マルチメディア通信と分散処理　57－8
(1992. 9. 24)

知的ネットワーク管理へのMIKBアプローチ

G.MANSFIELD*, K.JAYANTHI*, 村田眞人*, 樋口謙一*,
根元義章†, 野口正一†

*高度通信システム研究所,
†東北大学

〒989-32 仙台市青葉区南吉成6-6-3 ICRビル

あらまし　本稿では，人手を煩わせることのない知的ネットワーク管理の目標について概説する．
このようなシステムにとって重要かつ必要なコンポーネントは，管理者の推論や判断をモデル化した知識ベースである．このような知識ベースを作成する場合の自然なアプローチは，Management Information Base(MIB) のオブジェクトから始めることである．なぜなら管理者はこれらの値からネットワークの状態を知るからである．ネットワーク管理者の主要な視点は，値，値の増減，増減の速度，長期間あるいは短期間での値の変化，障害の重要度のような様々な定量的なパラメータへとモデル化される．
我々が提案する知識ベースは，MIBに関連し個々のネットワークとは独立したManagement Information Knowledge Base(MIKB) と，その他の，ネットワークに依存したNetwork Knowledge Base(NKB) からなる．

和文キーワード　知識ベース，ネットワーク，MIB，管理オブジェクト，重要度

The MIKB Approach to Intelligent Network Management

G.MANSFIELD*, K.JAYANTHI*, M.MURATA*, K.HIGUCHI*
Y.NEMOTO†, S.NOGUCHI†

* AIC Systems Laboratories
† Tohoku University

AIC Systems Laboratories, 6-6-3 Minami Yoshinari,
Aoba-ku, Sendai 989-32

Tohoku University,Sendai.

Abstract　The target of Intelligent Network Management, as a hands off management system, is outlined. An important and necessary component of such a system is the knowledge base that models the reasoning/judgment of the human operator. A natural approach in the context is to start from the objects in the Management Information Base(MIB), as the human manager looks at the network in terms of these. The subjective view of the network manager is modelled into various quantitative parameters, such as value, velocity, acceleration, long and short term values, abnormality weights and fuzzy handling. The MIB related component of the KB - the Management Information Knowledge Base (MIKB), is network independent. The other component of the KB- the Network Knowledge Base (NKB), comprises the network dependent part.

英文 key words　knowledge base, network, MIB, managed objects, weight

1 Introduction

Any management basically involves gathering information about the subject that needs to be managed. The gathered information is then processed and analyzed using the knowledge about the subject, to make judgements about the status of the subject and to take decisions regarding the future course of action.

The work done in the area of network management is basically in two categories. The work done in the first category [1,2,3,4] has focussed on the information gathering aspect. The network management systems [1] basically gather information about the network status and display them in user friendly/attractive formats. Little or no *management* is carried out *by* these management systems. The omnipresence and omniscience of the human network manager is implicitly assumed. He/she knows what information to look for and also knows what to make out of the information. The *management system* is simply a tool that aids the manager in obtaining the information. But this is contrary to the growing requirements in the area, where networks need to be running round the clock, are being maintained on a temporary and voluntary basis by researchers/students who have other primary occupations. The issue of professional management is getting critical in the field. Moreover, knowledgeable network managers tend to use more involved, 'expert-friendly' and direct means to obtain network information, than offered by the network management systems. Without any *value added* to the raw information, there is no incentive for most managers to use such systems.

Work in the second category [5,6,7,8] has concentrated on the development of expert systems for network management. These systems have generally approached network management from the problem point of view. Working from the problem, the solution is sought and the necessary interaction with the network, to detect, diagnose and solve the problem, is carried out generally using proprietary/non-standard protocols, data-structures and pieces of information. Naturally systems following this second approach suffer from the drawback of lack of interoperability, and are generally too specific and do not cover the wide ranging daily chores of the human network manager.

While in the first approach the focus is on making information available, the second approach focuses on using information for management purposes. It does seem logical that the second approach would build on the first, i.e., the management would be carried out based on the information available (extending the base if more information was necessary). But unfortunately this has not been the case; the two approaches have developed independently.

