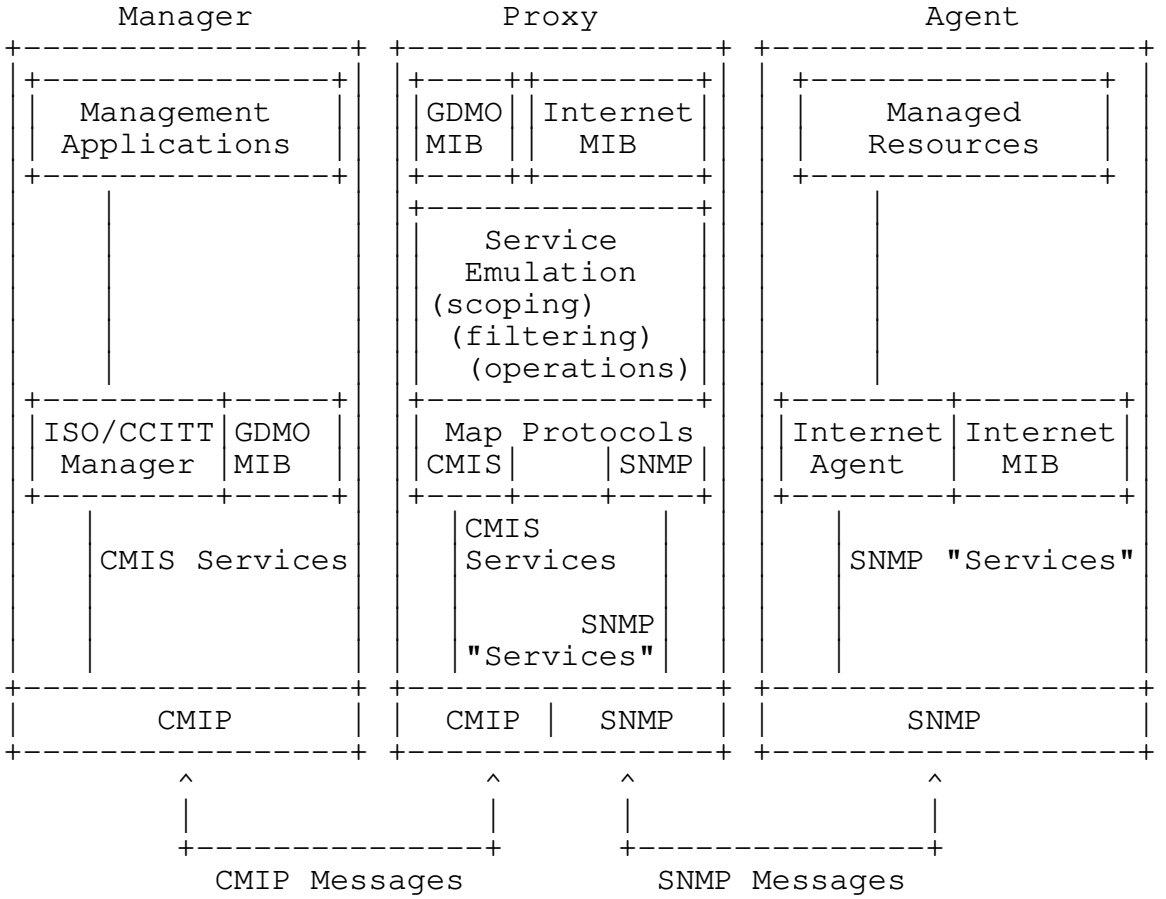
The response to the need for OAM&P of enterprise networks has been the development of network management standards within various networking communities - most notably the ISO/CCITT and Internet community. However, coordination of standards activities between these two communities has not occurred. As a result, although they share a nearly common management model, differences in their management protocols and structure of management Information (SMI) have developed due to differing management philosophies.

The ISO/CCITT community has developed the Common Management Information Protocol (CMIP) [ISO9596], and related SMI documents [ISO10165-1,3,4]. The Internet community has developed the Simple Network Management Protocol (SNMP) [RFC1157], and is developing its successor, SNMP-2, based on [SMPPROT]. The Internet SMI is defined in [RFC1155] and [SMPSMI]. Although functionally similar, the Internet and ISO/CCITT protocols and SMIs differ in terms of their complexity and specific operations.

The focus on the need for end-to-end enterprise management has indicated the need to integrate the management of components managed by ISO/CCITT management, Internet management and proprietary management mechanisms in a manner which presents a unified view of the network despite protocol and SMI differences. One way to integrate management is by the development of "proxy" mechanisms which translate between functionally equivalent services, protocol and SMI differences to create this unified view.

A body of telecommunications and computer vendors, represented by organizations such as the Network Management Forum (NMF), and the U.S. government, as specified in the Government Network Management Profile (GNMP) have based their integrated management model on the ISO/CCITT management model using CMIP and the ISO/CCITT SMI. These organizations are particularly interested in the development of proxies for devices that use the Internet management protocols and SMI. Their interest is primarily due to the widespread commercial implementation and use of such devices within their enterprises, especially devices that use the Internet TCP/IP protocol suite.

The basic model for ISO/CCITT-Internet proxy management is illustrated in the following diagram.



The proxy architecture provides emulation of CMIS services by mapping to the corresponding SNMP message(s) necessary to carry out the service request. The service emulation allows management of Internet objects by an ISO/CCITT manager. The left hand side of the proxy behaves like an ISO/CCITT agent, communicating with the ISO/CCITT manager using CMIP protocols. The right hand side of the proxy behaves like an Internet manager, communicating with the Internet agent using SNMP protocols.

The proxy relies on the existence of a pair of directly-related MIB definitions, where the Internet MIB has been translated into ISO/CCITT GDMO using the procedures specified in [IIMCMIBTRANS]. The proxy defined in [IIMCPROXY] uses these MIB definitions and rules to provide run-time translation of management information carried in service requests and responses.

The proxy architecture is designed with a specified interface between the proxy and the underlying protocol

stacks, and so deals primarily in terms of CMIS services and SNMP "services". The proxy emulates services such as CMIS scoping and filtering, processing of CMIS operations, and forwarding/logging of CMIS notifications by performing a mapping process which must be tailored for each protocol (for example, SNMP, Secure SNMP, and SNMP-2 are all variants of the same protocol mapping process).

Finally, [IIMCOMIBTRANS] specifies translation procedures for converting ISO/CCITT GDMO MIBs into Internet MIBs. MIBs generated by this translation process cannot be utilized by the Proxy defined in [IIMCPROXY], although another kind of Proxy could be defined for this purpose in the future.

1.3 Scope

A major reason for the rapid commercialization of devices manageable via the Internet management protocol is due to the speed with which the vendors in the Internet community have been able to develop MIBs based on the Internet SMI. To capitalize on this continuing Internet MIB development and their deployment in commercial devices, communities interested in integrated management via ISO/CCITT-Internet proxies require that procedures be defined for translation of Internet MIBs into ISO/CCITT GDMO MIBs, i.e., MIBs defined according to the ISO/CCITT SMI Guidelines for Definition of Managed Objects [ISO10165-4]. Such MIB translations may also minimize the independent development of ISO/CCITT GDMO MIBs for the same resources and thereby reduce the incompatibilities with the Internet MIBs.

Of particular interest to the community interested in ISO/CCITT-Internet proxies, are translation procedures which may be automated to a high degree and include unambiguous automated registration procedures. This memo defines such procedures.

This memo also defines the ISO/CCITT-Internet proxy management information and generic SNMP trap to CMIS notification mappings required for ISO/CCITT-Internet proxy managers, as well as common naming conventions and ASN.1 modules applicable to such proxies.

This memo assumes that the reader is familiar with the ISO/CCITT SMI and Internet SMI, and the terminology of each. The term SNMP will be used throughout the memo to indicate either SNMP or SNMP-2, unless a distinction needs to be made.

As of the publication date of this memo, the SNMP-2 SMI and protocol are evolving. It is expected that the [SMPPROT], [SMPSMI], [SMPMIB], [SMPTC] and related documents will be the basis of that evolution, and that changes will not

significantly affect the registration, naming, and translation procedures described in this memo.

Many, but not all, of the procedures defined in this memo are subject to automation. Comments are provided concerning possible aids to automation; however, it is not the intent of this memo to provide automated translation algorithms.

1.4 Terms and Conventions

TBD

2. Registration and Naming Procedures

Registration and naming procedures are crucial to the unique identification of management information. Registration assures the uniqueness of management information element types, while naming provides a way of distinguishing instances of a type and locating them within the MIB.

2.1 Registration Procedures

Registration procedures specify that changes in the syntax or semantics of registered entities require them to be registered as new entities. The process of converting Internet MIBs into the ISO/CCITT GDMO MIBs inevitably introduces subtle semantic changes in how data can be operated on, and changes in the content of the MIB element. For example, ISO/CCITT attributes that are converted from Internet Object Types acquire matching rules for use in filtering operations. ISO/CCITT object classes that are created from Internet groups acquire semantics related to their inheritance of new attributes from the "top" managed object class. The end result is that all the new ISO/CCITT object classes, attributes, and notifications created during the translation process must be registered. In addition, name bindings for inserting object instances into the naming hierarchy must be registered.

Registration procedures are critical to the goals of automating the translation of Internet MIBs to ISO/CCITT GDMO format, and the efficient implementation of ISO/CCITT-Internet proxies. Registration involves assignment of an ASN.1 Object Identifier (OID) to the entity. Management entities defined according to the principles of the Internet SMI may be registered under the IAB's "internet" arc, or registered under an arc in another organization's proprietary registration subtree. The effect of the registration procedure specified in this document is to graft the MIB elements defined as a subtree under the "internet", or proprietary, registration arc into another part of the registration tree without changing the subtree structure.

For example, OIDs registered under the internet arc are of the form:

OID = <internet> <internetEntityId>

where <internet> is the full registration path to the "internet" arc; and <internetEntityId> is the portion of the OID that uniquely identifies entities under that arc, i.e., the remainder of the OID.

OIDs registered under another organization's arc are of the form:

OID = <organization> <internetEntityId>

and <organization> = <org arc> <org internet part>

where <org arc> is the arc assigned to the organization by ISO, CCITT, ANSI, or some other registration authority; <org internet part> is the path within the organization's subtree down to the subtree that identifies MIB entities defined according to the Internet SMI (an organization's internet subtree); and <internetEntityId> is that portion of the OID that uniquely identifies entities within the organization's internet subtree.

Registration is accomplished by replacing the <internet> or <organization> portion of the OID with a new registration arc allocated for proxy registration such that the OID is of the form:

OID = <cmipsnmpProxyXX> <internetEntityId>

This procedure preserves the unique identification of the entities within the subtree, and since the new arc to which it is attached is unique, assures a unique OID. Also, since the original subtree information is present in the new OID, implementation of proxy translation of the corresponding management entities identified differently in the two SMIs is made easier.

Internet defined groups and objects must be registered as ISO/CCITT object classes and attributes; Internet traps must be registered as ISO/CCITT notifications; and ISO/CCITT name bindings must be defined and registered. This memo assumes that an arc has been allocated in the registration hierarchy for use in ISO/CCITT-Internet proxy registration. The arc is named "cmipsnmpProxy". This memo assigns sub-arcs under "cmipsnmpProxy" as follows:

```
cmipsnmpProxy       OBJECT IDENTIFIER ::= {...}

cmipsnmpProxyIMIB   OBJECT IDENTIFIER ::= {cmipsnmpProxy 1}
```

```
     -- for ISO/CCITT derived object classes and attributes

cmipsnmpProxyNB     OBJECT IDENTIFIER ::= {cmipsnmpProxy 2}
     -- for ISO/CCITT derived NAME BINDINGS

cmipsnmpProxyNOT    OBJECT IDENTIFIER ::= {cmipsnmpProxy 3}
     -- for ISO/CCITT Notifications
```

This memo assigns OIDs in 4.5 for registration of ISO/CCITT entities derived from internet entities registered under the "internet" arc, and for registration of entities defined in this memo.

The following describes registration procedures to facilitate automated translation and assure uniqueness of registered ASN.1 object identifiers for ISO/CCITT object classes, attributes, and notifications derived from internet entities; and a registration procedure for their name bindings.

