Inclusion of Expert Rules into Normalized Management Models for Description of MIB Structure

Antonio Martin and Carlos Leon

Dpto. Tecnología Electrónica. University of Sevilla
Spain

1. Introduction

With the technology of telecommunication network developing and the size of communication network enlarging constantly, network management needs additional requirements. To satisfy these needs of quality and performance, it is necessary to achieve an attended management network by advance software. The Artificial Intelligent is incorporated to the network management to provide the task of control and administration.

OSI and Internet are dominant network management models, which have been used for administration and control of the most of existing networks. Network management activities, such as fault, security, performance, etc. have performed for network management models. These activities are becoming supplementary functionalities in the network management, direct expert system involvement. Traditional intelligent management technologies can play an import role in the problems solving and reasoning techniques employed in network management. However, these systems are not flexible enough for today's evolving network needs.

It is necessary to develop new models, which offer more possibilities. We propose a hybrid solution that employs both managed model and AI reasoning technique for the management of heterogeneous distributed information networks. We present a new concept called Integrated Management Expert System that employs both managed model and AI reasoning techniques for the intelligent management of heterogeneous networks. This new paradigm contains aspects which clearly make the difference from the former management techniques which uses separately expert systems and management platforms.

We propose the normalization of the knowledge management necessary to administrate the resources that exist in the networks independently from the builder of the management resources. The goal is to get a syntactically uniformed definition of all the management knowledge supplied by the expert, independently from the maker. The novelty comes from the fact that the employed knowledge for the networks management (conditions and management operations to achieve on the different resources) is included and normalised itself in the definitions of the network elements and is treated as if it were a propriety.

So, we realise syntactically uniformed normalization of intelligence applied to the management. This technique integrates the Expert System within the Management Information Base. The advantage is that a large problem can be broken down into smaller

and manageable sub-problems/modules. For this goal, an extension of OSI and SNMP management framework specifications language has been investigated. A new property named RULE has been added into the Management Information Base (MIB), which gathers important aspects of the facts and the knowledge base of the embedded expert system.

2. Related work

The work that we present has two parts, figure 1. In the first part we analyse the current management models, its evolution and the applications of the expert systems in the network management. It offers a general vision of the traditional management expert system, analyzing its deficiencies and discovering the needs that push us toward new management paradigms.
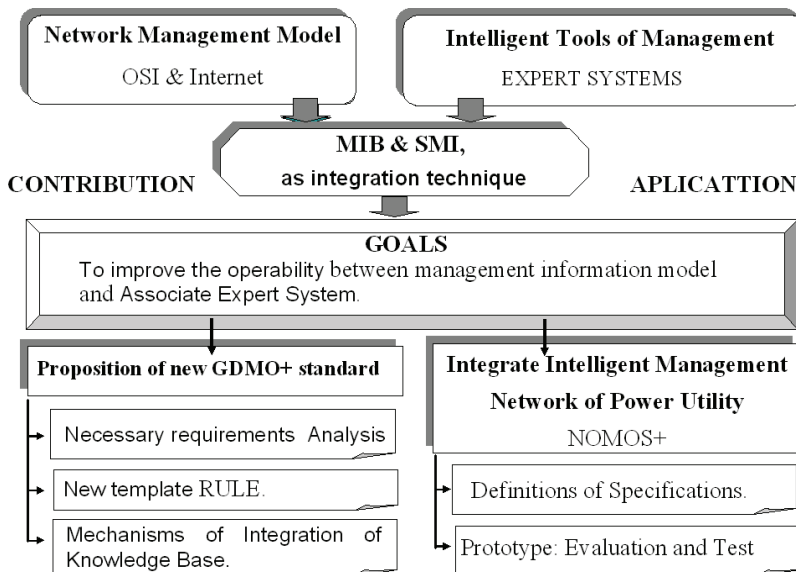

Fig. 1 Synopsis of the research.

The second step in the modelling sequence includes an introduction to standard Structure of Management Information (SMI) and intelligent management, showing the advantages and problems of integrated intelligent management. Next we present the extension of the standard GDMO (ISO/ITU-T, 1993), to accommodate the intelligent management requirements. We describe how the templates are modified using the new extension called GDMO+ and present the new template RULE, which defines the knowledge base of the management expert system. We study the characteristics of this new property RULE: template structure, behaviour, priority, inheritance of expert rules, etc.

As the last step in order to show the viability of our proposal, we perform a practical demonstration in which the information and the management knowledge is unified in a unique specification. We present a tool based on a GDMO+ standard. It shows an expert system prototype of integration rules. This prototype gives service at the moment to a power utility.

3. Network management

Once we have presented the scope and goals of the work, this section presents most important management models and we make a brief overview of relevant topics of the network management. There are several organizations which have developed services, protocols and architectures for network management. At the moment there are two main management models for computer communication: Open System Interconnection OSI and Internet. Those are standardized models for computer communication and starting point for understanding network management. It provides for common understanding between communicating processes in a heterogeneous environment, servers as the basis for a precise specification of network services and protocols and constitutes a vendor-independent concept (ITU-T, 1992).

ISO was the first which started, as part of its Open Systems Interconnection (OSI) program. The term OSI systems management actually refers to a collection of standards for network management that include a management service and a protocol. Common Management Information Protocol (CMIP), which provides the information exchange capability to support the Common Management Information Service (CMIS) and offers management services to management applications.

ISO issued a set of standards and draft standards for network management. A subset of these standards provides the foundation for Telecommunication Management Network (TMN) developed by the organization called Telecommunication Union (ITU). TMN conceptually a separate network that interfaces a telecommunications network at several different points (ITU-T, 1996).