In this work we take the logical approach of *adding value* to the raw information available from the system, in order to build an intelligent network management system, which takes care of at least the routine observation and decision making activities. Naturally, an important and necessary component of such a system as this is the knowledge base that models the reasoning/judgment of the human operator. The *Configuration Database* [9] contains information about how a network or network element *should* be behaving. It is seen as a logical extension of the carefully annotated network management maps that adorn the walls of Network Operation Centers. The design and contents of *Configuration Database* has been identified as a key issue that needs to be addressed. Our approach, the MIB approach, shows a natural way of developing the core component of the knowledge base.

In section 2, we develop a model for intelligent network management. In section 3 we examine some operational aspects of network management, followed by a brief outline of an implementation of the system in section 4. Finally, in section 5, we summarize the current status and give our course for the future.

[1] For a survey of present network management systems refer (1)

2 A Model for Intelligent Network Management

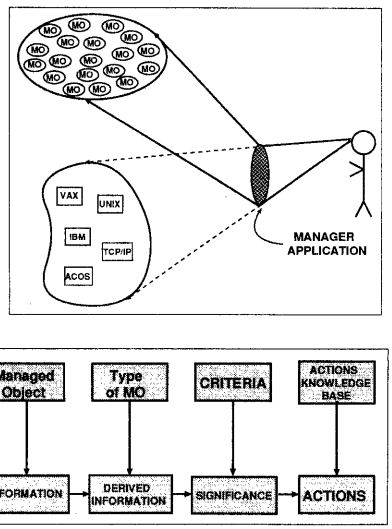

Fig. 1 Network Manager's View of the Network and The Management Model

A lot of the work in Network Management, so far, has been done in building the framework. Presently, there is an agreed management framework. Agents residing in network elements will provide the network manager application with information about the network. Standards have been fixed for representing [15,16] and communicating [17,18,19,20] management information between the manager and the agent. The objects which need to be managed, are listed up and described in terms of their properties in MIB's. In this framework, the human managers' view of the network is limited to objects in the Management Information Base(MIB). Of course extensive research has been carried out to develop these MIBs and the development and refinement is an ongoing process. It seems natural that MIB provides the core component in the KB design and a viable starting point for knowledge acquisition. The significance of various managed objects(MO) as seen by a network manager needs to be acquired and represented in

the KB.

The wildcard approach to building the knowledge base wherein the network manager is interviewed and asked to share their knowledge is tedious, inefficient and not always productive. Such approaches of building knowledge bases have been pursued by most Expert Network Management System researchers and designers [10,11]. Once input is obtained from the network expert, mapping it to the network management framework of MIBs is another complex procedure that has led to the trend of developing proprietary systems.

In the MIB approach to building the knowledge base we asked the network managers what the MO's present in the MIB signified to them. To further structure the interview we studied some of the widely used MIBs[25] to arrive at the nature of the MOs and hypothesised a model of the network management activity. In the model *the MO gives some information, from which the Manager derives some information and then applies some criteria to decide the significance of the derived information (good/bad ...). Based on the perceived significance the manager then proceeds to decide the necessary action (no-action, some action, emergency ..)* (Fig.1).

2.1 Types of Managed Objects

An examination of the MIBs for TCP/IP based internets shows that MOs are basically of the types shown in Fig.2. Appendix A gives the break-up of the MO's for each type.

Status objects	These take values from an enumerated set
Counter objects	Monotonically increasing non-negative numbers
Gauge objects	Non-negative numbers which may increase or decrease but latch at a maximum value
TimeTicks	A measure of the time in hundredths of seconds since some epoch
Descriptive	Descriptive information - names, addresses, routes,...
Sequences	Combinations of some or all of above
Tables	Array of sequences

Fig. 2 MO types in MIB

The information content of various types of objects from the management point view is given below:

- Status : generally gives a direct indication of the state of a system; for example whether the status of an interface is 'UP', 'DOWN' or 'TESTING'
- Counters, Gauges : generally a measure of some quantity (e.g. input packets, output errors, Q-lengths, ..) that gives an indication of the performance and/or status of the system. There are generally allowable/normal values and when the values go beyond these thresholds, the situation calls for special investigation/action.For example, continued and rapid increase of traffic is cause for alarm.
- Descriptive objects, sequences, tables : Changes in these objects generally indicate some change in the network and may merit an investigation. For example a change in the routing table is indicative of the addition/deletion of some link somewhere. A deletion of a link is in turn indicative of some faulty gateway/interface.
- TimeTicks : measures the time since an event has occurred.