2.1.1 Object Classes and Attributes Registration

Follow the procedure described in 2.1, using the internet assigned OID for the group, conceptual table, or conceptual table entry from which the object is derived, with {cmipsnmpProxyIMIB} substituted for {cmipsnmpProxyXX}.

2.1.2 Notifications Registration

Notifications are registered using the OID corresponding to the value of the Internet smpTrapOID object defined in [SMPMIB].

For SNMP trap PDUs the smpTrapOID is derived as stated in the SNMP/SNMP-2 Coexistance memo [SMPCOEX] 4.1.2(2). That definition is repeated below:

"... if the value of the generic-trap field is 'enterpriseSpecific' then the value used is the concatenation of the enterprise field from the trap PDU with additional sub-identifiers, '0', and the value of the specific-trap field."

For traps defined according to the SNMP-2 SMI, the registered OID for the TRAP-DEFINITION macro is the value of the smpTrapOID.

The registered OID for the ISO/CCITT derived notification is then translated from the value for smpTrapOID according to the procedures in 2.1 with {cmipsnmpProxyNOT} substituted for {cmipsnmpProxyXX}.

2.1.3 NAME BINDINGs Registration

As described in 2.2.2 , the ISO/CCITT SMI requires that managed object instances be bound into a naming hierarchy, or tree, for purposes of naming. ISO/CCITT NAME BINDING templates are used to register the manner in which such instances may be bound. These name bindings must be registered.

The Internet SMI does not include the concept of a naming tree and name binding. Thus, there exists no registered Internet entity from which an OID for the ISO/CCITT NAME BINDING template can be derived. One solution is to use the object class OID to register name bindings under an special registration arc {cmipsnmpProxyNB} which is different from the arc used to register notifications, object classes and attributes. The following procedure is recommended for registration of name bindings for an object class to be inserted into the naming hierarchy, i.e., the subordinate object class:

- Replace the {cmipsnmpProxyIMIB} portion of the ISO/CCITT defined OID for the object class, derived using the procedures of 2.1.1, with {cmipsnmpProxyNB <sub-dentifier>}. The integer <sub-identifier> is used to distinguish possible multiple name bindings for the same object class. Sub-identifier values shall be greater than or equal to 1.

2.2 Naming Procedures

ISO/CCITT object instances are identified by specifying read-only attributes within the object class as naming attributes. The naming attribute is used to form the relative distinguished name of the object instance. The sequence of relative distinguished names that trace the path in the naming hierarchy from the root to the object forms a distinguished name that uniquely identifies the object instance.

2.2.1 Naming Attribute

The conversion of Internet MIBs into ISO/CCITT GDMO MIBs requires that naming attributes be defined and registered for each ISO/CCITT object class.

This paper specifies a generic naming attribute, and the conventions for its value definition, to facilitate automated generation of naming attributes for object classes derived from Internet MIBs. This generic naming attribute is applicable to all ISO/CCITT object classes derived from Internet defined MIBs.

The generic naming attribute, "internetClassId", is of ASN.1 type OBJECT IDENTIFIER with the following conventions for its value, as per the conventions in [RFC1212]:

- For ISO/CCITT object classes that may have only a single instance per managed system, the value is the OID for the object class concatenated with a "0". Object classes derived from the Internet TCP, UDP, and IP groups are examples of such object classes.

- For object classes that may have multiple instances per managed system, such as table entries, the value is the concatenation of the ISO/CCITT object class OID and the Internet object instance identification (an OID) of the form specified for the table entry instance identification in the original Internet MIB definition. The Internet object instance identification is the the concatenation of the values of the Internet OBJECT-TYPE(s) identified in the conceptual table entry OBJECT-TYPE INDEX clause. If an SNMP-2 AUGMENTS clause is present, the instance identification is the concatenation of the values of the OBJECT-TYPE(s) identifed in the INDEX clause of the table entry referenced in the value of the AUGMENTS clause.

  The formats for naming attributes of table entries are compatible with instance identification conventions used by the Internet, thereby facilitating the automated conversion of Internet MIBs into ISO/CCITT SMI format and the name mapping required for proxy management.

2.2.2 ISO/CCITT-Internet Proxy Naming Tree

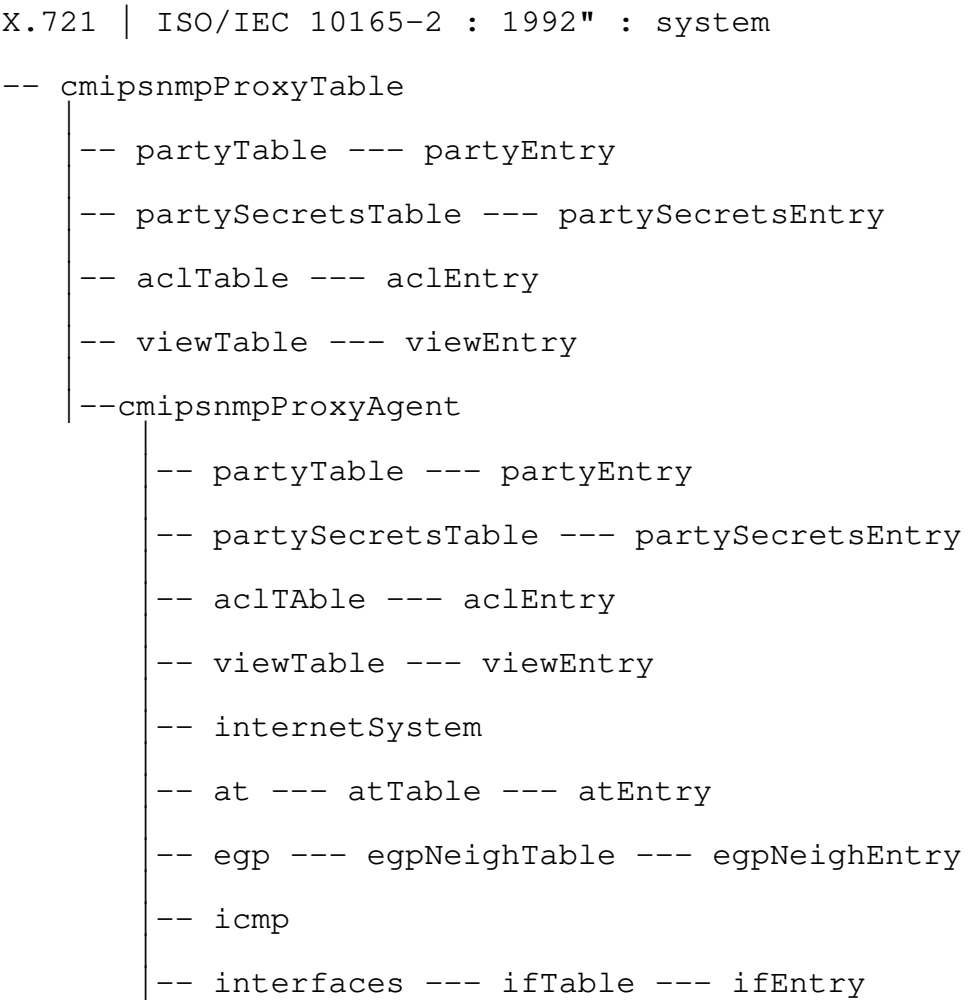
The ISO/CCITT SMI requires that managed object instances (conventionally just called managed objects) be bound into a naming hierarchy, or tree, for purposes of naming. This hierarchy is often called the containment hierarchy. The binding must specify for each managed object class: the object class which is superior to it in the containment hierarchy; and a naming attribute in the object class that is used to distinguish instances of the object class at a given level in the hierarchy. The name binding is specified after the object class has been defined. Multiple name bindings may be defined and registered that locate the object instances at different places in the naming hierarchy.

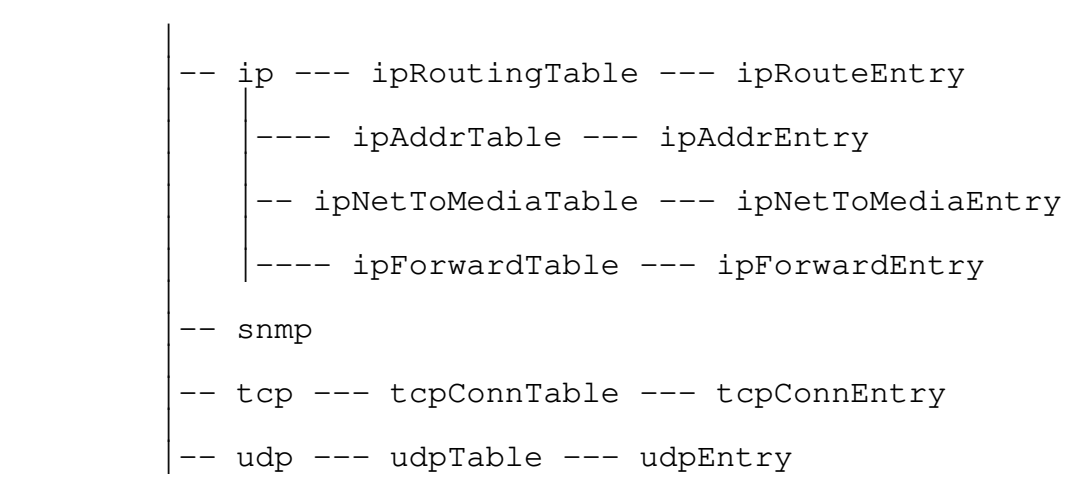
This memo defines a subtree of the naming hierarchy for the ISO/CCITT-Internet Proxy MIB to be used in ISO/CCITT-Internet proxies. The subtree is rooted at the "cmipsnmpProxyTable" object defined in 4. Each entry in the table, called "cmipsnmpProxyAgent", corresponds to an SNMP agent in a managed system. MIB elements specific to each agent are subtrees of a "cmipsnmpProxyAgent" instance.

One name binding for the "cmipsnmpProxyTable" object is specified which binds it to the ISO/CCITT "system" object class defined in [ISO10165-2]. Other name bindings may be specified.

The ISO/CCITT-Internet Proxy MIB is organized using the conventions of tables and table entries to optimize the benefits of CMIS scoping. While this is not required by the ISO/CCITT SMI or CMIS, it minimizes the number of objects to which filtering must be applied, or the number of objects that are returned if filtering is not applied with scoping.

A Naming Tree diagram for the ISO/CCITT-Internet Proxy MIB defined in this memo, the object classesdefined in the Party MIB [IIMCPARTY], and the MIB-II [IIMCMIB-II] is illustrated below. Although the ISO/CCITT "system" object class is shown to be the root of the tree, other object classes may be used. Object classes derived from groups in other Internet defined MIBs, that are specific to an SNMP agent, are bound to the "cmipsnmpProxyAgent" managed object, or to other objects within the cmipsnmpProxyAgent sub-tree as appropriate.





2.2.3 Distinguished Names

The distinguished name (DN) of a managed object consists of a sequence of relative distinguished names (RDN), one for each managed object on the naming path from the root to the managed object. Each relative distinguished name contains exactly one attribute, the "naming" attribute for the corresponding class, as specified by a NAME BINDING template. This DN is used as the CMIP ManagedObjectInstance parameter for identifying managed objects.

For example, a distinguished name designating a particular routing table entry (of class ipRouteEntry) might be

```
{
          O O O
                                -- higher level object RDNs
{internetClassId = {cmipsnmpProxyPMIB 1 0}}
                                -- cmipsnmpProxyTable
{cmipsnmpProxyAgentId = "troi.mitre.org" }
                                -- cmipsnmpProxyAgent
{internetClassId = {cmipsnmpProxyIMIB 2 1 4 0}}
                                -- ip
{internetClassId = {cmipsnmpProxyIMIB 2 1 4 21 0}}
                                -- ipRouteTable
{internetClassId = {cmipsnmpProxyIMIB 2 1 4 21 129 83 2 17}}
                                -- ipRouteEntry
}
```

The "internetClassId" naming attribute definition, to be used for object classes derived from the Internet MIBs, can be found in the ISO/CCITT-Internet Proxy MIB defined in 4 of this memo.

