Integrating Object-oriented O:XML Semantics into Autonomic Decentralised Functionalities

Frank Chiang, Hugo Fernandez, Robin Braun, Johnson Agbinya
Faculty of Engineering, University of Technology Sydney, Broadway, NSW 2007, Australia
Email: frankj@eng.uts.edu.au

Abstract

Autonomic Communications have attracted huge attention recently for telecommunication network management in the European Network Research Community. The aim of this research is to propose an Object- oriented hybrid O:MIB structure to replace current MIB architecture and establish a way of enabling O:XML technology into autonomic communications more efficiently with such abilities as autonomy, scalability, adaptation as well as simplicity for autonomic management application in complex networks. This paper presents an experiment that evaluates the performances of O:MIBs in terms of the loss rate, link utilization, management polling with regards to the congestion avoidance process. Simulation results show that O:MIBs provide compatibility with existing MIBs and proved to be an efficient extension over traditional SNMP MIB approach in a view that the workload of manager is greatly reduced. We conclude this end- to- end approach maintains self- managing capability, easy- to- implement scalability, involving heterogeneous smart algorithms for variant tasks.

1 Introduction

Next generation communication networks are evolving from conventional circuit-switched structures to the IP-based packet-switching network. It is envisaged that the high-bandwidth IP Internet backbone network will be seamlessly integrated with most mobile networks, such as wireless networks, Mobile ad-hoc networks (MANET), Wireless Ad-hoc Sensor Networks (WASNs), with a variety of access technologies, such as UMTS, GSM, WiMAX, CDMA2000, etc. Multimedia services in relation to video, voice and broadband data transportation are being frequently accessed via portable handsets. Hence, these broadband Wireless technologies, P2P communications, Pervasive communications, MANET, WASNs, WiMAX, Wi-Fi, Autonomic communications conceptions and portable communication devices are changing the focus of current network research from centralized control to distributed and parallel scenarios.

Large-scale, unpredictability, uncertainty, complexity and dynamics in both changing environments and variety of user demands are the main characteristics of these distributed networks currently concerning network community and OSS designers. These main characteristics present the challenges of efficiently handling network complexity, adaptability, autonomy, interoperability and service mobility.

While these distributed networks presents huge complexity which leads to be uncertain and unpredictable behaviors. The vision of Autonomic Communication Networks (ACNs) aims at assigning intelligence into local components according to Service Level Agreements (SLAs) and high-level business objectives. ACNs are not a brand new network, it considers all IP-based distributed networks described above. This vision is expected to improve the survivability of services and applications for the future next generation communication networks. Whereas, the proposed Self- X functionalities (e.g., self-optimization, self-protection, self-configuration and self-healing) for ACNs are essential requiring the capabilities of (a) collecting associated information, (b) modifying the network nodes' own attributes and (c) managing its own functions without network operators immediate care, which is termed as network autonomy. To achieve sufficient autonomy, the ACNs nodes should have sufficient privileges to access/read/write and modify their own variables and core files. System adaptation capabilities therefore increase with the growing of these privileges. But the system also takes a risk of unauthorized modifications and attacks, and consequently the system becomes much more vulnerable as well. Therefore, it is not good to always maintain high adaptation capabilities for autonomic systems. This becomes an unavoidable problem in current network infrastructure. However it is not in the scope of this paper, we have done research on the relationship between adaptability and vulnerability, they are not in the scope of this paper. The algorithm and protocol are also designed and implemented in order to seeking an optimal

1-4244-0977-2/07/$25.00 © 2007 IEEE

solution of efficiently reducing system vulnerability while maintaining self-adaptability. They are not in the scope of this paper too.

Our research motivation for this paper comes from such thinking: can we redesign the structure of network fundamental Management Information Base (MIB) such that the new structure is able to efficiently work with the future distributed complex networks and seek a specific software technique or programming language enables embedding smart algorithms and communication with agent framework such that to avoid the dilemma between autonomy, adaptability, robustness and survivability?