2.2 Derived information: Velocity, Acceleration & History

The current value of a MO is a valuable piece of information. It tells a knowledgeable person more than just the figure, but only when read in the context of the history of the object. The frequency at which the object representing the operational status of an interface is changing gives the operator an insight into the reliability/error proneness of the interface. The same holds for counter and gauge type objects. The information that some counter has a value x does not signify much when judged in isolation. The significance is seen by the manager by using the history of the object to estimate the time development of the MO. For example, it is crucial to know whether an error count is constantly increasing, rapidly increasing, or only transient. For a more quantitative analysis, it was clear that the *velocities* and the *accelerations* would be effective measures to capture time-development related derived information of Counter and Gauge type MOs.

The history of MOs are described in terms of Means, Medians, Modes, deviations, ... on a periodic basis (hourly/daily/weekly ..) and are retained in the Network Information Base (NIB), described in detail in section 3.6. These are indications of the patterns, if any, that the MOs are having in space and time and provide valuable indicators in determining the significance (normal/abnormal) of the current states of MOs.

2.3 Criteria for evaluating status

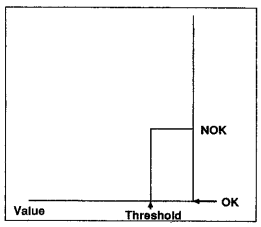

Fig. 3a The Step function

In the proposed model the network manager judges the status of the network by judging the information available (information, derived information, history) on objects against a status evaluation criteria which represents the managers' knowledge concerning the MO. This knowledge, MIKB, can be represented by the set $\{ C_i \}$ where C_i is the criterion to judge whether an MO indicates any abnormality in the network.

$$\text{MIKB} = \{ C_i \}$$

The abnormality a_i due to an object i is given by the criterion represented as a function as follows:

$$a_i = C_i(x_i, v_i, a_i, t_i)$$

where x_i is the value of the MIB variable, v_i is the velocity, a_i is the acceleration and t_i is the time interval in which the variable x_i is studied. The function C_i is a step function (Fig.3a) which assumes the high value if x_i, v_i, or

a_i exceeds their respective thresholds (for objects of type counter/gauge).

This was our basic proposal; but on our first interaction with the network manager we came across such statements like: *in the short term two packets lost per second is bad, and in the long term 60 packets lost per minute is bad.* This led us to refine our abnormality model to:

$$a_i = C_i(x_i, vs_i, vl_i, as_i, al_i, ts_i, tl_i)$$

where, vs_i, as_i, vl_i, al_i denote the velocity and acceleration in the short term and long term respectively, and, ts_i, tl_i define the time intervals for short term and long term respectively. The step function C_i assumes the high value if x_i, vs_i, vl_i, as_i or al_i exceed their respective thresholds and is low otherwise.

The composite abnormality of the system is then given by the *composition* of all the abnormalities:

$A = a_1 \oplus a_2 \oplus ... \oplus a_n$, where n parameters are being observed.

This leads us to a very simple means of quantitatively analysing the network status by ascribing thresholds to the counter, gauge type MO's and by ascribing status to the state variables.

2.4 Degree of abnormality

The knowledge represented in the model so far indicates whether the status of an MO is abnormal or not. There are several possible abnormalities in a network and not all of them are of the same degree of seriousness. An increase in the number of error packets would not warrant sending an alarm to the network manager to wake up that person in the middle of the night. However, the change of the operational status of an important network interface from "up" to "down", may warrant immediate alarm. Thus arose the necessity of assigning weights to each of the abnormal conditions. The weight represented the degree of seriousness of the abnormality of an MO. The weights themselves were functions of time. For example, if disk space is approaching criticality during a working day, since assistance is available at short notice, we can afford to wait longer. Whereas if it is weekend, then appropriate time must be given to allow the operator to reach and rectify the situation, which means that we cannot permit the critical situation to be actually reached.