2.3 OID Translation

The procedures required to translate between Internet registered OIDs and names, and ISO/CCITT registered attribute and class OIDs are described in this section.

2.3.1 OID/Name Translation ISO/CCITT to Internet

The general procedure for deriving ISO/CCITT registered OIDs from Internet OIDs was explained in 2.1, and the structure of the naming attribute value was explained in 2.2. This paragraph explains how the information used in an ISO/CCITT class OID, attribute OID, and the naming attribute value may be used to create the identifier for naming Internet objects.

The following definitions apply: ((c) and (a) refer to class and attribute, respectively)

From 2.1,

```
{classOID}     ::= {cmipsnmpProxyIMIB
                    <internetEntityId>(c)}
```

For example

```
ipRouteTable ::= {cmipsnmpProxyIMIB 2 1 4 21 }
where <internetEntityId>(c) ::= 2 1 4 21.

{attributeOID}  ::= {cmipsnmpProxyIMIB
                     <internetEntityId>(a)}
```

For example,

```
ipRouteNextHop ::= {cmipsnmpProxyIMIB 2 1 4 21 1 7}
where <internetEntityId>(a) ::= 2 1 4 21 1 7.
```

From 2.2, the naming attribute value contains the OID formed as, (the "( )" indicates "contents of"):

```
(naming attribute)  ::=  {cmipsnmpProxyIMIB
                          <internetEntityId>(c)
                          <internet instanceId>}
```

where the <internet instanceId>, the OID created uniquely for each Internet object instance, is "0" for object classes that may only have a single instance.

For example, the naming attribute value for the instance of ipRouteEntry specific to IP address 129 83 2 17 is {cmipsnmpProxyIMIB 2 1 4 21 1 129 83 2 17}, where <internetEntityId>(c) ::= 2 1 4 21 1, and <internet instanceId> ::= 129 83 2 17.

The Internet uses the following convention to uniquely identify an Internet object instance:

```
{internet object name}::= {<internet> or <organization>
                           <internetEntityId>(a)
```

<internet instanceId>}

For example, the internet object name for ipRouteNextHop corresponding to IP address 129 83 2 17 is {internet 2 1 4 21 1 7 129 83 2 17}, where <internetEntityId>(a) ::= 2 1 4 21 1 7, <internet instanceId> ::=129 83 2 17.

Therefore, given the contents of the naming attribute for the ISO/CCITT object instance being accessed the <internet InstanceId> and <internet> or <organization> are extracted. Given the attributeOID for the attribute being operated upon, the <internetEntityId>(a) is extracted. The {internet object name} is then formed from the results.

For example, assume that a CMIS request is issued specifying a distinguished name for an ipRouteEntry managed object as illustrated in 2.2.3; and an attribute for ipRouteEntry, say ipRouteNextHop. The ipRouteNextHop attribute has been assigned the identifier {cmipsnmpProxyIMIB 2 1 4 21 1 7} in the MIB defined in [IIMCMIB-II]. Therefore, the ipRouteNextHop attribute identifier would first have to be translated into the corresponding Internet identifier {internet 2 1 4 21 1 7} by substituting "internet" for "cmipsnmpProxyIMIB". Then, from the RDN value for the ipRouteEntry extract the <internet instanceId> {129 83 2 17}. Finally the Internet identification for this piece of management information would be constructed according to RFC1213 as {ipRouteNextHop 129 83 2 17}, or equivalently, {internet 2 1 4 21 1 7 129 83 2 17}. The system in which this information resides is identified in the "cmipsnmpProxyAgent" RDN as "troi.mitre.org."

2.3.2 OID/Name Translation Internet to ISO/CCITT

Given the definitions shown in 2.3.1, the ISO/CCITT {classOID}, {attributeOID}, and distinguished name can be derived from Internet OIDs and names.

- The cmipsnmpProxyIMIB value is known.

- The <internetEntityId>(a) value is extracted from the {internet object name}, and combined with the cmipsnmpProxyIMIB value to form the {attributeOID}.

- The {classOID} can be determined by finding the ISO/CCITT object class to which the attribute identified as {cmipsnmpProxyIMIB <internetEntityId>(a) } belongs.

NOTE: Avoid the temptation to derive the {classOID} from the {attributeOID} by assuming that the {attributeOID} ={classOID <integer>}. Sometimes post-processing procedures may combine attributes from multiple classes into a single class. See 3.3 for a discussion on this issue.

2.4 Inheritance for Object Classes

The "top" class defined by [ISO10165-2] is the ultimate superior in the ISO/CCITT inheritance hierarchy. The class "top" contains attributes which are inherited by all managed object classes that are defined using the ISO/CCITT SMI and GDMO templates.

Not all attributes of "top" need to be instantianted in any single managed object. All objects should instantiate the mandatory "objectClass", and "nameBindings" attributes. If multiple packages may apply, an object should instantiate the "packages" attribute.

2.5 Reference Labels for Derived Entities

The labels used to reference Internet entities (groups, objects, traps) should be used to reference the corresponding templates for the derived ISO/CCITT entity (object class, attribute, notification).

3. Internet to ISO/CCITT MIB Translation Procedures

The procedures for translating Internet SMI MIBs into ISO/CCITT SMI MIBs consist of pre-translation procedures, GDMO template translation procedures, and post-translation procedures. Many of the procedures are subject to automation; some are not. Comments are provided concerning possible aids to automation; however, it is not the intent of this memo to provide automated translation algorithms.

3.1 Pre-translation Procedures

Pre-translation procedures are outlined below. The rationale for steps (a) - (e) is that the ISO/CCITT object classes and associated attributes must be identified. The rationale for steps (f) - (g) is that all ASN.1 syntax references in templates must be an ASN.1 Externaltypereference or Externalvaluereference, i.e., must reference a label that appears on the left side of an ASN.1 construct within some ASN.1 module that appears in the document that defines the derived MIB. Internet MIBs often reference basic ASN.1 constructs, such as INTEGER and OCTET STRING, which must be converted into an Externaltypereference. Default values must reference an Externalvaluereference.

(a) Identify Internet groups and OBJECT-TYPEs associated with each group. For SNMP-2 defined MIBs, the OBJECT-GROUP macro includes this information. For SNMP defined MIBs, the group may be identified manually and then the members of the group identified by the fact that their

OIDs contain the group object identifier. For SNMP defined MIBs, procedure (c) must be followed.

(b) Identify conceptual table OBJECT-TYPEs, conceptual table entry (row) OBJECT-TYPEs associated with each table, and columnar OBJECT-TYPEs associated with each conceptual table entry.

Note: For SNMP-2 defined MIBs, the MAX-ACCESS clause of the OBJECT-TYPES macro will have a value of 'not-accessible' and the convention often used is to include the word "Table" or "Entry" in the macro label. Once a table has been identified the corresponding entry OBJECT-TYPE macro can be identified via its registered object identifier through the convention of appending a 1 to the table object identifier. SNMP defined MIBs are not so consistent in their use of "not-accessible"; however, the other conventions apply.

Note: For SNMP-2 defined MIBs, tables may be defined with table entries that augment (SNMP-2 AUGMENT clause) entries for an existing table. The table object classes derived from such tables will be deleted from the ISO/CCITT MIB during post-translation (see 3.3.3). The table entry object class for the deleted table will be bound to the table entry object class that corresponds to the reference in the AUGMENTS clause.

(c) For each group, the OBJECT-TYPEs not identified in procedure (b) and not having an ACCESS clause value of "not-accessible" are identified for translation into attributes of an ISO/CCITT object class.

(d) For each conceptual table entry OBJECT-TYPE, the columnar OBJECT-TYPEs associated with the table entry and not having an ACCESS clause value of "not-accessible" are identified for translation into ISO/CCITT attributes of an ISO/CCITT object class.

(e) For each conceptual table, an ISO/CCITT object class is created that contains the generic naming attribute "internetClassId". OBJECT-TYPES that are directly associated with the table become attributes of that table.

(f) Create an ASN.1 module. For each OBJECT-TYPE that is to be translated into an ISO/CCITT attribute, check the value of the SYNTAX clause, and if it is not one of the Internet attribute types defined by the SNMP or SNMP-2 SMI (Counter, IpAddress, Gauge, TimeTicks, Opaque, Counter32, Gauge32, Counter64, NsapAddress), or defined using an SNMP-2 TEXTUAL-CONVENTION macro, then do one of the following:

i) If the value is not an Externaltypereference: create an Externaltypereference for the value in

the SYNTAX clause and put it into the ASN.1 module.

ii) If the value is an Externaltypereference: put the Externaltypereference value into the ASN.1 module. The ASN.1 module is used for the GDMO template translations.

g) Create an Externalvaluereference which has the type indicated by the OBJECT-TYPE SYNTAX clause and is assigned the value of the DEFVAL clause.

For convenience, an ASN.1 module of common definitions for Externaltypereferences of the basic ASN.1 types included in the SNMP SMI and SNMP-2 SMI (e.g., INTEGER, OCTET STRING) is defined in 4.5. The Externaltypereferences in this module must either be imported into a local ASN.1 module within the document that defines the derived MIB, or appropriate equivalent references need to be declared within the local ASN.1 module.

3.2 GDMO Translation Procedures

Readers of this memo are assumed to have familiarity with the GDMO templates and their format. The templates presented here should be considered exemplar, and not definitive.

The GDMO templates in this paragraph contain elements indicated by "< x >", where "x" is to be interpreted and the result substituted into the template.

The "||" symbol indicates concatenation of elements.

Adjacent elements that are not concatenated should be separated by at least one space.

The "[ ]" symbols surrounding a construct indicate that it may be present or absent in any particular instance of the template.

The "[ ]*" symbols surrounding a construct indicate that it may be present or absent in any particular instance of the template an undefined number of times.

An "|" symbol is used to indicate that a choice must be made between alternate constructs.

The "!" symbol is used to delimit text strings in the templates. Other delimiters may be used, as specified by the GDMO. The delimiter may not be used in the text string unless it is "doubled", e.g., "!!".

Elements that are defined in one template are not repeated for other templates unless changes are made to its interpretation.

Note: other SNMP-2 SMI macro clauses contain textual or other information that may be inserted into the BEHAVIOUR clause of the an ISO/CCITT template, e.g., UNITS, REFERENCE. Provisions for including information in these macro clauses are not explicitly described in the translation procedures below, however the contents of these clauses should be included in the BEHAVIOUR clause.

The Internet SNMP-2 SMI also includes macros for specifying compliance with the MIBs. These macros correspond to ISO/CCITT managed object conformance statements (MOCS), and are not addressed here.

3.2.1 Translation of Groups

Internet groups may be translated to ISO/CCITT managed object classes by filling in the following GDMO template:

```
<group label>  MANAGED OBJECT CLASS
          DERIVED FROM
              "Rec. X.721 | ISO/IEC 10165-2 : 1992" :top
     CHARACTERIZED BY
              <group label> || "Pkg" PACKAGE
          [BEHAVIOUR
           <group label> || "PkgBehaviour" BEHAVIOUR
           DEFINED AS !<optional textual definition>!;;]
          ATTRIBUTES
           internetClassId     GET
           ["," <OBJECT-TYPE label n>
           <OBJECT-TYPE label n ACCESS clause translation>
           [DEFAULT VALUE  <DEFVAL clause translation>]]*;
     [NOTIFICATIONS  <notification label>
                ["," <notification label>]*;];;
REGISTERED AS  { {cmipsnmpProxyIMIB} <internetEntityId>(c)
};
```

The following definitions apply:

<group label> - The label associated with the Internet group.

<optional textual definition> - Any textual description that is applicable to the resource modelled by this group, and the resource's management behaviour. Text in the SNMP-2 DESCRIPTION clause of the OBJECT-GROUP macro may be used.

<OBJECT-TYPE label n> - The label of an Internet OBJECT-TYPE which is included in the group and does not represent a conceptual table, a conceptual table entry,

or an OBJECT-TYPE included in a conceptual table entry. These become attributes of the object class.

