ATN Panel

Systems Management

Proposal 2: MO Template for ICAO SARPs
12 February 1998

Prepared By:
Pam Tupitza
Open Network Solutions

Presented By:
James Moulton
Open Network Solutions

Table of Contents

1. Scope 3
1.1 References 3
2.0 Overview 3
3.0 Analysis of Proposed Templates for ICAO SARPs 5
3.1 Packages 5
3.2 Attributes 5
3.3 Actions 5
3.4 Notifications 6
3.5 Containment and Naming 6
3.6 Allomorphism 6
3.7 Relationships 7
3.8 MIB/Relationships 7
4.0 Managed Object Profile 7
4.1 Column Descriptions 8
5.0 Managed Object Definition 9
6.0 Annex A - Open Issues 10
7.0 Annex B - Example MO Definition 11
7.1 RDE Pair Managed Object 11
7.2 ASN.1 Definitions 15

1.0 Scope

This paper reviews the proposed MO Template for ICAO SARPs. It offers an alternative Information Model for managed objects and associated information that corresponds to the systems management model introduced in the Systems Management Overview CCITT Rec. X.701 | ISO/IEC 10040. It suggests the use of those modeling concepts for the ICAO ATN systems management specifications.

1.1 References

ISO/IEC 8802-2:1994 Amendment 6: *Managed objects definition for Logical Link Control (LLC)*
ISO/IEC 10165-1:1993 *Information technology - Open Systems Interconnection - Structure of Management Information: Management Information* Model
ISO/IEC 10165-4:1992 *Information technology - Open Systems Interconnection - Structure of management information - Part 4:Guidelines for the definition of managed objects.*
ISO/IEC 10165-7:
ATNP/WG1/SG3:WP3-7 *Proposed MO Template for ICAO SARPs*

2.0 Overview

ATN SARPs will reuse previously defined ISO Managed Objects as well as define new Managed Objects or new subclasses of existing Managed Objects. The Information Model used should be consistent with ISO standards to maximize reuse of MO's from other standards and reduce errors when subclassing. The interoperability of the information model is as crucial as the interoperability of the system.

Agreeably the template should be as succinct and clear as possible, without adding unnecessary complexity. However the models used to describe the SARPS need to be complete enough to allow the specification and evolution of a complex network system. The use of the OSI GDMO MO templates are suggested to take advantage of the rigor of a fully developed methodology.

However..to address the concerns raised about unnecessary complexity the following 'amendments' are proposed to simplify the OSI methodology:

- Do not allow the use of multiple inheritance. Multiple inheritance can be normalized away with the proper design of the object class hierarchy combined with the use of aggregations. So, each subclass need derive from only one superclass.

- Model templates as part of both inheritance and aggregation hierarchies instead of containment hierarchies. Typically OSI uses a containment hierarchy that provides a combination of part/whole description and naming scheme. The containment hierarchy contains no multiplicity specification (how many components are required to constitute the aggregate object). OSI GRM suggests a complicated workaround for the cardinality limitation. The SARPS can simplify by using a true aggregation hierarchy (which includes cardinality) to specify the part/whole relationships. The naming scheme can then be created independently of the aggregation semantics.

- Enforce monotonic (strict) inheritance for the inheritance hierarchies[ISO/IEC 10165-1]. This means every subclass must inherit each and every attribute and function specified for its ancestral classes and may not cancel any of them. This will prevent complications created by allowing the classes idiosyncratic behavior.

- Disallow n-ary relationships. OSI relationship classes permit more than two objects to participate in a relationship. In reality n-ary relationship (with n>2) are rare and most can be decomposed into multiple binary relationships.

- Model a separate relationship hierarchy. OSI allows a intertwining of relationship classes with MO classes, this is confusing. If relationships need to be maintained in a hierarchy, separate the trees: one for MO classes and one for relationship classes.

This paper adds a proposal to modify the MO template to a MO profile template. This new template would provide organization and references to established ISO MO's as well as 'new' ATN MO's used for the ATN project. The actual definition of new ATN MO's and new subclasses would be formatted into ISO templates.