2.5 Fuzziness of decision-making

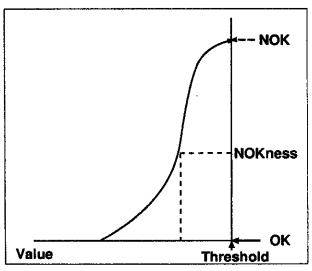

Fig. 3b The Fuzzy function

The other problem we faced with the abnormality was in evaluating threshold conditions. In the discrete mode of evaluation, if x_i or v_i is $\geq th_i + \alpha$, the sitution was deemed abnormal, while if v_i is ¡ th_i - α, the situation was deemed normal for however small α. This decision did not reflect the actual situation, and it did not find favour with the network managers, who preferred a more subjective decision like, if x_i is equal to th_i - α the situation is not "very" good (Fig.3b).

Thus it was necessary to employ a fuzzy algorithm to compute the membership function that measured the $NOKness$ of an MO, contrary to the step function C_i in section 2.3. The total weighted abnormality WA of the system is then given by

$$WA = w_1 * a_1 \oplus w_2 * a_2 \oplus ... \oplus w_n * a_n$$

where w_i is the weightage of the abnormality corresponding to the ith MO. Of course for the status type objects the traditional two/multi-valued logic is applied.

2.6 Inter-dependency of MO's

In the model developed so far, MOs have been treated individually. However the inter-dependency of the MOs does appear several times in the criteria for deciding the abnormality of an object. For example, the time threshold beyond which if the OutQlen persists (continues to be non zero) the condition will be labelled abnormal, is dependent on the speed of the interface as shown in the example (Fig.4). Also, the Operational Status of an interface being down will be considered abnormal only if the administrative status of the interface is UP.

```
MO: Output Q length
      if    interface speed = 10Mbps (Ether)
      then  time_threshold for (OutputQlen > 0) = 1s
      if    interface speed = 1Mbps (ThinEther)
      then  time_threshold for (OutputQlen > 0) = 10s
      if    ifSpeed = 64Kbps
      then  time_threshold for (OutputQlen > 0) = 1min

MO: interface operational status
      if interface AdminStatus = Up
        if  interface OperStatus = Down
        then Status is abnormal
```

Fig. 4 Inter-dependency of MO's

2.7 Network Knowledge Base

The focus of the network management knowledge base so far has been the MIB, i.e. we have been concentrating on that part of the knowledge which is not specific to a network. However, in actual management several judgments, decisions and actions are dependent on the local environment and configuration. Thus it may so happen that the MIKB is indicating the likelihood of an abnormality but the NKB on the basis of some local knowledge overrides it. Also, the importance of an event may be further emphasized in the NKB. For example, all interfaces are important and should be operationally UP but some interfaces may be more important than the others. Actions are generally logging messages, sending e-mail, fax messages and in the most critical cases ringing pocket-bells, etc. These of course very much depend on the local circumstances and are coded in the NKB, along with information about alter-

nate routes, special requirements etc. of the network. The NKB is the interface through which the local administrator can tailor the management system to suit their specific requirements.

The KB is formed by the union of the rules in the MIKB and the NKB, with the rules in the NKB taking precedence.

$$NMKB = \{ MIKB \} \cup \{ NKB \}$$

3 Operational issues and features of the system

In this section we examine some of the operational issues of network management systems in general and explain the features built into the proposed system[8] to tackle these issues. The system targeted the management of TCP/IP based networks. And, due to its simplicity, wide-spread acceptance and market position the Simple Network Management Protocol (SNMP) was adopted.

3.1 Time labels

In the network management system the manager polls the agents and obtains pieces of information about the network. From the pieces of information obtained from the agents, the manager reconstructs the overall picture of the network status, performance, problems, etc. An indispensable factor in this reconstruction is the time-labels of relevant information.