Internet Model by the Internet Engineering Task Force (IEFT). Internet model is a structured and standardized approach to Internet management. This uses the Simple Network Management Protocol (SNMP) (Black, 1995), figure 2.
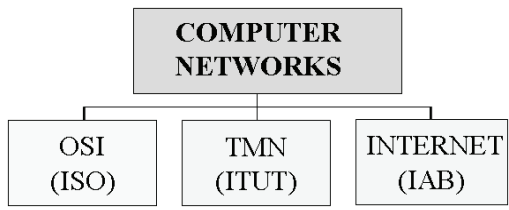

Fig. 2. Management Models

3.1 Network management elements

These network management systems operate using client/server architecture. Four fundamental concepts of these models are (ISO/ITU-T, 1998):

- Manager or Manager Role: In the network management model a manager is an unit that provides information to users, issues requests to devices in a network, receives responses to the requests and receives notifications.
- Agent or Agent Role: It is an unit that is part of a device in the network that monitors and maintains status about that device. It can act and respond to requests from a manager.
- Network Management Protocols: Managers and agents require some form of communication to issue their requests and responses. The Common Management

Information Protocol (CMIP) is the protocol used in management model ISO and TMN. SNMP is the protocol used to issue requests and receive responses in a management model Internet. The combination of Internet and OSI management requires the use of different network management protocols working with different levels of modelling complexity. CMIP requires the use of full OSI stack, for implementation while the SNMP protocol operates with the lower layers of stack. CMIP and SNMP have been designed to scale as network grows, i.e. the ability to perform "manager" or "agent".

- Management Information Base (MIB): In this information Base, we add the knowledge management. In addition to being able to pass information back and forth, the manager and the agent need to agree on and understand what information the manager and agent receive in any exchange. This information varies for each type of agent. The collection of this information is referred to as the management information base. A manager normally contains management information describing each type of agent the manager is capable of managing. This information would typically include ISO and Internet MIB definitions for managed objects and agents (Morris, 2003).

3.2 Management information

Information modelling plays a large part in modern network management systems. It is a way to represent the capabilities of a Managed object interfacing directly to the equipment's native command. This abstraction has given rise to management information models. This is the information associated with a managed object that is operated on by the OSI and Internet Management protocols to control and monitor that object.

The description of management information has two aspects. First, a Structure of Management Information (SMI) defines the logical constitution of management information and how it is identified and described. Second, the MIB, which is specified using the SMI, defines the actual objects to be managed. The MIB can be seen as kind of database. It is a conceptual repository of management information.

The elements that make up a management information model are referred to as managed objects. The content of this database is not set of managed objects themselves, but the information that is associated with the managed objects. These resources are viewed as 'managed objects'. A managed object is the abstract view of a logical or physical system resource to be managed. Thus, a managed object is the abstraction of such a resource that represents its properties as been by, and for purpose of, management. These special elements provide the necessary operations for the administration, monitoring and control of the telecommunications network. A managed object is a management view of a resource that is subject to management, such as a layer entity, a connection or an item of physical communications equipment. Managed objects are used for management functions, performance monitoring and analysis, and the setup and teardown of virtual circuits and virtual paths.

The managed object concept is refined in a number or additional standards, which are called the Structure of Management Information (SMI) standards. The managed objects are defined according to the SMI, which defines how network objects and their behaviour are to be specified, including the syntax and semantics (Clemm, 2006).

In the Open System Interconnection (OSI) and TMN systems management the information architecture is based on an object-oriented approach and the agent/manager concepts that are of paramount importance. In OSI, SMI provides the Guidelines for Definition of Managed Objects (GDMO), for definition objects contained in the MIB.

Internet management model doesn't use the Object Oriented Programming such as it is used by the OSI model. This is one of the reasons for the Internet model simplicity. The definitions contain objects, specified with ASN.1 macros. We are studying the way to integrate the expert knowledge in the management Internet model. The resources specifications can only be groups of scalar variables and cells tables in spite of not being an Object Oriented Programming model, we can use the tables as classes where the attributes are the table columns and every file contains an instance of the class. The same as in OSI every object has an OID associated identifier.

4. Overview on expert systems and telecommunications management

After this analysis of management elements in common to OSI an Internet models we present an overview of the state art in traditional expert management system. ISO classifies the systems management activities into five functional areas: fault management, accounting management, configuration management, performance management and security management. This specific functions performed by OSI are also applicable in the Internet model. We can categorize the expert systems within these five groups. In the next table 1 we can observe a synopsis of some expert systems applied of network management. This table indicates the applications area and technique use.

	MANAGEMENT DOMAIN					
TECHNIQUE	**Fault**	**Accounting**	**Configuration**	**Performance**	**Security**	**Expert Interface**
Expert Rules	Max & Opti ANSWER	ESS-ES ECXpert	ACE XCON NMCS	TASA	NIDES P-BEST NIDX	NetHELP
Bayesian Network	Trouble Locator	APRI				
Case-Based Reasoning (CBR)	CRITTER		EXSim	NETTRA		
Blackboard				Scout		

Table 1. Expert system applications developed for network management.

We can observe that most of expert systems are built for fault management and use models based in expert rules. These rules obtain an experience accumulated by human experts. The last years new techniques are applied: Bayesian networks, blackboard, the systems based in case based reasoning, which make an approach to most similar case of the all cases existing in the knowledge base, etc. (Negnevitsky, 2002).

In traditional intelligent management these expert systems can be treated as a black box which receives the events coming from the elements that compose the managed system. There must exist between them a certain level of compatibility and interoperability that makes sure that the stream of data and information control flow in both directions. An unique interface must be developed, according to the possibilities and tools that the system possesses for the external communication with these applications: sockets, CORBA, etc.

The traditional expert management presents inconvenients, which show the limitations of current systems management. Some disadvantages of traditional Expert Management are the next:

- There are disadvantages caused by the existence of two independent elements, the management platform and the expert system, with a permanent and continuous dialogue between them.
- Restrictions may appear when choosing a determined managed platform.
- This presets incompatibility problems between management platform and expert system.
- Because of the need to use heterogeneous networks, it is difficult to normalize the knowledge for an adequate management of resources.

This disadvantage can be solved by using the integration management that we propose. In the integrate intelligent management, it is not necessary to interconnect the expert system and the management platform. Both components are defined using the same syntax, being finally both completely integrated in some unique specifications.

In next sections we will approach our research in the integration of knowledge management into MIB of OSI and Internet management models.