<OBJECT-TYPE label n ACCESS clause translation> - The mapping of the ACCESS (or SNMP-2 MAX-ACCESS) clause value as defined below (multi-valued attributes are not permitted in the Internet SMI):

OBJECT-TYPE ACCESS Clause Value*	**ISO/CCITT**
read-only	GET
read-write	GET-REPLACE
write-only	REPLACE
read-create	GET-REPLACE

* OBJECT-TYPEs with an ACCESS clause value of 'not-accessible' will not become ISO/CCITT attributes.

<DEFVAL clause translation> - The value of the DEFVAL clause in the form of an Externalvaluereference, i.e., <module-name>.<value-name>, where the module-name is the name of an ASN.1 module within the document in which this object class is defined, and the value-name is the label assigned to the ASN.1 value definition contained in the DEFVAL clause. Where it is necessary to refer to the label of a definition contained in another document, this may be achieved by means of a local ASN.1 module which makes use of the ASN.1 IMPORTS mechanism to import the appropriate value definition.

3.2.2 Translation of Table Objects

Internet conceptual table objects may be translated to ISO/CCITT managed object classes by filling in the following GDMO template:

```
<OBJECT-TYPE label>  MANAGED OBJECT CLASS
     DERIVED FROM
             "Rec. X.721 | ISO/IEC 10165-2 : 1992" :top;
     CHARACTERIZED BY
              <OBJECT-TYPE label> || "Pkg" PACKAGE
     [BEHAVIOUR
          <OBJECT-TYPE label> || "PkgBehaviour" BEHAVIOUR
          DEFINED AS
               !<OBJECT-TYPE DESCRIPTION clause text>!;;]
     ATTRIBUTES
          internetClassId     GET
          ["," <OBJECT-TYPE label n>
          <OBJECT-TYPE label n ACCESS clause translation>
          [DEFAULT VALUE  <DEFVAL clause translation>]]*;
     [NOTIFICATIONS  <notification label>
                ["," <notification label>]*;];;
```

```
REGISTERED AS  { {cmipsnmpProxyIMIB} <internetEntityId>(c)
};
```

The following definitions apply:

<OBJECT-TYPE label> - The label associated with the OBJECT-TYPE macro.

<OBJECT-TYPE DESCRIPTION clause text> - The text in the DESCRIPTION clause.

3.2.3 Translation of Table Entry Objects

Internet conceptual table entry objects may be translated to ISO/CCITT managed object classes by filling in the following GDMO template:

```
<OBJECT-TYPE label>  MANAGED OBJECT CLASS
         DERIVED FROM
              "Rec. X.721 | ISO/IEC 10165-2 : 1992" :top;
         CHARACTERIZED BY
               <OBJECT-TYPE label> || "Pkg" PACKAGE
         BEHAVIOUR
          <OBJECT-TYPE label> || "PkgBehaviour" BEHAVIOUR
          DEFINED AS
     !<OBJECT-TYPE DESCRIPTION/INDEX/STATUS clause text>!;;
         ATTRIBUTES
          internetClassId     GET
          ["," <OBJECT-TYPE label n>
          <OBJECT-TYPE label n ACCESS clause translation>
          [DEFAULT VALUE  <DEFVAL clause translation>]]*;
     [NOTIFICATIONS  <notification label>
                  ["," <notification label>]*;];;
REGISTERED AS {{cmipsnmpProxyIMIB} <internetEntityId>(c) };
```

The following definitions apply:

<OBJECT-TYPE label> - The label of an Internet OBJECT-TYPE which represents a conceptual table entry.

<OBJECT-TYPE label n> - The label of an Internet OBJECT-TYPE which represents an OBJECT-TYPE included in a conceptual table entry. These become attributes of the table entry managed object.

<OBJECT-TYPE DESCRIPTION/INDEX/STATUS clause text> - The text in the DESCRIPTION clause and the INDEX clause including the keyword INDEX. Note: in the post-translation phase, information about the status variable used for creation and deletion will be inserted.

Note: Table object classes that contain table entry object classes which were derived from OBJECT-TYPES with an

AUGMENTS clause will be deleted in the post-translation phase according to 3.3.3.

3.2.4 Translation of Other OBJECT-TYPES

Other Internet OBJECT-TYPEs are translated into ISO/CCITT attributes by filling in the following GDMO template:

```
<OBJECT-TYPE label> ATTRIBUTE
     [WITH ATTRIBUTE SYNTAX
         <module identification>|| "." ||
                          <SYNTAX clause translation 1>
      [MATCHES FOR <SYNTAX clause type matching rules>;] ]
   |  [DERIVED FROM <SYNTAX clause translation 2>]
      [BEHAVIOUR
          <OBJECT-TYPE label>  || "Behaviour" BEHAVIOUR
          DEFINED AS  !<DESCRIPTION clause text>!;;]
REGISTERED AS {{cmipsnmpProxyIMIB} <internetEntityId>(a)};
```

The following definitions apply:

> <module identification> - The label of the ASN.1 module that contains the ASN.1 syntax being referenced. The module must appear in the document that defines the translated MIB.

ISO/CCITT have the concept of a generic "attribute type", which is a defined type that can be used in the definition of specific attributes. Attribute types have a defined syntax, generic semantics, and matching rules. For example, counter and gauge are attribute types.

The SNMP-2 SMI has a similar concept embodied in the TEXTUAL-CONVENTIONS macro, which allows the definition of "Internet attribute types" with associated syntax and semantics. See 3.2.6 for translation procedures associated with the TEXTUAL CONVENTIONS macro.

Attributes of the defined SNMP types (Counter, IpAddress, Gauge, TimeTicks, Opaque, Counter32, Gauge32, Counter64, NsapAddress) should inherit the semantics associated with the type - not just the syntax. As such, they could have been defined as Internet attribute types using a TEXTUAL CONVENTIONS macro. See 4.3 for the conversion of these types into ISO/CCITT attribute types. In addition, 4.3 contains ISO/CCITT attribute type definitions derived from [SMPTC].

Attribute templates derived from OBJECT-TYPE macros that specify these Internet attribute types in their SYNTAX clause should contain the DERIVED FROM clause. Attribute templates derived from other ASN.1 types should contain the WITH SYNTAX and MATCHING RULES clauses.

<SYNTAX clause translation 1> - Translation of the SYNTAX clause into a valid type reference name using the ASN.1 Externaltypereference notation as GDMO requires.

<SYNTAX clause type matching rules> - The matching rules for use in CMIS filtering operations.

Note: This normally is a manual process; however matching rules that generally apply for some specific types are provided in the table below. These rules also to Externaltypereferences that reference the syntax type. These matching rules may be applied by automated mechanisms and then examined in the post-translation.

Syntax Type	Matching Rules
INTEGER	EQUALITY, ORDERING
OCTET STRING	EQUALITY, ORDERING, SUBSTRINGS
BIT STRING	EQUALITY
OBJECT IDENTIFIER	EQUALITY, ORDERING
SEQUENCE	EQUALITY
Counter (x)	EQUALITY, ORDERING *
Gauge (x)	EQUALITY, ORDERING *
IpAddress	EQUALITY, ORDERING SUBSTRINGS*
NsapAddress	EQUALITY, ORDERING SUBSTRINGS*
TimeTicks	EQUALITY, ORDERING SUBSTRINGS*
Opaque	EQUALITY, ORDERING*

The * indicates the matching rules that are inherited from the ISO/CCITT attribute type as defined in 4.3.

The (x) indicates counters and gauges of differing sizes.

<SYNTAX clause translation 2> - Attributes of the defined SNMP/SNMP-2 types (Counter, IpAddress, Gauge, TimeTicks, Opaque, Counter32, Gauge32, Counter64, NsapAddress),and Internet attribute types defined using the SNMP-2 TEXTUAL CONVENTIONS macro, should be treated as ISO/CCITT attribute types. Specific attributes are derived from these types.

The following table indicates the translations required for Internet attribute types as defined either in this memo or Internet documents. ISO/CCITT attribute types for Internet attribute types types are defined in 4.3.

Syntax Type	Substituted Value
AutonomousType	"IIMCMIBTRANS" :autonomousType
Counter	"IIMCMIBTRANS" :counter32
Counter32	"IIMCMIBTRANS" :counter32
Counter64	"IIMCMIBTRANS" :counter64
DisplayString	"IIMCMIBTRANS" :displayString
Gauge	"IIMCMIBTRANS" :gauge32
Gauge32	"IIMCMIBTRANS" :gauge32
InstancePointer	"IIMCMIBTRANS" :instancePointer
IpAddress	"IIMCMIBTRANS" :ipAddress
MacAddress	"IIMCMIBTRANS" :macAddress
NsapAddress	"IIMCMIBTRANS" :nsapAddress
Opaque	"IIMCMIBTRANS" :opaque
PhysAddress	"IIMCMIBTRANS" :physAddress
RowStatus	"IIMCMIBTRANS" :rowStatus
TestAndIncrement	"IIMCMIBTRANS" :testAndIncrement
TimeInterval	"IIMCMIBTRANS" :timeInterval
TimeStamp	"IIMCMIBTRANS" :timeStamp
TimeTicks	"IIMCMIBTRANS" :timeTicks
TruthValue	"IIMCMIBTRANS" :truthValue

3.2.5 Translation of Traps

The Concise MIB Definitions [RFC1212] for SNMP does not contain a macro for representing traps since, in SNMP, traps were considered part of the protocol and not part of the MIB. A subsequent attempt was made to correct this problem in [RFC1215] by defining a TRAP-TYPE macro. The SNMP-2 SMI [SMPSMI] defines a TRAP-DEFINITION macro and its mapping onto an SNMP-2 PDU. The memo on SNMP/SNMP-2 Coexistance [SMPCOEX] defines a mapping between SNMP and SNMP-2 trap PDUs. Therefore, by inference, there exists a mapping of SNMP trap PDUs into an SNMP-2 TRAP-DEFINITION macro. This memo assumes that SNMP trap PDUs will be translated into SNMP-2 trap PDUs, and that by inference the mapping into a SNMP-2 TRAP-DEFINITION exists.

The translation of Internet traps, as defined for SNMP Trap PDUs and the SNMP-2 TRAP-DEFINITION macro, may be translated to ISO/CCITT notifications by filling in the following GDMO template:

```
<trap label>  NOTIFICATION
          BEHAVIOUR
          "<trap label>" || "Behaviour" BEHAVIOUR
          DEFINED AS
          !<DESCRIPTION and
               OBJECTS/VARIABLES clause translation>!;;
               WITH INFORMATION SYNTAX
               <module identification>|| "." ||
                         "MonitoredAttributes";
               -- MonitoredAttributes must be imported
```

```
                -- from Attribute-ASN1Module, defined in
                -- Rec. X.721 | ISO/IEC 10165-2 : 1992,
                -- into the ASN.1 module defined in the
                -- document that contains the MIB
                -- translation
           AND ATTRIBUTE IDS
                monitoredAttributes
           "Rec. X.721 | ISO/IEC 10165-2 : 1992"
                                :monitoredAttributes;
REGISTERED AS {<notificationOID>};
```

<trap label> - The label of the TRAP-DEFINITION (or TRAP-TYPE) macro, or one assigned by local means for SNMP enterprise specific traps.

<DESCRIPTION and OBJECTS/VARIABLES clause translation>- The value of the TRAP-DEFINITION OBJECTS (or TRAP-TYPE VARIABLES) clause indicates the names of the attributes to be inserted into the monitoredAttributes field of the trap. A sentence indicating this fact should be inserted into the text of the DESCRIPTION clause and the result becomes the text for the template BEHAVIOUR clause. The text describing behaviour for SNMP traps and the attributes to be inserted into the monitoredAttributes field for traps not specified using a macro is not defined.

The IpAddress of the SNMP agent in the SNMP Trap PDU is missing from the SNMP-2 TRAP-DEFINITION macro. The address may be inferred from the managed system's domain name in the "cmipsnmpProxyEntryId" RDN of the DN contained in the ManagedObjectInstance field of the CMIP event report that carries the notification.

Similarly, the SNMP PDU "generic-trap", and "specific-trap" fields are missing from the SNMP-2 trap PDU and SNMP-2 TRAP-DEFINITION macro. The information contained in these fields are combined into the registered object identifier for the SNMP-2 TRAP-DEFINITION macro. This object identifier corresponds to the smpTrapOID object defined in [SMPMIB].