In our management framework, the agents, when they collect information about the network, affix a timestamp to the information. This allows the manager to see the time development of the MIB's of the agents. This is essential, also for offline diagnosis of problems. Time labels are also necessary for the manager to correlate various events to see cause and effect. Also, for these time-labels to make sense it is important that the time-labels refer to the same clock i.e. that the clocks of the manager and the agents should be synchronized.

3.2 Sampling time windows and transients

The image of the network, seen for management purposes, depends very much on the polling interval employed. As is seen in the case of traffic analysis on a certain network segment (Fig.5) narrow windows are essential in studying the load characteristics of a network especially when the traffic is bursty. The burst observed with a sampling interval of 10 minutes is missing in the graph taken at 20 minute intervals. Also, when observing the status of MOs, large time windows (polling intervals) are likely to even out all the transients and present a misleading view of the network.

Part of the log showing the operational status of an interface of a machine is shown in (Fig.5). It is clear that with polling intervals of 30 secs most of the events (status changing to down/up) would have been missed. However, it is important that we do not add to the load in the network by our management system by making the polling interval arbitrarily small.

3.3 Limit on queries from Manager to Agent

In essence SNMP is a polling protocol. Polling protocols impose a limit[21] on the number of agents that a manager can poll. This limit arises from the time taken in processing the relevant packets at the respective hosts and the time spent in traversing the intervening media. In a LAN environment a single query may take 200ms approx. While in a WAN environment a single management query may take as much as 1.5 secs. This leads to a limit on the number of network elements that can be managed (150 and 25 for LANs and WANs respectively, assuming a polling cycle of 30 secs.)

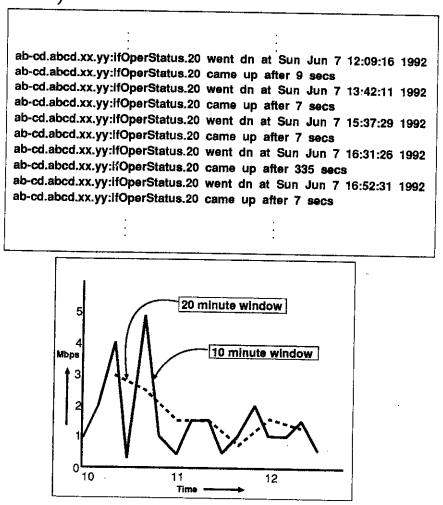

Fig. 5 Effect of different time windows

3.4 Network load due to management traffic

The number of management packets generated is roughly 2 X $(1/\Delta)$ X N , where Δ = polling interval and N = number of agents. As small time-windows are essential to obtain a reasonably accurate description of the network Δ is small and in a large network N is large. Consequently the management traffic is significantly large. The burden on the network and the communication software is heavy.

3.5 Composite time labelled objects

To introduce the concept of time management, to allow for time labels, small time windows, and to keep the management traffic low, it is necessary to introduce new Managed Objects in the MIB. In these Managed Objects additional time related attributes have been associated to the original object of interest. To monitor a particular object starting at a time T, n times, at intervals i , n Get *value* requests would be sent and answers received. Using the *Composite Time labelled Objects* the manager uses the SNMP set primitive to set the *StartTime*, *Interval* and *NoOfReadings* attributes of that object to T,n and i respectively. The intelligent time synchronized agents maintain the objects accordingly and on a Get *StringofValues* request from the

manager send the complete list of readings in one shot, thereby significantly reducing the traffic load generated by management systems.

3.6 Network Information Base

The history of the network system is maintained in the Network Information Base (NIB). In the NIB all events and short/medium/long term statistical charactersistics of the MOs are recorded. This allows the Network Manager application to make intelligent decisions, by looking at the current status of an MO in the perspective of the history of the MO and that of the network. The NIB is maintained in a DB form and is accessible for offline diagnosis and maintenance purposes too.