5. Knowledge integration in OSI and internet models

This paper focuses on a framework and a language for formalizing knowledge management descriptions and combining them with existing managed object definitions. To find a solution, we must develop an interface that makes it possible to introduce expert rules of management within the definition of the MIB objects. This is a solution that will be generally more flexible than the one found by traditional methods (formal language). The solution is the inclusion of formal knowledge descriptions in regular RULE-templates.

An essential part of the definition of an intelligent managed object is the relationship between their properties and the management knowledge of the resource. This relationship is not modelled in a general way. Unfortunately, the knowledge of managed objects is defined in another way using programming language. This results in resources specifications which are not often information about the knowledge base of expert system, and increasing the possibility of different intelligent implementations not being interoperable. To achieve consistent, clear, concise, and unambiguous specifications, a formal methodology has to be used.

OSI and Internet Management models are faced with several impediments to improve the integrated intelligent management. Management models explain how individual management operations should be performed. The current management standards do not specify however the sequence in which intelligent operations should be performed to solve specific management problems. OSI and Internet management is rather complicated. The model has introduced several new concepts, which are sometimes difficult to comprehend. This makes if difficult the construction of intelligent platforms that work with the model. The basic questions raised are:

1. Formulate, describe, distribute and assign the knowledge between the different intelligent agents defined.
2. Communicate the knowledge between different objects and what communications language or protocol must they use.
3. Ensure that the objects act coherently when they make a decision or conduct.

4. It enables the objects to be reason about the actions, plans and knowledge of other objects.

To make this possible, it is essential to explore the capacities that have the management information models. In particular, the OSI GDMO standard specifies the proprieties and characteristics that make easy the inclusion of the expert management rules as a part of the defined managed resources. In this way the intelligent network managers can interpretate the rules. Finished by the expert system and achieve an intelligent treatment of the management information. The integrated framework that we propose has numerous advantages. Among them:

-   Abstraction of the users from the management systems and good compatibility between different management platforms.
-   The ability of different network management system to communicate the knowledge management of the expert systems without overloading the management applications.
-   The flexibility to define new managed objects than contain knowledge management without modifying all the system that needs to interact with them.
-   Possibility of standardization of the new expert rules, by an organism such as the ISO and a common set of network management knowledge based in standards.
-   Reusability of the knowledge management. GDMO has an object oriented syntax. Its classes inheritance mechanisms allows management expert rules reuse.
-   Easily to build new expert management systems. A general top-down view of the integrated multi-vendor network and structured list of network management objects that contains some set of expert rules for the administration.

This structure also provides a richer level of abstraction facilitating the coexistence knowledge management, allowing different levels of modelling complexity, and organizing the knowledge management of the managed objects, figure 3.
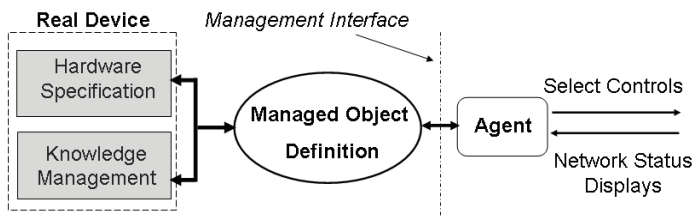

Fig. 3. Integration of Knowledge management and Resources properties in one single specification.

6. Including formal knowledge management in OSI

To allow deployment of equipment from different vendors, the managed models, OSI framework define the language GDMO (Guidelines for the Definition of Managed Objects). GDMO has been standardized by ITU (International Telecommunication Union) in ITU-T X.722 and is now widely used to specify interfaces between different components of the TMN architecture. This section introduces a framework for the inclusion of formal Knowledge management descriptions into GDMO specifications and focuses on the syntax and semantics of the language GDMO and the extension GDMO+.

According to OSI's Management Information Model, the management view of a managed object is viable at the managed object boundary. Managed object can be viewed as mediator

between the network management interface and the hardware in the networks. Managed Object is modelled by attributes, actions and notifications, figure 4.
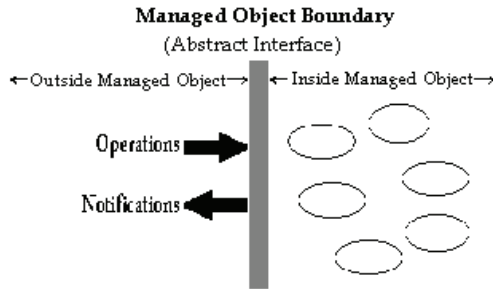

Fig. 4. Managed Object Boundary

They can usually represent a certain part of the internal state of an element. Actions invoke certain functions which a device can perform. Notifications are spontaneous messages emitted if certain events occur. Managed objects which are alike are grouped together to form Managed Object Classes. Classes can inherit their appearance from other classes and add new features (Goleniewski & Jarrett, 2006).

The appearance of managed objects can be formally described using the language GDMO. This language defines a number of so-called templates, which are standard formats used in the definition of a particular aspect of a real device in the networks. A complete object definition is a combination of interrelated templates. The nine templates that conform the actual GDMO standard are next:

- Managed Object Class Template references all other templates either directly or indirectly to make up the managed object class definition: Package, Parameter, Attribute, Attribute Group, Behaviour, Action and Notification.
- Package Template, defines a combination of behaviour definitions, attributes, attribute groups, operations, actions, notifications and parameters for later inclusion in a managed object class template. A package can be referenced by more than one managed object class definition.
- Attribute Template defines individual attribute types used by managed object classes. A single attribute can be referenced by more than one managed object class definition. If desired these attribute types can be combined in a group by using the Attribute Group template.
- Action Template defines actions that can be performed by a managed object class. These actions can be executed by using the Common management information service (CMIS). In particular the M-ACTION service that requests an action to be performed on a managed object. An action can be referenced by more than one managed object class. The Action template defines the behaviour and syntax of an action. The syntax definitions specify the contents of the action information and action reply fields in CMIS action requests and responses.
- The Notification template defines the behaviour and syntax of a notification that can be emitted by a managed object class. A notification can be referenced by more than one managed object class. The syntax definitions specify the contents of the event information and event reply fields in CMIP, event report requests and responses.