3.0 Analysis of Proposed MO Templates for ICAO SARPS

There are a number of OSI 'specifications' that are not specifically alluded to in the template, but are included in the example. Propose the following to clarify the exact 'fields' in each proposed template and what their purpose is.

3.1 Packages

Problem: There is no Package Template proposed. Was the package attribute in the CO TPM MO example supposed to cover this? Typically the Packages attribute is used to identify the packages that have been instantiated. However, the proposed template offers no provision for a Package template to define the packages.

Suggestion: The ISO Package Template represents collections of properties such as attributes and notifications. Add back the package template to allow the grouping of commonly used attributes. We definitely want to allow the specification re-use of attribute groups. We don't want to get into the respecification of the same attribute group in different MO's. If we do not use packages, current predefined MO's may have to be 'normalized' into separate object classes to allow reuse of attributes and notifications. Obviously if a package occurs only in one managed object classes, there is no issue.

3.2 Attributes

Problem: It is not specifically stated in the proposed template if the attribute specification is using the ASN.1 notation. If so, the respecification of the attribute in every managed object class would be required. It is also not stated if attribute groups are allowed. (Attribute groups can allow the managed object class which contains it to extend it with object specific state attributes.)

Suggestion: Use ASN.1 notation to specify attributes. Use the Object Identifier label to identify each attribute construct and list it in the template table. Specify the attribute structures in a following section of the SARPS. This would be consistent with the format of other SARPs.

3.3 Actions

Problem: No explicit description of information syntax or reply syntax.

Suggestion: Use the ACTION template and information/reply syntax to specify the ASN.1 data type of the action parameter that is carried in the management protocol.

Question: Only actions allowed are those using CMIS M_action service?

3.4 Notifications

Problem: The OSI Notification Template specifies which notifications the managed object can issue to the management system. Each notification template should specify the syntax of the information associated with the notification, the syntax of the reply associated with and indicate whether or not it is confirmed. These are not explicitly defined as identifiers in the proposal for the template, although similar information is contained in the example.

Suggestion: Include explicit OSI identifiers required to populate template.

3.5 Containment and Naming

Containment specifies which managed object classes may be contained in other managed object classes and is defined using a construct known as a name binding. Do we intend to use the OSI containment hierarchy to specify the aggregation hierarchy (part/whole relationships)? In OSI containment is also used as a naming mechanism, the addressing scheme for MO's arises out of their logical containment in other objects. Each MO can be addressed using its distinguished name. At each level of containment the attribute used to name the MO could be different; each name binding therefore specifies the attribute used by the superior managed object to name the subordinate managed object. This helps ensure uniqueness among multiple subordinate managed objects contained in the same superior managed object.

Problem: Name binding does not indicate how many instances of the subordinate component may be contained in an instance of the superior. OSI GRM suggest a RELATIONSHIP CLASS defined to work around the cardinality limitation. This is overkill. Suggest an aggregation diagram with cardinality.

3.6 Allomorphism

Typically a MO (OSI) can declare itself allomorphic using the allomorphs attribute present in every MO. Allomorphism is the ability of a MO to present itself as if it were an object of a class other than its own. For example, a management system may not know the model of the new version of some MO. In this case, the new MO can declare itself to be allomorphic to some old class already known to the management system and thereby permit the management system to manage it in some limited fashion.

Problem: It is not specifically stated in the template if this attribute is used although it appears in the example. Does ICAO want to use it? Could this be useful for interoperability as states, CAAs, etc continue to evolve their own MO's? Or is this adding unnecessary complexity?

Suggestion: ISO/IEC 10165-1 [Sec. 5.2.3] Methods for providing interoperability offers specific information on how to use allomorphism for interoperability.

3.7 Relationships

Problem: There is no proposal for the modeling of relationships. There are two relationship models: the old OSI standard [ISO/IEC 10164-3] defines only five specific relationships between MO's: service, fallback, backup, group and peer, or the General Relationship Model [ISO/IEC 10165-7]. GRM provides more general and flexible relationship modeling mechanisms. In the GRM the roles of a relationship can be dynamic or static. Will we allow both static and dynamic relationships? (If a role is static all MO's playing that role must be bound in a relationship at the same time that the relationship is established and can only be unbound when the relationship is destroyed.)