We propose an object-oriented O:XML-based approach applied to distributive essences of autonomic communication networks to enable autonomic functionalities. An object-oriented Management Information Base (O:MIB) structure is proposed as a theoretical foundation for our following work in the hope of replacing current Management Information Base (MIB) in current industries. Moreover, O:XML as a practical technology to implement O:MIB is explored and implemented with platform-independent java agents. A simple simulation is carried out as a validation example of synchronization process during program runtime.We demonstrate the desirable autonomic functionality/features can be separated from applications adhere to network components and assigned/fulfilled by specific configuration settings of autonomic individuals, and thereby reducing the cost (including TCO and operational costs) and the overhead of maintaining autonomy as well as quality of adaptation [1].

This paper is structured as follows: a brief introduction and motivation of applying O:MIB is stated in section 2. In section 3, the relations between network adaptability, autonomy in complex networks and new O:MIB structure are analyzed. The O:XML is described in section 3 and agent semantics and its interaction with O:XML. Network deployment and simulation are reported in section 4. Finally, section 5 concludes and provides a description of future challenges.

2 Object-Oriented MIB – O:MIB

2. 1 O:MIB

Inspired from Active Network approach [2], we propose an object-oriented Management Information Base termed as O:MIB in this paper. MIBs usually have 3 branches: iso, user profiles, and vendor profiles. The unique Object Identifiers (OID) is applied to represent each managed object, or MIB object. The whole MIB is constructed by many managed objects which form a hierarchical tree structure for managed information represented. The top level of the tree structure is controlled by ISO and ITU organization for any additions to the MIB trees for new managed objects, interfaces, or devices. Figure 1 is an example of a MIB tree structure and naming scheme, and the dotted line in figure 1 representing those subentries from O:MIB that can be added on afterwards.

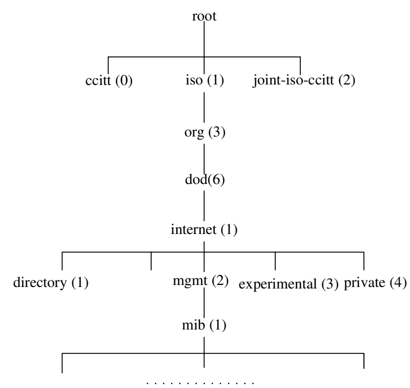

Figure 1. The Functional Domain of O:MIB in MIB Tree Structure

On the basis of widely-adopted RFC1213 specification for MIB-II [3], we believe it is possible and efficient to add up more sub entries in current MIB structure, for instance, these subentries consisting of more attributes and methods can be added on below (object identifier) OID = 1.3.6.1.2.1 (i.e., iso. org. dod. internet. mgmt. mib). On some circumstances, more subentries could be embedded into private enterprise network OID = 1.3.6.1.4.1 (i.e., iso. org. dod. internet. private. enterprise). We hope this O:MIB can partially replace the NGOSS Shared Information/Data (SID) model [4] to some extent in that the information synchronization process is not necessarily periodically invoked as current database-derived scenarios, rather, only act on the on-demand requests (ODREQ) which trigger the get/set/update/trace information processes together with customized methods in O:MIB (e.g., getUpdateFrequency() or setAccessPermission()) to get the information , update information contents and other more actions defined in methods. The on-demand approach is more efficient in distributed networks with less messaging overhead where resources (e.g., bandwidth, battery power, capacity) are limited.

Hence, current communication networks are struggling to survive due to the fact they are fragile and rigid in nature. They appear to be difficult to operate, configure and repair. Traditional SNMP-MIB technology cannot meet the network requirements due to the fact that contemporary distributed multi-services/multi-technologies environments provide new challenges of interoperability

Conventional MIB	Object-oriented MIB
Hierarchical Structure	Hierarchical Structure
Information Elements Stored	Information Elements + functions, Algorithms + Embedded Agent Semantics
Data Oriented	Object-oriented/On-demands
SMI/ASN.1- Standardized	O:XML-Enabled
Static/Fixed	Dynamic/Extensible/Reconfigurable

Figure 2. Comparison Between Conventional SNMP MIB and O:MIB

and flexibility in both data transportation networks and service/application management processes such as Service Fulfillment (SF), Service Configuration (SC) to Service Provider (e.g., ILECs, CLECs, IXC, CAPs, ISPs and ASPs) in delivering advanced IP networks. While current MIB structure is fundementally describing network data structures with QoS information, Devices information, etc . These cannot fi t into increasingly distributed services and application networks any more. Therefore, our question is whether we can build up a new network management base structure with good compatibility with legacy MIB technology but much more effi cient and help enable desirable self- organization patterns to improve its survivability for the complexity in future telecommunication networks.