- The Parameter template specifies and registers the parameter syntaxes and associated behaviour that may be associated with particular attributes, operations, and notifications within Package, Attribute, Action, and Notification templates. A parameter can be referenced by one or more of each of these templates. The type specified in a Parameter template is used to fill in the ANY DEFINED BY oid construct.
- Attribute Group Template defines one or more attributes that can be referenced as a group. A managed object class definition can include all attributes of a group by referencing the group, rather than referencing each attribute individually. More than one managed object class definition can reference an attribute group. Attribute groups make it easier to collectively perform operations on a large number of individual attribute.
- Behaviour Template describes the expected behaviour of another element of the standard: Managed object classes, Name bindings, Parameters, Attributes, Actions and Notifications. The behaviour may be defined by readable text, high level languages, formal description techniques, references to standard constructs, or by any combination of the preceding definition methods.
- Name Binding Template, specifies instantiation and legal parameters for managed objects. Containment, creation, and deletion constraints are initiated from this template.

The elements that at the moment form the GDMO standard do not make a reference to the knowledge base of an expert system. To answer these questions, it will be necessary to make changes on the template of the GDMO standard. We present an extension of the standard GDMO, to accommodate the intelligent management requirements, figure 5.
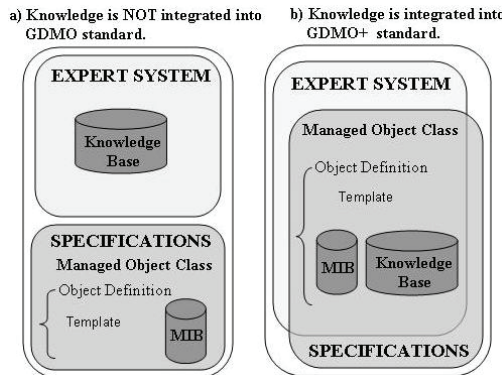

Fig. 5. Extension of GDMO+

Before considering a concrete language for specifying managed object's knowledge, a methodology for combining knowledge management with GDMO+ definitions is presented. To improve the quality of the descriptions and the resulting implementations, a formal method for specifying knowledge is desirable. We describe how to achieve this goal using a new extension called GDMO+. An object-oriented logic programming language is presented, which can be used in conjunction with the framework to specify the management knowledge of a managed object. The methodology is independent of the language used and can be combined with other approaches for formalizing knowledge It is based on the notion of events and supports the object-oriented features of GDMO (Hebrawi, 1995).

Management knowledge is introduced in GDMO+ which defines a number of new templates that contain certain aspects of the expert rules. This extension presents a new element RULE, which defines the knowledge base of the management expert system. This template groups the knowledge base supplied by an expert in a specific management dominion. It allows the storage of the management knowledge in the definition of the resources that form the system to be managed, figure 6.
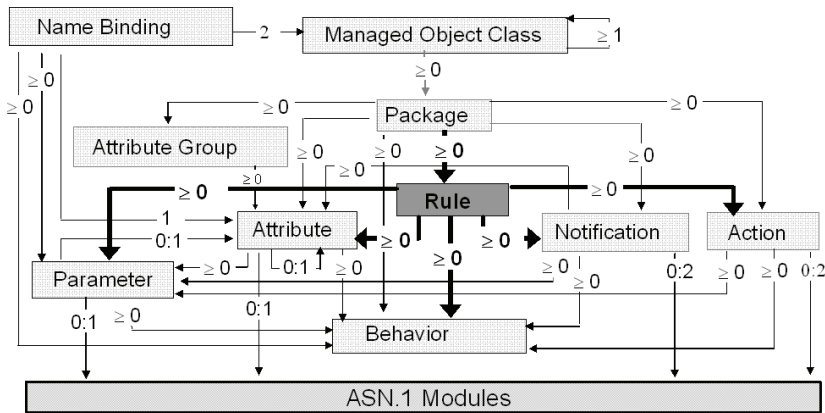

Fig. 6. Relations between proposed standard templates

The standard we propose contains the singular template RULE and its relations to other templates. Two relationships are essential for the inclusion of knowledge in the component definition of the network: Managed Object Class and Package Template. In the standard we propose, both templates have the new property RULES. Let us study both relationships.

7. Template for management of object classes

This template is used to define the different kinds of objects that exist in the system.

```
<class-label> MANAGED OBJECT CLASS
    [DERIVED FROM  <class-label>  [,<class-label>]*;]
    [CHARACTERIZED BY <package-label> [,<package-label*;]
    [CONDITIONAL PACKAGES <package-label> PRESENT IF condition;
                              ,<package-label>] PRESENT IF condition]*;]
   REGISTERED AS object-identifier;
```

DERIVED FROM plays a very important role, when determining the relations of inheritance which makes it possible to reutilize specific characteristics in other classes of managed objects. In addition, a great advantage is the reusability of the object classes and therefore of the expert rules which are defined.

This template also can contain packages and conditional packages, including the clauses CHARACTERIZED BY and CONDITIONAL PACKAGES. A package is used to define a package that contains a combination of many characteristics of a managed object class: behaviours, attributes, groups of attributes, operations, parameters, actions and notifications. The structure of package template is next:

```
<package-label> PACKAGE
    [BEHAVIOUR <behaviour-label> [,<behaviour-label>]*;]
    [ATTRIBUTES <attribute-label>propertylist[,<parameter-label>]*
                [,<attribute-label>propertylist[,<parameter-label>]*]*;]
    [ATTRIBUTE GROUPS <group-label> [<attribute-label>]*
                    [<group-label> [<attribute-label>]*]* ;]
    [ACTIONS <action-label> [<parameter-label>]*
            [<action-label> [<parameter-label>]*]* ;
    [NOTIFICATIONS <notification-label> [<parameter-label>]*
                [<notification-label> [<parameter-label>]*]* ;]
    [RULES    <rule-label>  [,<rule-label>]*;]
    REGISTERED AS object-identifier;
```

In addition to the properties indicated above, we suggest the incorporation of a new property called RULES, which contains all the specifications of the knowledge base for the expert system. Next definition shows the elements of a package template, in which it is possible to observe the new property RULES.