<notificationOID> - The OID for the notification as derived in 2.1.2.

3.2.6 Translation of Internet Attribute Types

ISO/CCITT has the concept of a generic "attribute type", which is a defined type that can be used in the definition of specific attributes. Attribute types have a defined syntax, generic semantics, and matching rules. For example, counter and gauge are attribute types.

The SNMP-2 SMI has a similar concept embodied in the TEXTUAL-CONVENTION macro, which allows the definition of

"Internet attribute types" with associated syntax and semantics.

Attributes of the defined SNMP types (Counter, IpAddress, Gauge, TimeTicks, Opaque, Counter32, Gauge32, Counter64, NsapAddress) should inherit the semantics associated with the type - not just the syntax. As such, they could have been defined as Internet attribute types using a TEXTUAL-CONVENTION macro.

ISO/CCITT attribute types are defined using the ATTRIBUTE template, without the REGISTERED AS clause.

```
<Internet attribute type label> ATTRIBUTE
          [WITH ATTRIBUTE SYNTAX
           <module identification>|| "." ||
                    <SYNTAX clause translation 1>
          [MATCHES FOR
               <SYNTAX clause type matching rules>;] ]
      |   [DERIVED FROM <SYNTAX clause translation 2>]
          [BEHAVIOUR
          <Internet attribute type label>   || "Behaviour"
               BEHAVIOUR
               DEFINED AS  !<DESCRIPTION clause text>!;;]
```

The following definitions apply:

> <Internet attribute type label> - The label associated with the TEXTUAL-CONVENTION macro, or with the generic type that could have been defined using that macro.

Attribute templates derived from OBJECT-TYPE macros that specify Internet attribute types in their SYNTAX clause should specify the corresponding ISO/CCITT attribute types in their DERIVED FROM clause.

3.3 Post-translation Procedures

Post-translation procedures generally includes manual checking of the BEHAVIOUR clause text for proper behaviour definitions, the addition of information concerning variables for creation and deletion, the assignment of notifications to an appropriate ISO/CCITT object class, and the creation of name bindings for placing the derived ISO/CCITT objects into the containment hierarchy. However, sometimes the procedures are not so straightforward.

Post-translation procedures may result in deletion of some ISO/CCITT MIB elements derived from the procedures described in 2.2. For example, if the table object class contains entries that were derived from Internet OBJECT-TYPES that contained an AUGMENTS clause, then the table is deleted from the MIB.

Post-translation procedures may also result in the combination multiple ISO/CCITT MIB object classes to form one class. The multiple object classes should perhaps belong to a single object class, but were created as separate object classes due to specific SNMP SMI requirements. The resulting multiple object classes may be constrained to synchronize their behaviour, e.g., to synchronize creation and deletion procedures. However, when viewed from an ISO/CCITT context, the rationale for using multiple objects may disappear, or ISO/CCITT SMI and protocol constraints may prohibit their use. See [IIMCPARTY] for an example of this situation.

3.3.1 Post-translation of BEHAVIOUR Cause

The SNMP and SNMP-2 text descriptions often contain SNMP/SNMP-2 protocol, or SMI, relevant information that is inappropriate for an ISO/CCITT object class or attribute; such text should be removed or properly modified.

A reference to the Internet MIB entity from which the ISO/CCITT entity was derived should be added.

Descriptions of circumstances that cause the generation of specific notifications should be included.

Information should be added that is relevant to attributes for instance identification and creation/deletion status. Such information should be added in a way that facilitates automated parsing of the BEHAVIOUR clause. The use of keywords is recommended.

Multiple instance objects should be indicated by the keywordS "MULTIPLE INSTANCE" in the text.

The text should include relevant information related to instance identification in the form of the full INDEX or AUGMENTS clause.

The status variable, and its value for deletion, must be described for table entries that may be created and deleted by management action. The following format should be used:

```
STATUSVAR      ::= <label>
STATUSDELETE   ::= <value>
```

where <label> is the label of the attribute used to indicate deletion of an entry, and <value> is the value which indicates deletion. The value specification should be consistent with the type of the status attribute.

3.3.2 Post-translation of Notifications

The ISO/CCITT SMI models notifications as being emitted by specific managed objects. As a consequence, notifications must be assigned to appropriate object classes at the time the object class is defined. New notifications cannot be added to an object class without changing its registration.

The fact that the ISO/CCITT SMI models notifications as being emitted by specific managed objects implies that notifications may only contain information known to the managed object. However, the Internet SMI has no explicit concept of traps being associated with an object. They may contain information related to multiple internet objects. Consequently, some notifications may be derived from SNMP traps that contain variables not affiliated with the same derived ISO/CCITT object class. This memo allows variables to be placed into a notification even if they are not attributes of the object class from which the notification is emitted.

The following procedures should be applied:

1) identify the object class that corresponds to the resource to which the notification applies.

2) insert the notification into the NOTIFICATIONS clause of the object class.

3.3.3 Creation of NAME BINDING Templates

The ISO/CCITT name bindings for object classes to be bound into the naming hierarchy described in 2.2.2 are created by filling in the GDMO template defined below.

```
<object class label>-NB  NAME BINDING
              SUBORDINATE OBJECT CLASS
               <object class label>  AND SUBCLASSES;
              NAMED BY SUPERIOR OBJECT CLASS
               <superior object class label> AND SUBCLASSES;
              WITH ATTRIBUTE internetClassId;
              [CREATE [<create modifier>] ;]
              [DELETE [<delete-modifier>];]
REGISTERED AS { <name binding OID>};
```

<object class label> - the reference label of the object class to which the name binding applies.

<superior object class label> - the reference label of the superior object class in the naming hierarchy.

Object classes derived from groups normally have the "cmipsnmpProxyAgent" object class as their superior.

Table object classes, derived from conceptual tables, have the object class derived from the group in which

they were defined as their superior. One way to determine the group is to use the structure of the OID for the table object, i.e., it contains the internet specific portion of the OID for the group. However, if the table object class contains entries that were derived from Internet OBJECT-TYPES that contained an AUGMENTS clause, then the table is deleted from the MIB. The reason for this is that the ISO/CCITT SMI prohibits adding attributes to an object class. The solution used in this memo is to make a table entry object class that augments another table entry the direct subordinate of the table entry object class being augmented. The table is no longer needed.

Table entry object classes, derived from conceptual table entries, have the corresponding table object class as their superior. One way to determine the table is to use the structure of the OID for the table entry object class, i.e., it contains the internet specific portion of the OID for the table.However, table entry object classes derived from OBJECT-TYPES that contain an AUGMENTS clause have the table entry object class derived from the OBJECT-TYPE referenced in the AUGMENTS clause as their superior.

The Internet SMI only allows the possibility of conceptual table entries being created and deleted. Many table entries are automatically created and deleted as a result of normal resource operation.

For SNMP-2 defined MIBs, if the table entry contains an OBJECT-TYPE that has a SYNTAX clause value of "RowStatus" and a MAX-ACCESS clause value of "read-create", then the table entry may be created and deleted.

For SNMP defined MIBs, the use of an OBJECT-TYPE within a table entry which indicates "RowStatus" is inconsistant. Usually, the text definition for the table entry may need to be consulted to determine if creation and deletion are allowed, and the columnar object and associated value which indicates deletion.

<create modifier> - the "WITH-AUTOMATIC-INSTANCE-NAMING" modifier should be part of the CREATE clause if the object class is being used for proxy. This allows the proxy, which is closer to the resource than the manager, to assign instance names. The "WITH-REFERENCE-OBJECT" may also be be used.

<delete-modifier> - The delete modifier for the DELETE clause should be "DELETES-CONTAINED-OBJECTS" if the table entry contains an object class added as a result of an SNMP-2 OBJECT-TYPE AUGMENTS clause.

<name binding OID> - As defined in 2.1.3.

4. ISO/CCITT-Internet Proxy MIB

The ISO/CCITT-Internet Proxy MIB defines a set of objects for specifying the information that is needed for both community-based and party-based SNMP management on a per SNMP agent basis.

The ISO/CCITT-Internet Proxy MIB defines the "cmipsnmpProxyTable" object class which contains entry object classes named "cmipsnmpProxyAgent". The "cmipsnmpProxyAgent" has information to identify SNMP agents and how they may be reached. Its naming attribute, which contains the administratively assigned name of the managed device where the SNMP agent is located, is used in the naming hierarchy to identify the SNMP managed device.

The definition of these two object classes and their attributes are defined in 4.1.

4.1 Object and Attribute Definitions

ISO/CCITT-Internet proxy MIB object classes and attributes are assigned OIDs under the {cmipsnmpProxyPMIB} arc. The Internet convention of registering attributes under the object class to which they belong is followed in this memo.

```
cmipsnmpProxyAgent MANAGED OBJECT CLASS
            DERIVED FROM
               "Rec. X.721 | ISO/IEC 10165-2 : 1992" :top;
            CHARACTERIZED BY
               cmipsnmpProxyAgentPkg PACKAGE
                   BEHAVIOUR
               cmipsnmpProxyAgentPkgBehaviour BEHAVIOUR
          DEFINED AS
          !This managed object class is used to represent an
          SNMP/SNMP-2 agent in the proxy MIB.  Each agent
          that the proxy manages is represented by an
          instance of this object class.

          The cmipsnmpProxyAgentId attribute contains the
          administratively assigned name of the managed
          system that contains the SNMP/SNMP-2 agent.
          Usually this is an Internet Domain Name.  The
          value is determined by the manager when the object
          is created.

          The proxyOID attribute contains the proxy arc
          used in the re-registration and derivation of
          OIDs.  The following sub-arcs are assigned:

               {<proxyOID> 1} - object classes, attributes
```

{<proxyOID> 2} - name bindings
{<proxyOID> 3} - notifications

The replaceByProxyOID attribute contains the "internet" OID or an organization specific OID. The portion of the OID for an internet defined object that corresponds to the replaceByProxyOID contents is replaced by {<proxyOID> 1} when translating internet object OIDs into ISO/CCITT OIDs, and {<proxyOID> 3} when translating internet trap OIDs into ISO/CCITT notification OIDs.

The managementProtocol attribute specifies the Internet management protocol used by the proxy to manage devices. It may be an OID indicating SNMP, SNMP-2, or some other protocol. This attribute is assigned a value (an OID) by the manager that is appropriate to the agent.

The supportedMIBs attribute contains the set of OIDs of registered MIBs that the agent supports. The manager may add or remove elements to/from this attribute.

The specificAccessControlUsed attribute indicates whether access control specific to an SNMP agent is none, uses Internet defined mechanisms and MIB elements, or uses ISO/CCITT defined mechanisms and MIB elements.

The accessControlEnforcement attribute indicates where access control is applied: SNMP agent, proxy, or both.

The operationalState attribute indicates the perceived state of the agent in the managed system:

The "enabled" state means that the agent is operational, as perceived by the proxy, i.e., it can be reached.

The "disabled" state means that the agent is not operational, as perceived by the proxy, i.e., it cannot be reached.

The administrativeState attribute is used to suspend and resume the proxy activity relative to the agent.