In addition, the pervasive issues in Telecoms Network Management (TNM) more focuses on decentralization and cooperation in contrast to traditional centralized or highly coordinated management paradigms. When the network develops quickly in size and complexity, the centralized management doesn't suffi ce. Distributed agent-based systems as an active fi eld of Distributed Artifi cial Intelligence (DAI) in last two decades has maturely developed so far. Therefore, Autonomous Decentralized Systems (ADSs) are the best solutions for large-scale distributed telco systems so far. Therefore, an Object-Oriented Management Information Base (O:MIB) is needed to fulfi ll the service management (e.g., service discover, service confi guration, service deployment, etc.), application activation process, resource allocation process, etc. O:MIB plays a role as distributed information model to enable autonomous agent behavioral model which is being explored with the EML model of Tele-Management Force (TMF) [1] in our previous research framework [5].

Seeking an optimal solution to achieve enormous service tasks with limited network resources is crucial in current network operational system. The latest XML technology - Object-oriented O:XML enables the implementation of the O:MIB becomes possible. The new challenge facing to-

[1] http://www.tmforum.org/

<table>
<tr><th colspan="2">Attributes</th><th>Methods</th><th>Embedded Algorithms</th></tr>
<tr><td>QoS Parameters</td><td>Bandwidth (GB)
CPU utility (%)
Delay (s)
Link capacity (MB)
Traffic conditions</td><td rowspan="4">getVariableValue( )
setVariableValue( )
recordUpdatedFrequency( )
getUpdatedFrequency( )
....
getDependency( )
synAssociatedMIBVar( )</td><td rowspan="4">(1) For security:
Probability for vulnerability

(2) For Wireless Sensor/ Ad-hoc network:

neighbourhood locating or discovery;
Directing;
Likelihood Calculating,
Throughput increase or Packet Drop Ratio, etc.</td></tr>
<tr><td>Devices Parameters</td><td>ManufactureID, ModelID, FirmwareID
Detailed Device Parameters: TX Range, Temperature, etc</td></tr>
<tr><td>Services Parameters</td><td>SrvID ; SrvSemantic
Availability_Status</td></tr>
<tr><td>Application Parameters</td><td>AppsID, AppFunctions
Availability</td></tr>
</table>

Figure 3. Methods, Algorithms in O:MIB

day is that the network outgrows the capability of existing OSSs' propositions. For instance, on one hand, the network has more dynamics and self-confi guring mechanism such as routing, bridging, dhcp, and on the other hand, the users requests cannot be fulfi lled in real time from the business point of view. The distributed O:MIB with embedded Autonomic Agents (AAs) technology enables the self-organized management (e.g., resource management, service managment, etc). The basic information unit of the O:MIB is the element or object. Each element includes:

- Attributes: specifi c data values that represent characteristics of the managed OIDs
- Method Behaviors: actions that help to achieve autonomic communications including the self-awareness function on time, intensity and spacial information
- Algorithms: algorithms that fulfi ll a particular network task can be embedded into O:MIB domains; it is a set of predefi ned use of aggregation of available method calls, such as monitoring temperature, humidity in wireless sensor networks, calculating vulnerability level or raising alarm levels in ACNs networks
- Messaging: messages that can be invoked by the local messaging daemon action as a response of ondemand requests in order to get general information (e.g., topology, geometrical neighbors, etc)

Figure 3 lists the necessary contents of an O:MIB class, which should cover four divisions of information - QoS parameters, device information, service information, application information and the dependency information may associated with services and devices. This O:MIB class is designed for each category of network components. Whenever a certain object is in a need of activation during run time, the java code that creates any instance of OMIB_Class is directed via the keyword "new": OMIB_Class OMIB_object = new OMIB_Class();

3 Use of O:XML

XML is a widely-used general purpose markup language used as an efficiently way for data structural recording, messaging and document storage and processing, both online and offline [6] . Its hierarchical structure is suitable for most types of documents in hierarchical networks. And XML is easily to be parsed by other parser algorithms because of its strict syntax and parsing requirements. XML is able to describe current MIB structures. Java agents can parse the document and apply write/read action onto XML easily, however, it is not possible or eligible in terms of its current syntax to have methods or functions into its structure. The emergence of O:XML makes it possible to record O:MIB.