8. Expert rule template

The RULE template permits the normalised definition of the specifications of the expert rule to which it is related. Figure 7 shows a UML representation of Rule template.
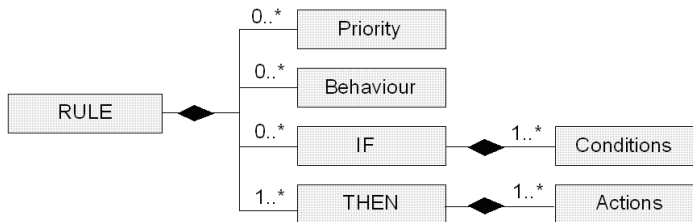

Fig. 7. RULE template in GDMO+ standard.

This template allows a particular managed object class to have properties that provide a normalised knowledge of a management dominion. A rule is an expression such as: "If the antecedent is true for facts in a list of facts, then it can carry out the actions specified in consequent" (Brachman, 2004). Each of the template type consists of a specification label, a template type, a list of keywords and an unique ASN.1 object identifier. The structure of the RULE template is shown here:

```
<rule-label> RULE
    [PRIORITY       <priority> ;]
    [BEHAVIOUR <behaviour-label> [,<behaviour-label>]*;]
    [IF occurred-event-pattern [,occurred-event-pattern]*]
    [THEN    sentence [, sentence]* ;]
    REGISTERED AS object-identifier;
```

The first element in a template definition is headed. It consists of two sections: <rule-label> this is the name of the management expert rule and RULE, a key word indicates the type of

template. After the head, the following elements compose a normalised definition of an expert rule. After the head, the following elements compose the RULE template.

-   BEHAVIOUR: This construct is used to extend the semantics of previously defined templates. It describes the behaviour of the rule. This element is common to the others templates of the GDMO standard.
-   PRIORITY: This represents the priority of the rule, that is, the order in which competing rules will be executed.
-   IF: We can add a logical condition that will be applied on the events occurred or their parameters. It contains all the events that must be true to activate a rule. Those events must be defined in the Notification template. The occurrence of these events is necessary for the activation of the rule and the execution of their associated actions. For the establishment of the conditions of fire, we must following structure:

    [IF <patter> [, <patter>]* <condition> [,condition]*;]

-   THEN: This gives details of the operations performed when the rule is executed. Those operations must be previously defined in the Action template. These are actions and diagnoses that the management platform makes as an answer to network events occurred.

Our formal structure is next: [THEN <sentence> [,<sentence>]*;]

-   REGISTERED AS is an object-identifier: A clause identifies the location of the expert rule on the ISO Registration Tree. The identifier is compulsory.

To finish this section next paragraph shows an expert rule integration; it defines the managed object of a resource port that belongs to a switch.

```
port MANAGED OBJECT CLASS
    DERIVED FROM "element_switch":top;
    CHARACTERIZED BY portPackage PACKAGE;
REGISTERED AS {nm-MobjectClass 1};

portPackage PACKAGE
    ATTRIBUTES portNum GET, portStatus GET,  …
    OPERATIONS diagnose, disconnect, connect, …
    NOTIFICATIONS portFailure, portInitializated…
    RULES transmissionError, powerError, …,
REGISTERED AS {nm-package 1};

transmissionError RULE
    PRIORITY 4;
    BEHAVIOUR transmissionErrorBehaviour;
    IF (?date ?time1 ?station portFailure ?portNum ALARM)
       (?date ?time2 ?station portFailure ?portNum ALARM
       & : (=(portstatus(? ?time1 ?time2)) OFF)
    THEN ("Severity:" PRIORITY) ,
       ("Diagnostic: "It damages in the modulate transmission of port", ?portNum),
       ("Recommendation "Revision port"),
       ("Actions "put off/on port");
REGISTERED AS {nm-rule 2};
```

In the previous example a Class of Managed Object is port, which defines the properties corresponding to the port of a switch. This class includes the compulsory portPackage

which contains all the specifications corresponding to the device. We can indicate which re the expert rules that have been associated with the defined class by means of the RULES clause. The rules are defined by using the RULE template. These rules detect anomalies or defects of operations produced in the port and suggest the necessary measures for solving the problem. The first rule transmisionError is in charge of detecting failures in the data transmission module and gives recommendations on how to solve this failure.

9. New expert management rules identification

The integration of the management knowledge in the proposal GDMO+, implies that the expert rules are defined together with the specifications of the managed objects. As the rest of the proprieties of the GDMO+ standard that are included in the definition of a managed object class: classes, attributes, actions, etc., the expert management rules will be attached to revision convention and analogue register structures. The CCIT Rec. X.208 | ISO/IEC 8824, provides a structure for the object identifies and the proprieties that composed it, as soon as the new property called RULES.

To make this possible, the systems management functions, function(2) and the management information model smi(3), must allow an amplification, so that these recommendations can pick up functions and mechanisms of definition that allow to contain the new property RULES. For this purpose we propose new versions of these recommendations. In the next figure 8 we show the amplification that shows the sections function(2) and smi(3) and the new elements RULES(11) is presented and gathers all necessary elements to define an associated knowledge.
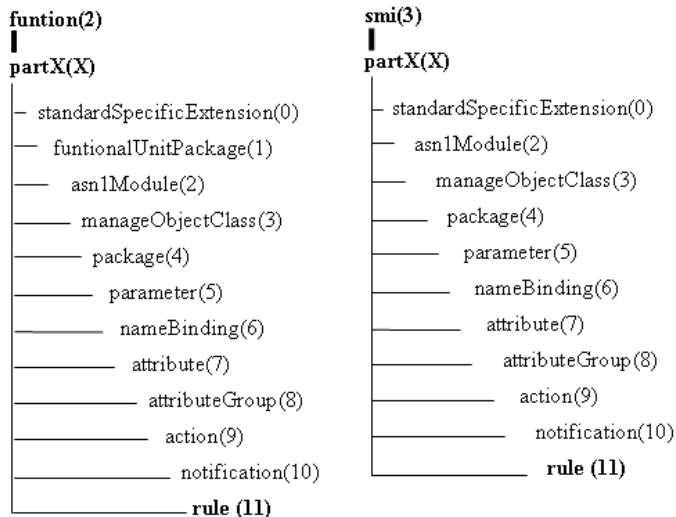
```
funtion(2)                              smi(3)
▌                                       ▌
partX(X)                                partX(X)

─ standardSpecificExtension(0)          ─ standardSpecificExtension(0)
── funtionalUnitPackage(1)              ── asn1Module(2)
─── asn1Module(2)                       ─── manageObjectClass(3)
──── manageObjectClass(3)               ──── package(4)
───── package(4)                        ───── parameter(5)
────── parameter(5)                     ────── nameBinding(6)
─────── nameBinding(6)                  ─────── attribute(7)
──────── attribute(7)                   ──────── attributeGroup(8)
───────── attributeGroup(8)             ───────── action(9)
────────── action(9)                    ────────── notification(10)
─────────── notification(10)            ─────────── rule (11)
──────────── rule (11)
```