3.7 Intelligent polling

Polling intervals for MOs can be chosen and set by the local manager/operator. There is a default moderate polling interval for all objects. However the polling interval is a very critical parameter in the syetem. It has been shown in 3.2 that polling intervals need to be short to capture the correct picture of the network including the transients. It has also been shown in 3.4 that the management traffic increases directly with the polling frequency. To alleviate this impasse intelligent polling strategies have been employed wherever possible, based on the experts knowledge of the system. For example, in the case of the operational status of an interface if there is no change noticed using the moderate polling interval but the lastChange MO for the interface indicates that there has been a change then the polling frequency is increased. On the other hand, if the interface is found relatively steady for a long time, the polling frequency is lowered till it reaches the moderate value.

4 Prototype System Design and Operation

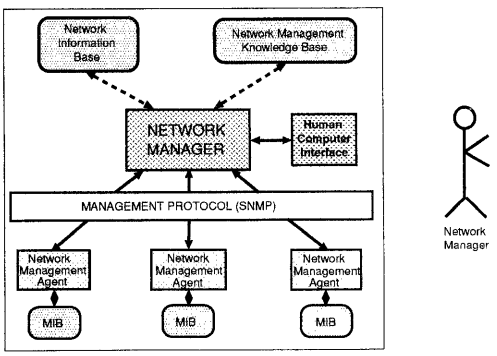

Fig. 6 Implementation of the system

The design of the management system is shown in Fig.6. The manager polls the agents for network information. On the agent's side, the *StartTime*, *Interval* etc., viz., the time-related attributes of the composite time-labelled objects are watched and corresponding network-related attributes are maintained accordingly.

The manager and the agent have synchronized clocks. The clock synchronization is carried out using the Internet time synchronization protocol NTP (Network Time Protocol). It is possible to analyze traffic in very small time windows (500 milliseconds). The status of the network as seen by the network manager application is contained in the Network Information Base(NIB). The management is carried out by using the information in this NIB, in conjunction with the knowledge in the *Network Management Knowledge Base*.

4.1 NMKB prototyping

The Knowledge obtained from the network manager is in the form of rules. These set of rules required several cycles of modification, refinement and enhancement with the cooperation of Network Experts before the rules could be used for a pilot system. At this stage the rules were described in a higher level language CESP (Common Extended Self-contained Prolog)[10] and the corresponding interpreter aided in the prototyping and in identifying the inadequacies and anomalies of the rulebase. Once the rule base was reasonably stable -the thresholds, values, weights and other definite details fixed, it was translated into a table which drove the pilot network management application written in C.

4.2 The Human Computer Interface

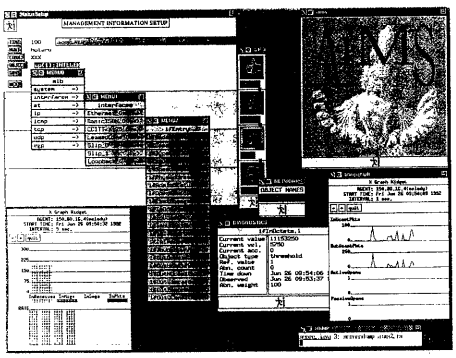

Fig. 7 The Human Computer Interface

The Manager application is provided with a point-and-select graphical user interface. Management functions like data and event reporting, configuration changes (setting MIB variables) and viewing the reported data and events are supported by the interface. The interface is based on the X window system and is portable to any system having an X platform. An interface with pocket bell / fax / telephone and similar communication equipment is implemented to ensure swift action in case of any mishap. The intelligence in the Human Computer Interface automatically selects the most appropriate person, after considering

several factors like availability, experience etc. to tackle a particular type of emergency.

The *netmanager* application shows the network administrator the overall network configuration. By a series of mouse-clicks and pull-down menus the administrator can delve deeper and deeper down the network for lower levels of information. A click on the leaf element allows the operator to see the system performance figures like CPU, Disk, Memory/Network interface utilization etc. A *Mgmtinfosetup* (Fig.7) utility aids the Administrator in selecting MIB objects for monitoring. The *monitor* utility interfaces with the expert system. The network operator may also directly access the Expert system for advice/help in an interactive manner at any stage by a mouse click on the related object. When there is a problem the Expert system guides the operator to the trouble spot, suggests diagnostics, and possible remedies using a check-and-see step-by-step interactive mechanism.