O:XML is an object-oriented language, its syntax is straight-forward and on the basis of conventional XML structural scripts. Other than traditional usage of XML for the purpose of only data structure description, O:XML has more features including function overloading, polymorphism, exception handling, threads, etc[2]. These features make O:XML available for describing the variables and functions in O:MIB as described above. The new release of O:XML is integrated with Spring application framework, which enables us to embedded the java code (termed as javabean) into o:xml, while the java beans could be containing algorithms which can function as finding hidden information such as probability of neighborhood availability; or self-reasoning about procedures; or even reconfiguring the data value/services of the component itself.

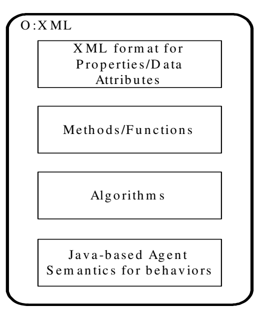

Figure 4. Hybrid Structure of O:XML and XML

ObjectBox complier/interpreter is needed to interpreter methods or algorithms within each O:XML file. And XML is easily to be parsed by java agents and the xml contents could be simply copied to the O:XML output of ObjectBox compiler, therefore, XML file is also being recognized by O:XML interpreter. This indicates the possibility that mixing XML format with O:XML formats. In order to fully make use of merits in both language, we combine their advantages and produce a markup structure with hybrid XML format and O:XML format, where data structures (e.g., node attributes) are described in XML format, functions and embedded algorithms are stated in the O:XML format. Figure 4 shows the possible structure of mixing XML format with other O:XML format. XML semantics are used to describe properties of MIB objects. O:XML syntax are used to describe the methods, algorithms, and the java extension of O:XML provides feasibility to embed Java agents inside. We take the MIB objects in ifTable (OID = 1.3.6.1.2.1.2.2) as an example, their attributes are depicted via XML semantics. As a hybrid structure mentioned above, O:XML format is hybridized with XML, and is used for describing methods and algorithms.

4 Use of Autonomic Agent

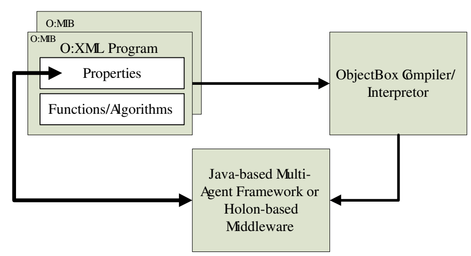

Figure 5. Integration of Multi-Agent Framework with O:XML-Implemented O:MIB

Figure 5 shows the process of software agents (or holon in holon-based middleware) calling functions or algorithms in O:MIB. Data properties are reconfigured by java-based agents. ObjectBox, as the interpretator of O:XML, could expose the information in O:MIB to the agent system. Figure 6 demonstrates the sample scripts how the java agents calling the methods/algorithms in O:XML files.

5 Experimental Validation

In this section, we present the performance analysis in order to evaluate the performance of the local operation and efficiency of our proposed scheme by applying O:MIB structure. An ad-hoc network topology that consists of six source-sink nodes is designed and implemented in J-Sim. J-Sim is a component-based java simulators, which is widely