Fig. 8. Amplification of the Model of Information and Management Functions

In this way, the definition of a new rule "New_Rule" would be included in the place indicated to the recommendation X.720 of CCITT | ISO 10165-1 [ISO92B]. In this case it is to add a new management rule to the Standard GDMO+ proposal. Extended, it would be achieved in the following level: {join-iso-ccitt ms(9) smi(9) part4(4) rule(11) New_Rule(45)}

10. Integrate environment for internet network management.

This section discusses and analyzes the intelligent management approach that is standardized by the Internet Engineering Task Force. Internet management can be compared to OSI management model. In fact, Internet management uses many of the concepts that existed in OSI at the time SNMP started. As a result, the remarks that were made above are to some extent also applicable to Internet management. As opposed OSI management, however, Internet management uses only a small part of management functions for exchange of management information.

Note that objects in the Internet and those OSI are different. Internet objects are similar to attributes in a OSI managed object, and an Internet object group can best be described as analogous to a OSI management object class. In the Internet, an object is more like a variable found in programming languages; it has a syntax and semantics. Each object can have one or more object instances, each of which, in turn, has one or more values.

An interesting difference between Internet and OSI is that the Internet management model is more simple that OSI model. The principal characteristics of their architecture are next:

- The cost of adding network management to existing systems is minimal.
- All systems connected to the network should be manageable with SNMP. It should be noted that SNMP protocol only defines how management information should be exchanged, they do not define which management information exits. Such information is defined by the various MIB standards.
- It should be relatively easy to extend the management capabilities of existing systems, by extending the existing MIB's or adding a new MIB.

Due to these aspects, it is questionable whether OSI management will reach the dominant market position that has originally been anticipated. Simple measures that solve all the above mentioned problems are difficult to find.

10.1 Internet extension to integrate the knowledge management

For two systems to communicate, each must understand the data sent from one to the other. This can be achieved by using a language that has the same syntax and semantics. In the application layer, we use abstract syntax, which states only how data are arranged and what meaning they have. One of the possible abstract syntaxes is abstract syntax notation one (ASN.1). Between the application layer and the presentation layer, a local set of rules can be used to transform data; however the syntax of the data transferred between presentation entities must be understood by each end. This is known as transfer syntax. Abstract syntax and transfer syntax are negotiated at the beginning, during association time.

One transfer syntax is Basic Encoding Rules (BER). BER state how data must be transferred to the other presentation entity. The local syntax can be purely dependent on the local protocols used, in this case SNMP. Figure 9 illustrates the concept of abstract syntax and transfer syntax. ITU-T Recommendations X.208 and X.680 describe standardized ways and steps to define data types and data values (Douglas & Kevin, 2005).

The textual MIB representation is called module and consist on a plane text file. This file is used as a subset language of ASN.1 and based in the SMI specifications. The MIB-II is the most important and probably best known MIB. It contains all the variables to control the major Internet protocols: IP, ICMP, UDP, TCP, EGP and SNMP. The structure of this MIB is simple; all management variables that belong to the same protocol are grouped together. Within a protocol group there is hardly any additional structure that helps understanding the various variables within that group.
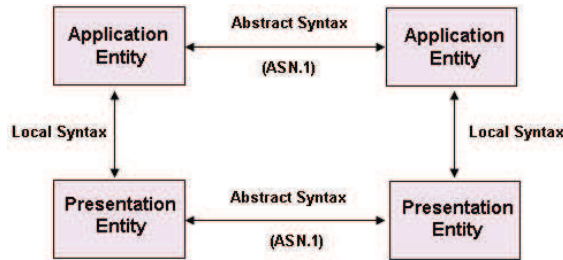
Fig. 9. Abstract Syntax Notation and Basic Encoding Rules

Soon after definition of the MIB-II other MIBs, some of these standardized MIB`s are FDDI, ATM, X.25, X.500 Directory Monitoring, etc. Next to the standardized MIBs there are also a large number of enterprise specifics MIBs. The objects to be managed in Internet must follow a certain set of rules as mentioned in the Structure of Management Information (SMI) such as an object defined by the X group is compatible with the definition of the object by the Y group.

We propose to add a new type named RULES and the incorporation of an extension of the MIB II named MIB II+. We broadly classify the ASN.1 built-in types as follows: simple types, structured types, Tagged types and subtypes. We introduced in ASN.1 a new concept denominated Expert Rule to definition the new group RULE existing in MIB II+. This group is introduced as a textual convention in this MIB II+ document. Group RULES contains all the aspect related with the expert management, figure 10.
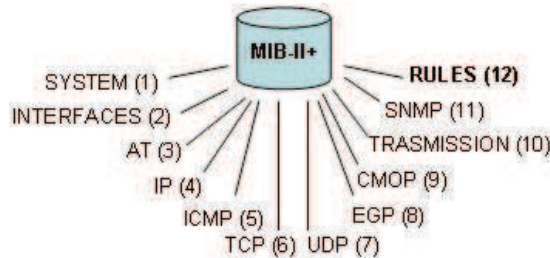

Fig. 10. MIB-II+ Objects group

To define an expert rule we use the modules definitions. Modules definitions are primarily used for grouping ASN.1 definitions. They also help in using type definitions characterized in the other places by making use IMPORT and EXPORT mechanisms. Modules are analogous to functions in C language or subroutines in PASCAL. There are module definitions in the definitions of managed object classes in standards and the others documents. The macro used for MIBs definition in SNMP was defined in RFC 1155 draft (Structure of Management Information) and later extended in the RFC 1212 (Concise MIB Definitions). RFC 1155 version is used to define objects in MIB-I. RFC 1212 version including more information and is used to define objects in MIB-II. Next definition shows OBJEC-TYPE macro in RFC 1212.