4.3 An example operation

Though there are several online displays of network behaviour, the system does not envisage a routine usage of these features. In a network the running system will be polling the agents obtaining information on the MOs and using the MIKB to check if there is a likelihood of some abnormality. There will be small icons on the manager's display screen to indicate the overall status of a network element. If there is a likelihood of some abnormality, then depending on the severerity the icon corresponding to that agent darkens. For example if an increase is detected in the count of ICMP destination unreachable packets, the system considers it to be an indication of possible abnormality and carries out further investigation. The manager application requests more details from the agent - in this case the packet dump routine analyses the source and destination and checks for similar occurences. If it has occurred before, the corresponding occurence counter is incremented, else the indication is sent to the manager. If several similar occurences are detected - it indicates that there is a likelihood of some interface failure - and the severity level is incremented. When the severity level goes beyond a specified threshold the operator is summoned - by means of a pocket bell or whatever option is specified in the NKB. The operator can always see the abnormalities in the network, if any, by clicking in the corresponding icon, whence the details of the abnormal objetcs, their history and present behaviou are displayed.

5 Conclusion

In this paper, we have presented an outline of an intelligent management, which targets 'hands off management'. The system takes care of the routine observation and decision making activity. The knowledge about 'what to look for' and 'what to make out from what is seen' is gathered from Network Experts, and put into the system. A simple and effective approach to the task of designing the Knowledge base and the subsequent acquisition of knowledge has been proposed and demonstrated. Taking the cue from the fact that in the existing network management framework the human managers' view of the network is limited to objects in the Management Information Base(MIB), the MIB is fixed as the core component in the KB design and as a viable starting point for knowledge acquisition. The significance of various managed objects(MO) as seen by a network manager was acquired and represented in the KB. The subjective view of the network manager is modelled in terms of the time development of MOs. In the model the judgment process takes the form of estimating the (ab)normality of an object by considering the value and/or time development of the object with some known (ab)normal state or threshold. To accomodate for the different levels of significance of abnormalities in different objects the concept of an abnormality weight was introduced. Also, the time development is considered in a long term and a short term perspective. In the course of building the KB it was found that discrete thresholding did not provide an accurate picture of the system. Thus the concept of fuzzy handling of thresholds was introduced. The MIB related component of the KB - the MIKB, was network independent. The other component of the KB- the NKB, comprised the network dependent part. The pilot system[26] is currently operational - it covers the SNMP MIB II and some enterprise-specific MIBs. The Knowledge base currently contains around 200 rules. Though the examples are given with reference to TCP/IP protocol suite, the same approach may be followed for other protocolsuites using the corresponding MIBs.

The development and maintenance of the Network Management Knowledge Base is a continual process. Currently, examination of the logbooks at the local network operation centres, brainstorming sessions of computer network professionals, and interviews with veteran computer network operators is being pursued for this purpose.

Work is also progressing in the area of configuration management[28]. For effective Configuration Management another orthogonal component of the knowledge base is required. It basically consists of a network map, containing all the relevant details. The map could be local or of a wider area depending on the intended reach. This will lead to an enhanced model of the knowledge base NMKB = MIKB $\cup$ NKB $\cup$ NCDB. The Network Configuration DataBase (NCDB) is a distributed database containing details of the network components (lines, nodes,... their names and properties). The preliminary pilot implementation is based on the X.500 directory services.

Acknowledgements

The authors acknowledge with thanks the valuable advice provided by Masato Sakata of Akita University and Norio Shiratori of Tohoku University. We thank Hideo Ogata of AIC Systems Laboratory for providing the encouragement and a congenial environment for the work. We would also like to thank the other computer networking professionsals of Tohoku University and of the WIDE Project for their continuing assistance through the various stages of this work. We would also like to thank the members of the AI Language Research Institute for the use the CESP interpreter developed by them.