[2] http://www.o-xml.org/

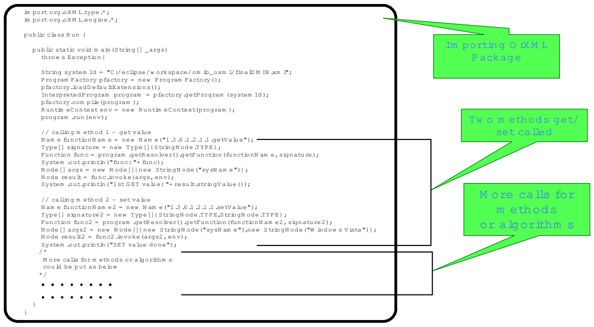

Figure 6. Scripts Sample of Callig Functions and Algorithms via Software Agents

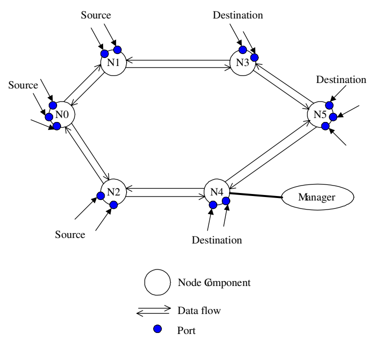

Figure 7. Simulation Topology

used to simulate the network traffic, INET protocols, wireless networks, sensor networks. The autonomous component in J-sim programming allows the components to be added in, removed or modified during the run-time execution. Node mobility currently is not considered in this topological model in terms of the simplicity of the model. Figure 7 shows the topology experimented in this paper. Node_0, Node_1, Node_2 have two or three traffic source. Node_3, Node_4, Node_5 are destinations for packet delivery. The traffic is a combination of constant traffic flows plus an exponential on-offs traffic peaks. The congestion happens whenever these exponential on-offs occur beyond the threshold we defined in the simulation.

We simulate a congestion avoidance algorithm deployed onto both SNMP MIB-based scheme and distributed O:MIB-based scheme. The key simulation parameters are summarized in Table 1. By studying the variables in MIB, we know that about 90% of the MIB variables currently are static information from vendors, only about 10% of them changible and flexible are for the purpose of O:MIB scheme. Therefore, practically only these 10% variables are to be instantiated into objects stored into the O:MIB, by so doing, further reduce the memory size and CPU load in terms of hardware implementations.

Parameters	Values
Network Topology	6 souce-sink Nodes + 1 Manager
Link_Capacity	1.5Mbps
Propagation_Delay	300ms
Queuing algorithm	Tail Drop
Traffic pattern	Constant flows + Exponential on-offs

Table 1. Key parameters for the experiment

As a part of the managed objects, the O:MIB variables associated with conventional MIB are listed in Table 2. Those MIB-II objects are stored in each of the node component of the topological figure 7. These O:MIB variables are polled and evaluated based on two different schemes as described above. One approach is based on the conventional SNMP manager, with the manager collecting information and activate the process of sending messages to SNMP agents who is subsequently operate MIB variables via existing commands. The other approach is based on proposed distributed agent scheme via O:MIB, which means when the manager receives the notification of signs of congestion on any node, the congestion avoidance tasks will be delegated to agent residing on node components to carry out the congestion control which, however, is done by the manager in SNMP MIB scheme.

The objects in Table 2 are under the ifTable (OID = 1.3.6.1.2.1.2), and are related with congestion control tasks, for instance, the link utilization is calculated on the basis of the information provided by the MIB object `ifOutOctets`. The object `ifOutDiscards` define the number of outgoing packets being discarded via the interface. In SNMP MIB approach, the manager is responsible for the execution of the management task at any time, such as defining maximum congestion time. While in our proposed O:MIB scheme, the manager delegates most of the control tasks to local agents, who carry out the tasks to release the traffic congestions. Two customized methods (e.g., `getValue()` and `setValue()`) are defined as global functions as described in the table.

Simulation results are shown in Figure 8. Figure 8(a) and 8(b) presents the packets condition in sequential event (e.g., the same exponential burst is applied to all source node) in a consective time period, while Figure 8(c) and 8(d) presents the packets condition in parallel events (e.g., more than one different exponential traffic bursts reach source nodes such as node_0, node_1 and node_2) during the same time period. Figure 8(a) and figure 8(c) shows the discarded packet rate on different nodes. The proposed scheme exhibits a better performance than the conventional MIB management