Suggestion: Use the OSI GRM Relationship template with the limitation that n-ary relationships are prohibited. Require n-ary relationships to be divided into multiple binary relationships. Suggest using two entirely separate trees in the model one for actual participant MO classes and another for relationship classes. (Typically OSI GRM intertwines relationship classes with MO classes.)

3.7.1 MIB/Relationships

Should contain information about components, protocol stacks, interfaces and applications which each managed object has_as_a_part, and the relationship instances which that managed obect has with other objects. (e.g. termination relationship instances with link objects) .

Question: Will interoperability be defined with respect to a particular layer of a particular reference model. (managed object A interoperates with managed object B at the transport layer or at the application layer.

4.0 Managed Object Profile

Following is a modified MO template. This template attempts to address the original need for a tabular description for MO conformance requirements. This template will meet the need for description of managed objects, but the definition (if needed) will be accomplished in another section of the SARPs (See Annex A for an example). The template will allow the specifier to list existing MO's, from other standards or systems, for reuse as well as 'new' MO's or subclassed MO's specifically for the ATN. It does not attempt to provide an inhertitance hierarchy. It only seeks to provide a description for all MO's used for the specification of ATN System Management SARP's.

Index	**Managed Object Name <Description>**	**Manager/ Agent**	**Protocol Layer**	**ISO Status**	**ATN Status**
mo1	comodeTPM Connection_Oriented Transport Protocol Machine MO	Manager	Transport Layer	O	M
mo2	RDEPair Contains local and remote SAP addresses and describes LLC services.	Agent	Data Link Layer	M	M
mo3	moexampleATN	Manager/ Agent	Transport Layer	N/A	MA

4.1 Column Descriptions

Index: Provides same function as previous table.

Managed Object Name: MANAGED OBJECT CLASS <name>. (Do we need REGISTERED AS identifier?)

Manager/Agent: Used to identify the role of the managed object as manager, agent or both.

Protocol Layer: Identifies the layer(s) the in which the managed object is used.

ISO Status: Indicates the conformance requirement as specified in the ISO/IEC base standard. Values for ISO status are:

M - Mandatory to implement
O- Optional to implement
C - Dependent up on a condition explained in a footnote to the table

ATN Status: Indicates the conformance requirement as specified in the ATN SARPS. Possible values for ATN status are:

M - Mandatory to implement
R - Recommended to implement
O - Optional to implement
A - MO is ATN-specific

An “A” may be combined with other values.

Question: Do we need other columns for other groups defining MO’s? (ATNSI, AEEC?)

5.0 Managed Object Definition

This paper proposes templates defined in ISO/IEC 10165-4.

1. Managed Object Class
2. Package
3. Attribute
4. Attribute Group
5. Action
6. Notification
7. Name Binding

This paper proposes supporting definitions of the templates defined in ISO/IEC 10165-4.

1. Behavior. The BEHAVIOUR template is actually a supporting template. It is used to provide behavioral information to qualify the characteristics of MO classes, attributes, name bindings, actions and notifications from the management perspective.

This paper proposes templates defined in ISO/IEC 10165-7.

1. Relationship Class
2. Relationship Binding

This paper proposed supporting definitions of the templates defined in ISO/IEC 10165-7.

1. Role.

6.0 Annex A - Open Issues

Issue 1: Is there a metamodel showing the models used for the SARPS and their relationships to one another? MIB versus MO templates versus aggregation hierarchy etc..need to be defined in respect to each other and to the layers of the protocol stack. There needs to be a model so sub-groups can see clearly the portion of the model they are researching and pass issues for other models onto the appropriate sub-groups. This will avoid "But I thought they were working on it!" It would be helpful to model portions of work completed by other standards bodies that will be included in the SARPS (AEEC, RCTA, ANTSI etc.)