The "unlocked" state means that proxy must continue to perform, or resume performing, proxy activities on behalf of the agent.

```
                The "locked" state means that the proxy
                must not perform, or suspend performing,
            proxy activities on behalf of the agent.

            The authenticationFailure notification is
            emitted when the SNMP agent is the addressee of a
            protocol message that is not properly
            authenticated.

            The coldStart notification is emitted when the
            SNMP agent reinitializes itself such that its
            configuration may be altered.

            The warmStart notification is emitted when the
            SNMP agent reinitializes itself such that its
            configuration is not altered.

            This object class may have MULTIPLE INSTANCES.!;;
           ATTRIBUTES
            cmipsnmpProxyAgentId     GET,
            proxyOID                 GET-REPLACE,
            replacedByProxyOID       GET-REPLACE,
            managementProtocol       GET
                                     REPLACE-WITH-DEFAULT,
            supportedMIBs            GET-REPLACE
                                          ADD-REMOVE,
            specificAccessControlUsed GET-REPLACE
                                     DEFAULT VALUE
                                CmipsnmpProxyASN1.Zero,
            accessControlEnforcement GET-REPLACE,
            "Rec. X.721 | ISO/IEC 10165-2 : 1992":
                 administrativeState GET-REPLACE,
            "Rec. X.721 | ISO/IEC 10165-2 : 1992":
                 operationalState    GET;
        NOTIFICATIONS
             authenticationFailure,
             coldStart,
             warmStart;;;
REGISTERED AS  { cmipsnmpProxyPMIB 1 1};

cmipsnmpProxyTable MANAGED OBJECT CLASS
            DERIVED FROM
                 "Rec. X.721 | ISO/IEC 10165-2 : 1992" :top;
            CHARACTERIZED BY
                cmipsnmpProxyTablePkg PACKAGE
                     BEHAVIOUR
                 cmipsnmpProxyTablePkgBehaviour BEHAVIOUR
             DEFINED AS
             !This managed object class is used to contain
             objects that represent an SNMP/SNMP-2 agent in the
             proxy MIB.

             The globalAccessControlUsed attribute indicates
```

```
          whether access control global (uniformly applied)
          to all SNMP agents is none, uses Internet defined
          mechanisms and MIB elements, or uses ISO/CCITT
          defined mechanisms and MIB elements.!;;
         ATTRIBUTES
          internetClassId          GET,
          globalAccessControlUsed  GET-REPLACE
                                   DEFAULT VALUE
                                CmipsnmpProxyASN1.Zero;;;
REGISTERED AS  { cmipsnmpProxyPMIB 1};

accessControlEnforcement ATTRIBUTE
          WITH ATTRIBUTE SYNTAX
          CmipsnmpProxyASN1.AccessControlEnforcement;
               BEHAVIOUR
               accessControlEnforcementBehaviour BEHAVIOUR
          DEFINED AS
          !The accessControlEnforcement attribute indicates
          where access control is applied: SNMP agent,
          proxy, or both.!;;
          MATCHES FOR    EQUALITY, ORDERING;
REGISTERED AS {cmipsnmpProxyPMIB 1 1 7};

cmipsnmpProxyAgentId ATTRIBUTE
          WITH ATTRIBUTE SYNTAX
               CmipsnmpProxyASN1.CmipsnmpProxyAgentId;
             BEHAVIOUR
          cmipsnmpProxyAgentIdBehaviour BEHAVIOUR
          DEFINED AS
          !This is the naming attribute for the
          cmipsnmpProxyAgent object class. It contains
          the Internet Domain Name of the managed system
          that contains the SNMP/SNMP-2 agent.  The value is
          determined by the manager at the time the
          object is created.!;;
         MATCHES FOR     EQUALITY, SUBSTRINGS;
REGISTERED AS {cmipsnmpProxyPMIB 1 1 1};

globalAccessControlUsed ATTRIBUTE
          WITH ATTRIBUTE SYNTAX
          CmipsnmpProxyASN1.AccessControlUsed;
               BEHAVIOUR
               globalAccessControlUsedBehaviour BEHAVIOUR
          DEFINED AS
          !The globalAccessControlUsed attribute indicates
          whether access control global (uniformly applied)
          to all SNMP agents is none, uses Internet defined
          mechanisms and MIB elements, or uses ISO/CCITT
          defined mechanisms and MIB elements.!;;
          MATCHES FOR    EQUALITY, ORDERING;
REGISTERED AS {cmipsnmpProxyPMIB 1 3};

internetClassId ATTRIBUTE
           WITH ATTRIBUTE SYNTAX
```

```
                    CmipsnmpCommonDef.ObjectIdentifier;
                   BEHAVIOUR
               internetClassIdBehaviour BEHAVIOUR
          DEFINED AS
          !This is a generic naming attribute
          intended to be used for naming all object
          classes derived from Internet MIB translation.

          For ISO/CCITT object classes that may have only a
          single instance per managed system, the value
          is the OID for the object class concatenated with
          the sub-identifier "0".  Object classes derived
          from the Internet TCP, UDP, and IP groups are
          examples of such object classes.

          For object classes that may have multiple
          instances per managed system, such as table
          entries, the value is the concatenation of the
          ISO/CCITT object class OID and the Internet object
          instance identifier (an OID) of the form
          specified for the table entry instance
          identification in the original Internet MIB
          definition.!;;
        MATCHES FOR     EQUALITY;
REGISTERED AS {cmipsnmpProxyPMIB 1 2};

managementProtocol ATTRIBUTE
          WITH ATTRIBUTE SYNTAX
          CmipsnmpCommonDef.ObjectIdentifier;
               BEHAVIOUR
               managementProtocolBehaviour BEHAVIOUR
               DEFINED AS
               !This attributes specifies the internet
               management protocol used for proxy to
               managed devices.  It may be either SNMP or
               SNMP-2.!;;
          MATCHES FOR    EQUALITY, ORDERING;
REGISTERED AS {cmipsnmpProxyPMIB 1 1 4};

proxyOID ATTRIBUTE
          WITH ATTRIBUTE SYNTAX
          CmipsnmpCommonDef.ObjectIdentifier;
               BEHAVIOUR
               proxyOIDBehaviour BEHAVIOUR
               DEFINED AS
          !The proxyOID attribute contains the proxy arc
          used in the re-registration and derivation of
          OIDs.  The following sub-arcs are assigned:
               {<proxyOID> 1} - object classes, attributes
               {<proxyOID> 2} - name bindings
               {<proxyOID> 3} - notifications!;;
          MATCHES FOR    EQUALITY, ORDERING;
REGISTERED AS {cmipsnmpProxyPMIB 1 1 2};
```

```
replaceByProxyOID ATTRIBUTE
          WITH ATTRIBUTE SYNTAX
          CmipsnmpCommonDef.ObjectIdentifier;
               BEHAVIOUR
               replaceByProxyOIDBehaviour BEHAVIOUR
               DEFINED AS
          !The replaceByProxyOID attribute contains the
          "internet" OID or an organization specific OID.
          The portion of the OID for an internet defined
          object that corresponds to the replaceByProxyOID
          contents is replaced by {<proxyOID> 1} when
          translating internet object OIDs into ISO/CCITT
          OIDs, and {<proxyOID> 3} when translating internet
          trap OIDs into ISO/CCITT notification OIDs!;;
          MATCHES FOR    EQUALITY, ORDERING;
REGISTERED AS {cmipsnmpProxyPMIB 1 1 3};

specificAccessControlUsed ATTRIBUTE
          WITH ATTRIBUTE SYNTAX
          CmipsnmpProxyASN1.AccessControlUsed;
               BEHAVIOUR
               specificAccessControlUsedBehaviour BEHAVIOUR
          DEFINED AS
          !The specificAccessControlUsed attribute indicates
          whether access control specific to an SNMP agent
          is none, uses Internet defined mechanisms and MIB
          elements, or uses ISO/CCITT defined mechanisms and
          MIB elements.!;;
          MATCHES FOR    EQUALITY, ORDERING;
REGISTERED AS {cmipsnmpProxyPMIB 1 1 6};

supportedMIBs ATTRIBUTE
          WITH ATTRIBUTE SYNTAX
          CmipsnmpProxyASN1.SupportedMIBs;
               BEHAVIOUR
               supportedMIBsBehaviour BEHAVIOUR
               DEFINED AS
               !This attributes specifies the set of
               Internet OIDs of registered MIBs that the
               agent supports.!;;
          MATCHES FOR    EQUALITY, SET-COMPARISON,
                                   SET-INTERSECTION;
REGISTERED AS {cmipsnmpProxyPMIB 1 1 5};
```

4.2 Name Bindings

ISO/CCITT-Internet proxy namebindings are registered under the {cmipsnmpProxyPMIBNB} arc which is the {cmipsnmpProxyMISC 2} arc. The name bindings are registered by appending two sub-identifiers: the first sub-identifier is associated with the object class to which the name binding applies, the second sub-identifier identifies the instance of the name binding for that object class. The second sub-identfier must be greater than or equal to 1.

```
cmipsnmpProxyAgent-NB  NAME BINDING
              SUBORDINATE OBJECT CLASS
               cmipsnmpProxyAgent AND SUBCLASSES;
              NAMED BY SUPERIOR OBJECT CLASS
               cmipsnmpProxyTable AND SUBCLASSES;
              WITH ATTRIBUTE cmipsnmpProxyAgentId;
              CREATE;
              DELETE DELETES-CONTAINED-OBJECTS;
              REGISTERED AS {cmipsnmpProxyPMIBNB 2 1};

cmipsnmpProxyTable-NB  NAME BINDING
              SUBORDINATE OBJECT CLASS
               cmipsnmpProxyTable AND SUBCLASSES;
              NAMED BY SUPERIOR OBJECT CLASS
               "Rec. X.721 | ISO/IEC 10165-2 :1992": system
                                    AND SUBCLASSES;
              WITH ATTRIBUTE internetClassId;
              CREATE;
              DELETE     ONLY-IF-NO-CONTAINED-OBJECTS;
              REGISTERED AS {cmipsnmpProxyPMIBNB 1 1};
```

4.3 Common SNMP Derived Attribute Types

Internet Attribute types defined by the SNMP-2 TEXTUAL-CONVENTION macro are translated into ISO/CCITT attribute types. Attributes of the defined SNMP/SNMP-2 types (Counter, IpAddress, Gauge, TimeTicks, Opaque, Counter32, Gauge32, Counter64, NsapAddress), which could also have been defined in a TEXTUAL-CONVENTION macro, are also considered to be Internet attribute types.

ISO/CCITT Attribute templates derived from these types should contain the DERIVED FROM clause.