Methods	OID	Descriptions
`ifNumber()`	`1.3.6.1.2.1.2.1`	number of interfaces
`ifSpeed()`	`1.3.6.1.2.1.2.2.1.5`	bandwidth of interface
`ifOutOctets()`	`1.3.6.1.2.1.2.2.1.16`	number of octets out of interface
`ifOutDiscards()`	`1.3.6.1.2.1.2.2.1.19`	number of outbound octets to be discarded
`ifOutQLen()`	`1.3.6.1.2.1.2.2.1.21`	queue length
`getValue()`	`1.3.6.1.2.1.global`	get current value
`setValue()`	`1.3.6.1.2.1.global`	reconfig the attribute values

Table 2. Methods used in this simulation

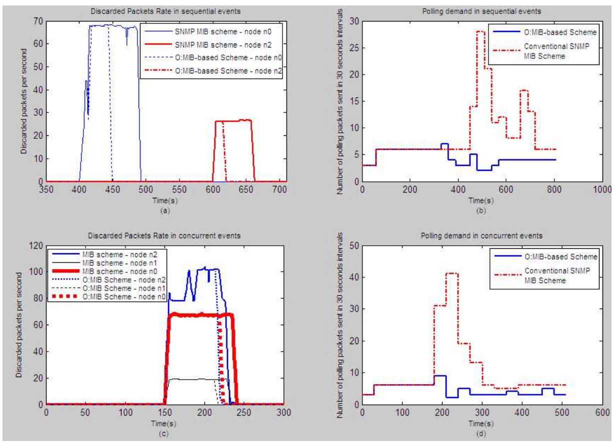

Figure 8. Simulation Results

scheme. The time duration of the discarded packet action in terms of both parallel and sequential events' scenarios are both minimized whenever the proposed scheme is applied in both figures. The reduced ratio ranges from 50% $(= \frac{50s}{100s})$ to 75% $(= \frac{75s}{100s})$ as depicted in Figure 8. The manager workload is compared in Figure 8(b) and figure 8(d). A periodic 30 second polling and the manager's task reassignment is simulated. A significant reduction of workload is achieved by adopting the proposed O:MIB scheme. Autonomous agents reside on node components.

6 Conclusion and Future Work

We have presented an efficient O:MIB- based approach implemented with a practical object-oriented hybrid O:XML technique, in order to cope with management autonomy in terms of self-configuring, self-optimizing, self-adapting, self-limiting and self-preserving. The simulation results on traffic control show that proposed O:MIB structure and embedded smart algorithms in O:XML format outperform the current SNMP MIB-based management structure. This new scheme is easily to be applied into other self-X functions in ACNs. The data attributes, methods as well as algorithms can be implemented in one O:XML file for each MIB object. This creates its possibility of being embedded into portable devices in the future for personal consumer networks and pervasive computing paradigm. The proposed scheme seems to be a first attempt in rethinking MIB structure in a new way and therefore avoid the problem which is encountered and experienced by traditional network management paradigm where is mainly building a "stovepipe" technology and finally results in stovepipe standards and protocols. The core technology and scheme proposed in this paper by applying O:XML technology into local O:MIB can be applied into many networks in the future, such as Wireless networks including MANET, WASN, P2P networks, Mesh Networks, and next generation networks.

References

[1] E. Gjorven, F. Eliassen, and J. Aagedal, "Quality of adaptation," in International Conference on Autonomic and Autonomous Systems, p. 9, 2006.

[2] S. F. Bush and A Kulkarni, Active Networks and Active Network Management: A Proactive Management Framework. New York, Boston, Dordrecht, London, Moscow: Kluwer Academic/Plenum Publishers, 2001.

[3] RFC1213-MIB, "Rfc 1213," tech. rep., http://www.simpleweb.org/ietf/mibs/modules/html/?category=IETFmodule=RFC1213-MIB.

[4] "Ngoss shared information/data model overview," 2004.

[5] F. Chiang, R. Braun, and J. Hughes, "A biologically inspired multi-agent architecture for autonomic service management," Journal of Pervasive Computing and Communications, vol. 2, no. 3, pp. 261–275, 2006.

[6] J. Bih, "Deploy xml-based network management approach," Potentials, IEEE, vol. 24, no. 4, pp. 26–31, 2005.