These can be enhanced by including formal descriptions. In this case, the specifications formal parts and the knowledge must be distinguishable. An easy solution is the separation by keywords. Next section shows an example of definition of an integrate expert rule in Internet model:

```
transminissionError OBJECT-TYPE
    SYNTAX SEQUENCE
    {conditionRule COMPONENTS OF transminissionErrorCondition,
    actionRule COMPONENTS OF transminissionErrorAction}
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
 "Rule devoted to the detection of errors in the data transmission module"
 ::= { tcp 13 }

trastimissionErrorCondition OBJECT-TYPE
    SYNTAX Condition
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION "Information about a particular condition"
    INDEX { date, time, local, signal, remote, alarm, operator}
::= { tcpConnTable 1 }

TcpConnEntry ::=
    SEQUENCE { date INTEGER,
    time TIMETICKS,
    local OCTETSTRING,
    signal OCTETSTRING,
    remote OCTETSTRING,
    alarm OCTETSTRING,
    operator OCTETSTRING}

 transminissionErrorAction OBJECT-TYPE
     SYNTAX OCTETSTRING
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION  "Action execute when fire the expert rule"
 ::= { tcp 13 }
```

This example defines a portion of the Management Information Base (MIB) for use with network management knowledge in TCP/IP-based internets. In particular, it defines objects for managing remote network monitoring devices. This standard extends that specification by documenting the knowledge management in SMIv2 format. These groups are defined to provide a means of assigning managed object, and to provide a method for implementers of managed agents to know which objects can be administrator.

11. Case of study and practical experiments

At present, some reports of experience with the OSI and expert system architecture are available, although the first OSI-based platforms are already on the market. We provide a rule-based expert system applied to the fault diagnosis in telecommunication system of a power utility (Maggiora et al., 2000). The communications systems employed to implement the integrated intelligent management prototype belongs to the SEVILLANA-ENDESA a major Spanish power utility. This Telecommunications network is made of several

equipments and systems, which differ from each other in terms of age, technology, network domain, etc. Each kind of equipment communicates to a particular supervisory system, which is responsible the equipment's operation and maintenance information to one or more operations centers. The supervisory are also responsible for sending commands from the operations' staff to the supervisory equipment. A supervisory system as composed from two kinds of equipment: Remote Terminal Units (UTR) and central units (UC). UTRs function is to gather, code and transmit information to the managed equipment and distribute command signals to those equipment (Boyer, 1999), figure 11.
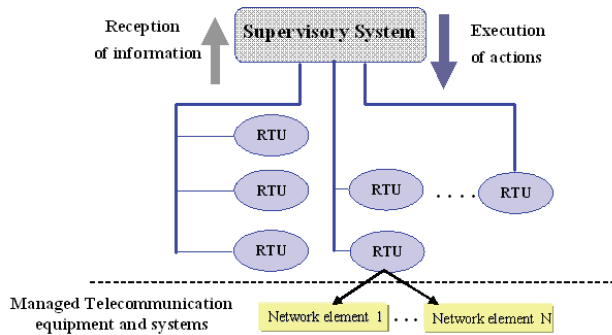

Fig. 11. Network framework

Management and control of that network as based on an Expert System called NOMOS+ developed by the Electronic Technology Department in the University of Seville. This tool understands transceivers and multiplex equipment. This expert system that integrates the knowledge base and the resources definition into an unique specifications. The knowledge base of this system is integrated in the specifications of the resources using for that purpose our GDMO+ proposal. It has been employed Workstation to program the expert system. The resultant expert system has about 200 rules.

NOMOS+ is implemented in Brightware's ART*Enterprise, an expert system shell. ART*Enterprise is a set of programming paradigms and tools that are focused on the development of efficient, flexible, and commercially deployable knowledge-based systems. Expert system shells simplify developer interactions by eliminating the developer's concern with operating system requirements. Its use can therefore reduce the design and implementation time of a program considerably.

11.3 Expert system architecture

The integrate expert system we propose is composed of three major components: a knowledge base, an inference engine and a user interface, figure 12.

The knowledge base is the core of the system, which is a collection of facts and if-then production rules that represent stored knowledge about the problem domain. The inference engine is the processing unit that solves any given problems by making logical inferences on the given facts and rules stored in the knowledge base. In our tool we used the ART*Enterprise. The user interface controls the inference engine and manages system input and output. The user interface of our tool contains a preprocessor for parsing GDMO+ specification files, a set of input and output handling routines for managing the system. Also, the user interface components allow administrators to inspect the definitions of

management object classes interactively (Giarratano & Riley, 2005). When new knowledge is uncovered, it will need to be incorporated into the system to keep it updated. The user interface allows to modify or include new expert management rules in the managed objects definition. The System Management window allows a system administrator to perform the following functions:

- Adding new Management Object Class and change the definitions of the existing management object class.
- Configure the Nomos+ workstation's for the appropriate function and actions.
- Set up alarm actions to be executed automatically when specific alarms occur.
- Examine and modify the GDMO+ specifications and log files, configuration files, and other text files using an online visual editor.
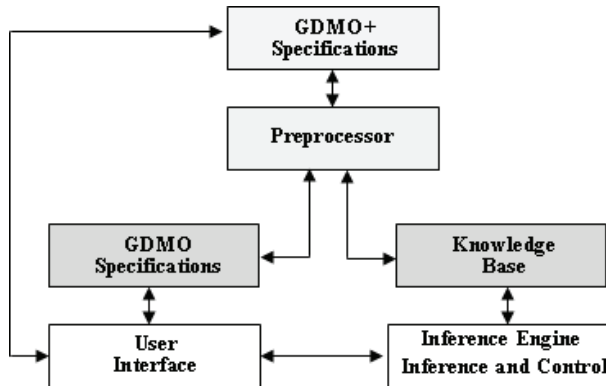