The following ISO/CCITT attribute types, listed in alphabetical order, are derived from Internet attribute types to facilitate Internet MIB translation. Other attributes may be derived from these types.

```
autonomousType ATTRIBUTE
     WITH ATTRIBUTE SYNTAX CmipsnmpCommonDef.OctetString;
          BEHAVIOUR
          autonomousTypeBehaviour BEHAVIOUR
          DEFINED AS
          !Represents an independently extensible type
          identification value.  It may, for example,
          indicate a particular sub-tree with further
          MIB definitions, or define a particular type of
          protocol or hardware.  This corresponds to the
          type defined in [SMPTC] by the same name.!;;
     MATCHES FOR    EQUALITY, ORDERING, SUBSTRINGS;;

counter32 ATTRIBUTE
```

```
     WITH ATTRIBUTE SYNTAX CmipsnmpProxyASN1.Counter32;
          BEHAVIOUR
          counter32Behaviour BEHAVIOUR
          DEFINED AS
          !As defined for counter type in ISO/IEC 10165-2.
          Only the value range constraint (0..4294967295)
          is added.  This corresponds to the type defined in
          [SMPSMI] by the same name!;;
          MATCHES FOR    EQUALITY, ORDERING;;

counter64 ATTRIBUTE
     WITH ATTRIBUTE SYNTAX CmipsnmpProxyASN1.Counter64;
          BEHAVIOUR
          counter64Behaviour BEHAVIOUR
          DEFINED AS
          !As defined for counter type in ISO/IEC 10165-2.
          Only the value range constraint
          (0..18446744073709551615) is added.
          This corresponds to the type defined in [SMPSMI]
          by the same name!;;
          MATCHES FOR    EQUALITY, ORDERING;;

dateAndTime ATTRIBUTE
     WITH ATTRIBUTE SYNTAX CmipsnmpProxyASN1.DateAndTime;
          BEHAVIOUR
          dateAndTimeBehaviour BEHAVIOUR
          DEFINED AS
          !DISPLAY-HINT "2d-1d-1d,1d:1d:1d.1d,1a1d:1d"
          A date-time specification.
          field  octets   contents                range
          -----   ------   --------                -----
          1      1-2     year                    0..65536
          2       3      month                   1..12
          3       4      day                     1..31
          4       5      hour                    0..23
          5       6      minutes                 0..59
          6       7      seconds                 0..60
          (use 60 for leap-second)
          7       8      deci-seconds            0..9
          8       9      direction from UT       "+" / "-"
          9      10      hours from UT           0..11
          10     11      minutes from UT         0..59

          For example, Tuesday May 26, 1992 at 1:30:15 PM
          EDT would be displayed as:1992-5-26,13:30:15.0,-
          4:0

          Note that if only local time is known, then
          timezone information (fields 8-10) is not present.

          This corresponds to the type defined in [SMPSMI]
          by the same name!;;
          MATCHES FOR    EQUALITY, ORDERING;;
```

```
displayString ATTRIBUTE
     WITH ATTRIBUTE SYNTAX
          CmipsnmpCommonDef.OctetString;
          BEHAVIOUR
          displayStringBehaviour BEHAVIOUR
          DEFINED AS
          !DISPLAY-HINT "255a"
          Represents textual information taken from the NVT
          ASCII character set, as defined in pages 4, 10-11
          of RFC 854.  Any object defined using this syntax
          may not exceed 255 characters in length.  This
          corresponds to the type defined in [SMPTC] by
          the same name.!;;
     MATCHES FOR    EQUALITY, ORDERING, SUBSTRINGS;;

gauge32 ATTRIBUTE
     WITH ATTRIBUTE SYNTAX CmipsnmpProxyASN1.Gauge32;
          BEHAVIOUR
          gauge32Behaviour BEHAVIOUR
          DEFINED AS
          !As defined for the integer gauge type in ISO/IEC
          10165-2.  Only the value range constraint
           (0..4294967295) is added.
          This corresponds to the type defined in [SMPSMI]
          by the same name!;;
          MATCHES FOR    EQUALITY, ORDERING;;

instancePointer ATTRIBUTE
     WITH ATTRIBUTE SYNTAX
          CmipsnmpCommonDef.ObjectIdentifier;
          BEHAVIOUR
          instancePointerBehaviour BEHAVIOUR
          DEFINED AS
          !A pointer to a specific instance of a conceptual
          row of a MIB table in the managed device.  By
          convention, it is the name of the particular
          instance of the first columnar object in the
          conceptual row. This corresponds to the type
          defined in [SMPTC] by the same name.!;;
     MATCHES FOR    EQUALITY, ORDERING;;

ipAddress ATTRIBUTE
     WITH ATTRIBUTE SYNTAX CmipsnmpCommonDef.OctetString;
          BEHAVIOUR
          ipAddressBehaviour BEHAVIOUR
          DEFINED AS
          !This attribute represents a 32-bit internet
          address.  It is represented as an octet string
          of length 4, in network Byte-order.
          This corresponds to the type defined in [SMPSMI]
          by the same name!;;
          MATCHES FOR    EQUALITY, ORDERING, SUBSTRINGS;;

macAddress ATTRIBUTE
```

```
     WITH ATTRIBUTE SYNTAX CmipsnmpProxyASN1.MacAddress;
          BEHAVIOUR
          macAddressBehaviour BEHAVIOUR
          DEFINED AS
          !DISPLAY-HINT "1x:"
          Represents an 802 MAC address represented in the
          `canonical' order defined by IEEE 802.1a, i.e.,
          as if it were transmitted least significant bit
          first, even though 802.5 (in contrast to other
          802.x protocols) requires MAC addresses to be
          transmitted most significant bit first.
          This corresponds to the type defined in [SMPTC]
          by the same name.!;;
     MATCHES FOR    EQUALITY, ORDERING, SUBSTRINGS;;

nsapAddress ATTRIBUTE
     WITH ATTRIBUTE SYNTAX CmipsnmpCommonDef.OctetString;
     BEHAVIOUR
          nsapAddressBehaviour BEHAVIOUR
          DEFINED AS
          !This attribute represents an ISO/CCITT network
          address.  It is represented as a variable
          length octet string.  The first octet of the
          string contains a binary value, in the range of
          0..20, and indicates the length in octets of
          the NSAP.  Following the first octet, is the
          NSAP expressed in concrete binary notation,
          starting with the most significant octet.  A
          zero-length NSAP is used as a "special"
          address, meaning "the default NSAP" (analogous
          to the IP address of 0.0.0.0).  Such an NSAP
          is encoded as a single octet containing the
          value 0.  All other NSAPS are encoded in at
          least 4 octets.  This corresponds to the type
          defined in [SMPSMI] by the same name!;;
     MATCHES FOR    EQUALITY, ORDERING, SUBSTRINGS;;

opaque ATTRIBUTE
     WITH ATTRIBUTE SYNTAX CmipsnmpCommonDef.OctetString;
          BEHAVIOUR
          opaqueBehaviour BEHAVIOUR
          DEFINED AS
          !This attribute represents arbitrary ASN.1
          syntax.  A value is encoded using the Basic
          Encoding Rules [ISO8825] into a string of
          octets.  This, in turn, is encoded as an OCTET
          STRING, in effect "double-wrapping" the
          original ASN.1 value.  This corresponds to the
          type defined in [SMPSMI] by the same name.!;;
     MATCHES FOR    EQUALITY, ORDERING;;

physAddress ATTRIBUTE
     WITH ATTRIBUTE SYNTAX CmipsnmpCommonDef.OctetString;
          BEHAVIOUR
```

```
          physAddressBehaviour BEHAVIOUR
          DEFINED AS
          !DISPLAY-HINT "1x:"
              Represents media- or physical-level addresses.
          This corresponds to the type defined in [SMPTC]
          by the same name.!;;
     MATCHES FOR    EQUALITY, ORDERING, SUBSTRINGS;;

rowStatus ATTRIBUTE
     WITH ATTRIBUTE SYNTAX
               CmipsnmpProxyASN1.RowStatus;
          BEHAVIOUR
          rowStatusBehaviour BEHAVIOUR
          DEFINED AS
          !The syntax used for the status column for a
           conceptual row.  If present, the value of the
           DEFVAL clause for an object having this syntax is
           either `underModification(3)' or `active(4)'.

           To create new object instances for a conceptual
           row, a management protocol set operation is
          issued which sets the new instance of the status
          column to `underCreation(1)'.  If the instance
          already exists, then the management protocol set
          operation fails with an error of
          `inconsistentValue'.
           Otherwise, the instance is created.  If the
           management protocol set operation created
           sufficient instances so that this conceptual row
           may be used by the correspondent SMP entity, and
           the value of the DEFVAL clause for the status
           column is `active(4)', then the SMP entity acting
           in an agent role immediately sets the value of
           this instance to `active(4)'.  Otherwise, the SMP
           entity acting in an agent role immediately sets
           the value of this instance to
           `underModification(3)'.

          See [SMPTC] for a description of the algorithm to
          create a new conceptual row.

          This corresponds to the type defined in [SMPTC]
          by the same name.!;;
     MATCHES FOR    EQUALITY, ORDERING;;

testAndIncrement ATTRIBUTE
     WITH ATTRIBUTE SYNTAX
               CmipsnmpProxyASN1.TestAndIncrement;
          BEHAVIOUR
          testAndIncrementBehaviour BEHAVIOUR
          DEFINED AS
          !Represents integer-valued information used for
           atomic operations.  When the management protocol
           is used to specify that an object instance having
```

```
          this syntax is to be modified, the new value
          supplied via the management protocol must
          precisely match the value presently held by the
          instance.  If not, the management protocol set
          operation fails with an error of
          `inconsistentValue'.  Otherwise, if the current
          value is the maximum value of 2^31-1 (2147483647
          decimal), then the value held by the instance is
          wrapped to zero; otherwise, the value held by the
          instance is incremented by one.  (Note that
          regardless of whether the management protocol set
          operation succeeds, the previous value held by
         the instance is returned.)

          The value of the ACCESS clause for objects having
          this syntax is either `read-write' or `read-
          create'.  When an instance of a columnar object
          having this syntax is created, any value may be
          supplied via the management protocol.
         This corresponds to the type defined in [SMPTC]
         by the same name.!;;
    MATCHES FOR    EQUALITY, ORDERING;;

timeInterval ATTRIBUTE
    WITH ATTRIBUTE SYNTAX
              CmipsnmpProxyASN1.Integer32;
              BEHAVIOUR
         timeIntervalBehaviour BEHAVIOUR
         DEFINED AS
         !A period of time, measured in units of 0.01
         seconds.
         This corresponds to the type defined in [SMPTC]
         by the same name.!;;
    MATCHES FOR    EQUALITY, ORDERING;;

timeStamp ATTRIBUTE
    WITH ATTRIBUTE SYNTAX
              CmipsnmpProxyASN1.TimeTicks;
              BEHAVIOUR
         timeStampBehaviour BEHAVIOUR
         DEFINED AS
         !The value of MIB-II's sysUpTime object at which a
         specific occurrence happened.  The specific
         occurrence must be defined in the description of
         any object defined using this type.
         This corresponds to the type defined in [SMPTC]
         by the same name.!;;
    MATCHES FOR    EQUALITY, ORDERING;;

timeTicks ATTRIBUTE
    WITH ATTRIBUTE SYNTAX
         CmipsnmpProxyASN1.TimeTicks;
         BEHAVIOUR
         timeTicksBehaviour BEHAVIOUR
```

```
          DEFINED AS
          !This attribute represents a non-negative
          integer which represents the time, modulo 2->32
          (4294967296 decimal), in hundredths of a second
          between two epochs.  When attributes are
          defined which use this attribute type, the
          description of the object identifies both of
          the reference epochs.  This corresponds to the
          type defined in [SMPSMI] by the same name!;;
     MATCHES FOR    EQUALITY, ORDERING;;

truthValue ATTRIBUTE
     WITH ATTRIBUTE SYNTAX CmipsnmpProxyASN1.TruthValue;
          BEHAVIOUR
          truthValueBehaviour BEHAVIOUR
          DEFINED AS
          !Represents a boolean value.
          This corresponds to the type defined in [SMPTC]
          by the same name.!;;
     MATCHES FOR    EQUALITY;;
```

4.4 Notifications for SNMP/SNMP-2 Traps

Notification templates for the SNMP generic traps are listed here in alphabetical order. They are registered under the {cmipsnmpProxyNOT} arc with sub-identifiers allocated according to the SNMP-2 assignment for generic traps.