Issue 2: How should MO's 'borrowed' from other standards be referenced? How will updates to those MO's be handled? Will they be jointly maintained? A jointly maintained metamodel (between standards bodies) might help resolve some of these issues.

Issue 3: Notification template: It typically identifies the event reports which a managed object can emit using the CMIS M-Event-Report Service. Are the managed objects permitted to use other services?

Issue 4: Relationship template: Relationships can be established, queried, modified, and terminated during the normal course of operations of a network, or explicitly by a management system. When relationships are altered, it may issue a relationship change notification to the management system. Where do these fit?

Issue 5: Which group determines addressing scheme for registering MO's???
How will MO's from other standard's bodies be addressed?

Issue 6: Do we want to consider SARPS for network operations? This allows devices to be able to determine their interoperability on-line, through the run-time interrogation of the operations information base (MIB??) Connectivity and interconnectivity relationship between network devices may be inferred from the object model. Instances of these relationships between network devices can also be determined in the operations information base. Connectivity only indicate whether a sequence of wire-line or wireless links exists between some pair of network devices, they do not indicate whether they can interoperate.

7.0 Annex B - Example MO Definition

Managed Object Example from ISO/IEC 8802-2 Amendment 6: Managed object definition for Logical Link Control (LLC). This is intended solely as an example of the proposed template.

7.1 RDE Pair Managed Object

An instance of this managed object will exist for each RDE pair used by any type of LLC service provided. This managed object contains the local MAC address and local SAP and the remote MAC address and remote SAP, plus counters.

```
RDEPair MANAGED OBJECT CLASS
  DERIVED FROM ‘DMI: 1992”:top;
  CHARACTERIZED BY
   rDEPair-P;
    BEHAVIOUR
      rDEPair-B BEHAVIOUR
        DEFINED AS
            !An instance of this managed object will exist for each RDE pair used by any type
           of LLC service provided.  This managed object contains the local MAC address
           and local SAP and the remotes MAC address and remotes SAP, plus counters.!;;
REGISTERED AS {iso(1)member-body(2) us(840) ieee802-
2(10032)managedObjectCLass(3)rdepair(8)};
```

--Name Bindings—

```
rDEPair-NM NAME BINDING
SUBORDINATE OBJECT CLASS
 rDEPair AND SUBCLASSES;
NAMED BY SUPERIOR OBJECT CLASS
 rDESetup AND SUBLCASSES;
 WITH ATTRIBUTE rDEpair;
REGISTERED AS {iso(1)member-body(2)us(840)ieee802-
2(10032)namebinding(6)rdepairnb(8)};
```

--Packages—

```
rDEPair-P PACKAGE
  ATTRIBUTES
      rDEPairName                          GET-REPLACE,
      discardCounter                       GET-REPLACE,
      nSRPDICounter                        GET-REPLACE,
      nSRSelectedCounter          GET-REPLACE,
      rIF                                  GET-REPLACE,
      sRFPDUCounter                        GET-REPLACE,
      queryCounter                         GET-REPLACE,
ACTIONS
      resetRoute,
      readCounters;
REGISTERED AS {iso(1) member-body(2) us(840)ieee802-
2(10032)package(4)rdepairp(32)};
```