Fig. 12. Elements of the prototype NOMOS+

Since the window is used to modify system files, only one administrator can use it. Figure 12a shows an example of a web based management interface where the system Object Class of a network is defined in NOMOS+. There is an Object Class called muxtiplexCTR190. If the "Modify" button is pressed then the definition of an Object Class dialog box, Figure 12b, will show. That dialog box allows describing different characteristics of an object class, such as attributes, actions and expert rules.
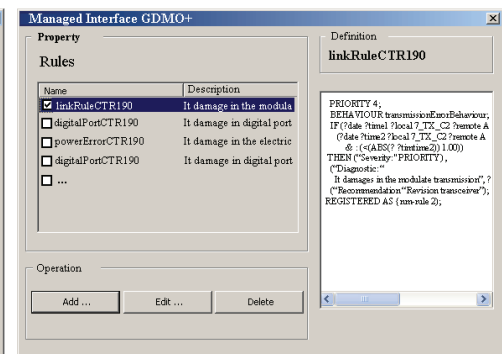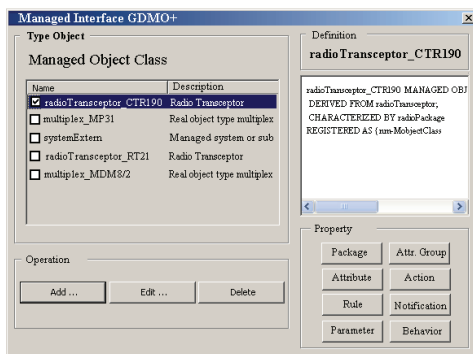

Fig. 13a System Object Classes Dialog Box   Fig. 13b System Object Classes Update

12. Conclusion

In this paper we showed possibilities to apply and integrated the artificial intelligence techniques in network management and supervision, using ISO and Internet network management models. In fact we believe that these kinds of applications underline the power of CMIS as both simple and powerful knowledge modeling languages, offering possibilities that simpler protocols such as SNMP and CMIP do not offer.

We have seen that the current management systems are not able to solve questions shown in the initial parts of this work. Until now the managed objects did not contain the knowledge coming from base of the expert systems. The managed objects are not able to use the knowledge management, which collects the management operations and control of a management domain. The point is to solve the current problem to undertake an intelligent integrated management. We offer an original contribution to include expert rules in the specifications of the network features, a new model named Integrated Intelligent Management an extension of standard GDMO+ and MIB II+.

This paper presented a language for formalizing the knowledge base of the expert systems descriptions in OSI and Internet telecommunications management network framework. A number of questions which arise when designing a language have been discussed and a general framework for the inclusion of formal knowledge management in MIB specifications has been presented. The proposed model was used to formally specify the expert knowledge. An expert system has been implemented and used to manage the specification to the language used by the simulation environment. This demonstrates that expert systems is capable of specifying the knowledge of a reasonably sized information model. A large amount of the knowledge could be described in a surprisingly short and easy to understand manner. The specification of the NOMOS+ information model showed that a large part of the knowledge management was specified in a rather imperative fashion.

Our research has demonstrated an useful and interesting modular approach in the development of a knowledge based integrated expert system which can be quite powerful in tackling the huge and enormously wide subject on diagnosis of common problems in management network. It is suggested that future work should aim to:

- Further development of this prototype system by adding more modules based on the framework provided by NOMOS+ so that more in-depth knowledge and specialized subjects may be captured; in particular the following are of great interest: Development of a design module, possibly a large system, for identifying specific areas as accounting management, configuration management, performance management and security management.
- Enhancement work in combining and integrating the various modules will be required as the number of modules increases with the growth of the knowledge base.
- Use of external programs and graphics interface to enhance the functions of the system will be desirable. A graphics interface has not been adopted in NOMOS+ but is an option that can be added in future enhancements to the system.

13. Acknowlegment

14. References

Boyer, Stuart A. (1999). *Supervisory control and data acquisition.* Research Triangle Park, NC: Instrument Society of America, cop.

Black, U.D. (1995). *Network Management Standards*. McGraw Hill

Brachman, Ronald J. & Levesque, Hector J. (2004) *Knowledge representation and reasoning*. San Francisco, CA: Elsevier/Morgan Kaufmann.

Clemm, Alexander. (2006). *Network Management Fundamentals*. Cisco Press

Douglas, Mauro and Kevin, Schmidt. (2005). *Essential SNMP*, 2nd Edition. O'Reilly.

Giarratano, Joseph C. & Riley, Gary D. (2005). *Expert Systems: Principles and Programming*. Book, Brooks/Cole Publishing Co.

Goleniewski, L. and Jarrett, K. W. (2006). *Telecommunications Essentials, Second Edition: The Complete Global Source.* Addison Wesley Professional.

Hebrawi, B. (1995). *GDMO, Object modelling and definition for network management*. Technology appraisals

ISO/IEC DIS 10165-4 / ITU-T. (1993). *Recommendation X.722, Information Technology - Part 4: Guidelines for the Definition of Managed Objects (GDMO)*, International Organization for Standardization and International Electrotechnical Committee.

ISO/IEC and ITU-T. (1998). *Information Processing Systems - Open Systems Interconnection - Systems Management Overview*. Standard 10040-2, Recommendation X.701.

ITU-T. (1992). *Recommendation X.700, Management Framework for Open Systems Interconnection (OSI)*. CCITT Applications.

ITU-T (1996). *Rec. M.3010, Principles for a Telecommunications Management Network (TMN)*. Study Group IV.

Maggiora, Paul L. Della., Elliott, Christopher E., Pavone, Robert L., Phelps, Kent J., Thompson, James M. (2000). *Performance and Fault Management*. Cisco Press

Morris, Stephen B. (2003). *Network Management, MIBs and MPLS: Principles, Design and Implementation by Publisher*: Addison Wesley.

Negnevitsky, Michael. (2002). *Artificial intelligence: a guide to intelligent systems*. New York: Addison Wesley.