Pending progression of SNMP-2 to an Internet standard, and the subsequent assignment of OIDs under the "internet" arc, this memo has not assigned OIDs for the standard notifications defined in [RFC1157] and [SMPMIB].

```
authenticationFailure NOTIFICATION
     BEHAVIOUR
          authenticationFailureBehaviour BEHAVIOUR
          DEFINED AS
          !This notification maps to the
          authenticationFailure generic trap defined in RFC
          1157, i.e., generic-trap=4.

          An authenticationFailure notification signifies
          that the SNMP/SNMP-2 entity, acting in an agent
          role, has received a protocol message that is not
          properly authenticated. The snmpEnableAuthentraps
          attribute of the snmp object class indicates
          whether this notification will be generated by an
          SNMP agent.!;;
        WITH INFORMATION SYNTAX
             CmipsnmpProxyASN1.MonitoredAttributes
          AND ATTRIBUTE IDS
             monitoredAttributes
          "Rec. X.721 | ISO/IEC 10165-2 : 1992"
                                  :monitoredAttributes;
```

```
REGISTERED AS {cmipsnmpProxyNOT ?};

coldStart NOTIFICATION
     BEHAVIOUR
          coldStartBehaviour BEHAVIOUR
          DEFINED AS
          !This notification maps to the coldStart generic
          trap defined in RFC 1157, i.e., generic-trap=0.

          A coldStart notification signifies that an
          SNMP/SNMP-2 entity acting in an agent role, is
          reinitializing itself such that its configuration
          may be altered.!;;
         WITH INFORMATION SYNTAX
               CmipsnmpProxyASN1.MonitoredAttributes
          AND ATTRIBUTE IDS
               monitoredAttributes
          "Rec. X.721 | ISO/IEC 10165-2 : 1992"
                                    :monitoredAttributes;
REGISTERED AS {cmipsnmpProxyNOT ?};

egpNeighborLoss NOTIFICATION
     BEHAVIOUR
          egpNeighborLossBehaviour BEHAVIOUR
          DEFINED AS
          !This notification maps to the egpNeighborLoss
          generic trap defined in RFC 1157, i.e., generic-
          trap=5.

          An egpNeighborLoss notification signifies
          that an EGP neighbor has been marked down
          and the RGP peer relationship no longer
obtains.!;;
         WITH INFORMATION SYNTAX
               CmipsnmpProxyASN1.MonitoredAttributes
          AND ATTRIBUTE IDS
               monitoredAttributes
          "Rec. X.721 | ISO/IEC 10165-2 : 1992"
                                    :monitoredAttributes;
REGISTERED AS {cmipsnmpProxyNOT ?};

linkDown NOTIFICATION
     BEHAVIOUR
          linkDownBehaviour BEHAVIOUR
          DEFINED AS
          !This notification maps to the linkDown generic
          trap defined in RFC 1157, i.e., generic-trap=2.
          The ifIndex of the interface causing the event
          shall appear in the monitored attributes.

          A linkdown notification signifies that the
     SNMP/SNMP-2, entity acting in an agent role,
          recognizes a failure in one of the communication
          links represented in its configuration.!;;
```

```
          WITH INFORMATION SYNTAX
               Attribute-ASN1Module.MonitoredAttributes
           AND ATTRIBUTE IDS
               monitoredAttributes
           "Rec. X.721 | ISO/IEC 10165-2 : 1992"
                                    :monitoredAttributes;
REGISTERED AS {cmipsnmpProxyNOT ?};

linkUp NOTIFICATION
     BEHAVIOUR
          linkUpBehaviour BEHAVIOUR
          DEFINED AS
          !This notification maps to the linkUp generic trap
          defined in RFC 1157, i.e., generic-trap=3.  The
          ifIndex of the interface causing the event shall
          appear in the monitoredAttributes.

          A linkup notification signifies that the
          SNMP/SNMP-2, entity acting in an agent role,
          recognizes that one of the communication links
          represented in its configuration has come up.!;;
         WITH INFORMATION SYNTAX
               CmipsnmpProxyASN1.MonitoredAttributes
          AND ATTRIBUTE IDS
               monitoredAttributes
          "Rec. X.721 | ISO/IEC 10165-2 : 1992"
                                    :monitoredAttributes;
REGISTERED AS {cmipsnmpProxyNOT ?};

warmStart NOTIFICATION
     BEHAVIOUR
          warmStartBehaviour BEHAVIOUR
          DEFINED AS
          !This notification maps to the warmStart generic
          trap defined in RFC 1157, i.e., generic-trap=1.

          A warmstart notification signifies that an
          SNMP/SNMP-2 entity acting in an agent role, is
          reinitializing itself such that its configuration
          is unaltered.!;;
         WITH INFORMATION SYNTAX
               CmipsnmpProxyASN1.MonitoredAttributes
          AND ATTRIBUTE IDS
               monitoredAttributes
          "Rec. X.721 | ISO/IEC 10165-2 : 1992"
                                    :monitoredAttributes;
REGISTERED AS {cmipsnmpProxyNOT ?};
```

4.5 ASN.1 Definitions

```
CmipsnmpProxyAssignedOIDs  {cmipsnmpProxyPMIBMOD 1}
DEFINITIONS ::= BEGIN
EXPORTS ;-- EVERYTHING
```

```
cmipsnmpProxy          OBJECT IDENTIFIER ::= {...}
cmipsnmpProxyIMIB   OBJECT IDENTIFIER ::= {cmipsnmpProxy 1}
     -- for ISO/CCITT derived object classes and attributes
cmipsnmpProxyNB      OBJECT IDENTIFIER ::= {cmipsnmpProxy 2}
     -- for ISO/CCITT derived NAME BINDINGS
cmipsnmpProxyNOT    OBJECT IDENTIFIER ::= {cmipsnmpProxy 3}
     -- for ISO/CCITT Notifications
cmipsnmpProxyPMIB  OBJECT IDENTIFIER ::= {cmipsnmpProxy 4}
     -- for ISO/CCITT Proxy MIB defined in this memo
cmipsnmpProxyPMIBNOT OBJECT IDENTIFIER ::=
{cmipsnmpProxy 5}
     -- for standard SNMP notifications
cmipsnmpProxyPMIBNB OBJECT IDENTIFIER ::= {cmipsnmpProxy 6}
     -- for ISO/CCITT Proxy MIB NAME BINDINGS
cmipsnmpProxyPMIBMOD OBJECT IDENTIFIER ::= {cmipsnmpProxy 7}
     -- for ISO/CCITT Proxy MIB ASN.1 Modules
cmipsnmpProxyMISC  OBJECT IDENTIFIER ::= {cmipsnmpProxy 8}
     -- for micellaneous

snmpOID   OBJECT IDENTIFIER ::= {cmipsnmpProxyMISC 1}

snmp2OID OBJECT IDENTIFIER ::= {cmipsnmpProxyMISC 1}
     -- required for management protocol identification
END

{Editor's Note:  Should OIDs be assigned for SNMP and SNMP-2
in this document?}

CmipsnmpCommonDef {cmipsnmpProxyPMIBMOD 2}
          DEFINITIONS ::= BEGIN
          EXPORTS -- EVERYTHING
          Integer, OctetString, ObjectIdentifier,
          Null, BitString;
          --
          Integer ::= INTEGER
          OctetString ::= OCTET STRING
          ObjectIdentifier ::= OBJECT IDENTIFIER
          Null ::= NULL
          BitString ::= BIT STRING
          END

CmipsnmpProxyASN1  {cmipsnmpProxyPMIBMOD 3}
          DEFINITIONS ::= BEGIN
          IMPORTS   MonitoredAttributes
          FROM Attribute-ASN1Module;
               -- defined in
               -- Rec. X.721 | ISO/IEC 10165-2 : 1992,
          Counter32 ::= INTEGER (0..4294967295)
          Counter64 ::= INTEGER (0..18446744073709551615)
          DateAndTime ::= ::= OCTET STRING (SIZE (8 | 11))
          Gauge32    ::= INTEGER (0..4294967295)
          TimeTicks ::=  INTEGER (0..4294967295)
          MacAddress ::= OCTET STRING (SIZE (6))
          TruthValue ::= INTEGER { true(1), false(2) }
```

```
TestAndIncrement ::= INTEGER (0..2147483647)
RowStatus ::=  INTEGER {
                    active(1)
                    underConstruction(2),
                    underDestruction(3),
                    underModification(4),
                         }

CmipsnmpProxyAgentId ::= GraphicString
AccessControlUsed   ::= INTEGER    {
                                   none      (1),
                                   internet  (2),
                                   osi       (3)}
AccessControlEnforcement ::= INTEGER    {
                                   agent     (1),
                                   proxy     (2),
                                   both      (3)}
SupportedMIBs ::= SET OF OBJECT IDENTIFIER
Zero ::= INTEGER 0
END
```

4.6 ISO/CCITT-Internet Proxy Communications

An ISO/CCITT-Internet proxy requires knowledge of the SNMP agent's "community string" or SNMP "party" values to communicate with the agent. This information may be placed in MIB elements defined in the SNMP Party MIB [IIMCPARTY] which is derived from the Internet MIB defined in [RFC1353]. Therefore the proxy must also include, at a minimum, the "partyTable" and "partyEntry" object classes defined in [IIMCPARTY]. These object classes contain the information required for party-based security (including authentication, integrity, and confidentiality services), and are adapted for use with community-based security. The MIB defined in [RFC1353] is also included in the SNMP-2 MIB [SMPMIM].

The Party MIB may also contain information required for controlling access to operations on management information based on an access control list mechanism. Access control list information is contained the "aclTable" entries. Constraints on what information may be accessed may be placed in the "viewTable" entries.

5. Acknowledgements

The author thanks the following individuals for their insightful comments and contributions:

Jon Biggar - NETLABS
April Chang - NETLABS
Dean Voiss - NETLABS
Jock Embry - Opening Technologies
Steve Ng - MPR Teltech

Lisa Phifer - Bellcore
Owen Newnan - US West Advanced Technologies

References

[ISO8824] ISO/IEC IS 8824: Information Technology - Open System Interconnection - Specification of Abstract Syntax Notation One (ASN.1),1990.

[ISO8825] ISO/IEC IS 8825: Information Technology - Open System Interconnection - Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1),1990.

[ISO7498-4] ISO/IEC IS 7498-4, Information Processing Systems - Open Systems Interconnection - Basic Reference Model Part 4 - Management Framework, 1989.

[ISO9595] ISO/IEC IS 9595, Information Technology - Open SystemInterconnection - Common Management Information Service Definition, 1991.

[ISO9596-1] ISO/IEC IS 9596-1, Information Technology - Open Systems Interconnection - Common Management Information Protocol - Part 1: Specification, 1991.

ISO10165-1] ISO/IEC IS 10165-1: Information Technology - Open Systems Interconnection - Structure of Management Information - Part 1: Management Information Model, 1991.

[ISO10165-2] ISO/IEC IS 10165-2: Information Technology - Open Systems Interconnection - Structure of Management Information - Part 2:Definition of Management Information, 1992.

[ISO10165-4] ISO/IEC IS 10165-4: Information Technology - Open Systems Interconnection - Structure of Management Information - Part 4: Guidelines for the Definition of Managed Objects, 1991.

[RFC1155] RFC1155, M. Rose and K. McCloghrie, Structure and Identification of Management Information for TCP/IP based internets, May 1990.

[RFC1157] RFC 1157, J.D. Case, M.S. Fedor, M.L. Schoffstall, C. Davin, Simple Network Management Protocol (SNMP), May 1990.

[RFC1212] RFC1212, M. Rose, K. McCloghrie - Editors, Concise MIB Definitions, March 1991.

[RFC1213] RFC1213, K. McCloghrie and M. Rose - Editors, Management Information Base for Network Management of TCP/IP-based internets: MIB-II, March 1991.

[RFC1214] RFC1214, L. LaBarre - editor, OSI Internet Management:Management Information Base, April 1991.

[RFC1215] RFC1215, M. Rose - Editor, Management A convention for Defining Traps for use with the SNMP, March 1991.

[RFC1353] RFC1353, K. McCloghrie, J.R. Davin, J.M. Galvin, Definitions of Managed Objects for SNMP Parties, July 1992.

[SMPCOEX] J.D. Case, K. McCloghrie, M.T. Rose, S.L. Waldbusser, Coexistance between the Internet-standard Network Management Framework and the Simple Management Protocol (SMP) Framework, Internet-draft, October 1992.

[SMPPROT] J.D. Case, K. McCloghrie, M.T. Rose, S.L. Waldbusser, Protocol Operations for the Simple Management Protocol (SMP) Framework, Internet-draft, October 1992.

[SMPSMI] J.D. Case, K. McCloghrie, M.T. Rose, S.L. Waldbusser, Protocol Structure of Management Information for the Simple Management Protocol (SMP) Framework, Internet-draft, October 1992.

[SMPMIB] J.D. Case, K. McCloghrie, M.T. Rose, S.L. Waldbusser, Management Information Base for the Simple Management Protocol (SMP) Framework, Internet-draft, October 1992.

[SMPTC] J.D. Case, K. McCloghrie, M.T. Rose, S.L. Waldbusser, Textual Conventions for the Simple Management Protocol (SMP) Framework, Internet-draft, October 1992.

[IIMCPARTY] L. LaBarre, ISO/CCITT and Internet Management Coexistence: Translation of Internet Party MIB (RFC1353) to ISO/CCITT GDMO MIB, October 1992.

[IIMCMIB-II] L. LaBarre, ISO/CCITT and Internet Management Coexistence: Translation of Internet MIB-II (RFC1213) to ISO/CCITT GDMO MIB, October 1992.

[IIMCPROXY] A. Chang, ISO/CCITT and Internet Management Coexistence: ISO/CCITT to Internet Management Proxy, October 1992.

[IIMCOMIBTRANS] O. Newnan, ISO/CCITT and Internet Management Coexistence: Translation of ISO/CCITT GDMO MIBs to Internet MIBs, October 1992.

[NMFMC92] NM Forum and X/Open, ISO/CCITT/CCITT and Internet Management: Coexistence and Interworking Strategy, October, 1992.