--Attributes—

```
rDEPairName ATTRIBUTE
  WITH ATTRIBUTE SYNTAX LLCDefinitions.RDEPairID
  MATCHES FOR EQUALITY
  BEHAVIOUR
   rDEPair-A-B BEHAVIOUR
    DEFINED AS ! The local LSAP and remote LSAPId and the local and remote
                  MACAdresses.!;;
REGISTERED AS {iso(1)member-body(2) us(840)ieee802-
2(10032)attribute(7)rdepaira(103)};

discardCounter ATTRIBUTE
  WITH ATTRIBUTE SYNTAX LLCDefinitions.Counter;
  MATCHES FOR EQUALITY
  BEHAVIOUR
   discardCounter-B BEHAVIOUR
      DEFINED AS ! A value which represents the number of times a new selection has
                   been made.!''
REGISTERED AS {iso(1)member-body(2) us(840)ieee802-
2(10032)attribute(7)discardcounter(104)};
```

```
nSRPDUCounter ATTRIBUTE
  WITH ATTRIBUTE SYNTAX LLCDefinitions.Counter2;
  MATCHES FOR EQUALITY
  BEHAVIOUR
   dnSRPDUCounter-B BEHAVIOUR
      DEFINED AS ! A value which represents the number of frames without a routing
                   field!;;
REGISTERED AS {iso(1)member-body(2) us(840)ieee802-
2(10032)attribute(7)nsrpducounter(105)};

nSRSelectedCounter ATTRIBUTE
  WITH ATTRIBUTE SYNTAX LLCDefinitions.Counter;
  MATCHES FOR EQUALITY
  BEHAVIOUR
   nSRSelectedCounter-B BEHAVIOUR
      DEFINED AS ! A value which represents the number of times that acceptable SRF
           route existed resulting in the selection of the NSR path.!;;
REGISTERED AS {iso(1)member-body(2) us(840)ieee802-
2(10032)attribute(7)nsrselectedcounter(106)};

rIF ATTRIBUTE
  WITH ATTRIBUTE SYNTAX LLCDefinitions.RIF;
  MATCHES FOR EQUALITY
  BEHAVIOUR
   rIF-B BEHAVIOUR
      DEFINED AS ! The routing information field varies in length up to 30 octets.  When
           source routing is not used the RIF is said to be NULL and contains no
           information. The first bit of the MAC source address is defined as the Routing
           Information Indicator (RII).  When the RIF is null, the RII is set to 0.  When the
           RIF is not null, the RLL is set to 1. The first two octets are the Routing Control
           field (RC) and the remainder of the RIF contains Routing Descriptors (RD). Each
           RD is two octets in length, therefore a maximum of 14 RD's can exist in a RIF.!;;
REGISTERED AS {iso(1)member-body(2) us(840)ieee802-
2(10032)attribute(7)rif(107)};

sRFPDUCounter ATTRIBUTE
  WITH ATTRIBUTE SYNTAX LLCDefinitions.Counter2;
  MATCHES FOR EQUALITY
  BEHAVIOUR
   sRFPDUCounter-B BEHAVIOUR
      DEFINED AS ! A value which represents the number of frames with routin field.!;;
REGISTERED AS {iso(1)member-body(2) us(840)ieee802-
2(10032)attribute(7)srfpducounter(108)};
```

querycounter ATTRIBUTE
  WITH ATTRIBUTE SYNTAX LLCDefinitions.Counter;
  MATCHES FOR EQUALITY
  BEHAVIOUR
   queryCounter-B BEHAVIOUR
      DEFINED AS ! A value which represents the number of Route Query commands
                  generated.!;;
REGISTERED AS {iso(1)member-body(2) us(840)ieee802-
2(10032)attribute(7)querycounter(104)};

--Actions—

readCounters-AC ACTION
  BEHAVIOUR
    readCounters-B BEHAVIOUR
            DEFINED AS !This action will be sent when management request
information
                  concerning RDE counters.!;;
REGISTERED AS {iso(1)member-body(2) us(840)ieee802-2(10032)action(9)
readcounters(5)};

flushRoute-AC ACTION
  BEHAVIOUR
    flushRoute-B BEHAVIOUR
            DEFINED AS !This action will be sent when management request an RDE
Pair to flush its route. This does not cause the RDE to discover a new route.!;;
REGISTERED AS {iso(1)member-body(2) us(840)ieee802-2(10032)action(9)
flushroute(6)};

7.2 ASN.1 Definitions

These definitions are presented as a continuation of the above example. It is proposed these definitions be represented in a separate section of the document.

```
Counter ::=SEQUENCE {
        counterInitTime         [0] TimeStamp,
        counterValue            [1] CounterN}

Counter2::=INTEGER  - - 64 bit counter

RIF ::=SEQUENCE {
        routingControl          [0] RoutingControl,
        routingDescriptors      [1] SET OF RoutingDescriptors}

RoutingControl::= OCTET STRING(SIZE(2))

RoutingDescriptors::= OCTET STRING(SIZE(2))
```