References

[1] Nasser Modiri "An Implementation of the Common Network Management Information Service element Interfaces", IEEE Communications Magazine, July 1991,

[2] Katherine Jones, " Network management in the world of Standards: The Role of the SNMP Protocol in Managing Networks" Int. J. of Network Management, vol. 1 No. 1 pp. 5-13

[3] F. Halsall et.al. "An Implementation of an OSI Network Management System ", IEEE Communications Magazine, July 1990,

[4] U. Warrier et.al A Platform for Heterogeneous Interconnection Network Management IEEE SAC V.8 No.1 Jan'90

[5] A. Dupuy et.al. "NETMATE": A network management environment IEEE Network Magazine, March 1991, pp 35-43.

[6] C. Solomon, J. L'Haire , J. Paccini LAN management by cooperation : Hewlett Packard and the Univ of Geneva Computer Networks and ISDN Systems 23(1991) 79-85

[7] T. Sugawara , Ken-ichiro Murakami , A Multiagent Diagnostic System for Internetwork Problems Proceedings of INET'92, Kobe, Japan, June, 1992.

[8] Bruce L. Hitson Knowledge-Based Monitoring and Control : An approach to Understanding the Behavior of TCP/IP Network Protocols Proc of the ACM SIGCOMM'88 Stanford CA 1988, pp 210-221

[9] L. N. Cassel et.al Network Management Architectures and Protocols: Problems and Approaches IEEE SAC V.7 No.7 Sep'89

[10] H. Kobayashi et.al. Integrated Network Management System: NETM (Japanese) Hitachi Hyoron, vol. 73, No. 5, May, 1991.

[11] Kiriha Y. et.al. Fault Analysis Expert System for Unified Network management: EXNETS Nec Res & Develop., vol 33 No. 1 Jan 1992 pp 117-125

[12] Elise Gerich Management and Operation of the NSFNET backbone Computer Networks and ISDN Systems 23(1991) 69-72

[13] B. Stockman NORDUnet Experiences in Network Management Computer Networks and ISDN Systems 23(1991) 73-78

[14] Sugawara T., "A cooperative LAN diagnostic and observation expert system", Proc of IPCCC'90,pp 667-674.

[15] ISO/IEC JTC1/SC21/WG4 N3324: "Information Processing Systems - Open Systems Interconnection , Management Information Services- Structure of Management Information - Part 1: Management Information Model", Sydney, Dec. 1988.

[16] M.Rose "Structure and Identification of Management Information for TCP/IP-based Internets" RFC 1155, May, 1990.

[17] ISO 9596/1 "OSI: Management Information Protocol Specification - Part 1, Overview" 1989.

[18] ISO 9596/2 "OSI: Management Information Protocol Specification - Part 2, Common Management Information Protocol" 1989

[19] U.Warrier et.al. "Common Management Information Services and Protocol over TCP/IP (CMOT)", RFC 1189, October 1990.

[20] J.D.Case et.al. "A Simple Network Management Protocol (SNMP)" RFC 1157, May 1990.

[21] A. Ben Ari, A.Chandna, Unni Warrier "Network Management of TCP/IP Networks: present and Future", IEEE Network Magazine, July 1990.

[22] ISO/DIS 10164-5, "Event Report Management Function", June 1990.

[23] ISO/DIS 10164-6, "Log Control Function", June 1990.

[24] D.L.Mills "Network Time Protocol (version 2) specification and implementation", RFC 1119, September 1989.

[25] K. McCloghrie et.al. "Management Information Base for Network Management of TCP/IP-based internets", RFC 1156, May 1990.

[26] G.Mansfield et.al. "An SNMP-based Expert Network Management System", to be published, IEICE Transactions, August, 1992.

[27] CESP Language Guide Version 3.0, AI Language Research Institute, Ltd.

[28] G. Mansfield, "Configuration Management using the distributed Network Configuration DataBase", AIC Internal Technical note, June, 1992.

Appendix A Basic defined object types [2]

Status objects	4 (MIB type INTEGER)
Counter objects	70 (MIB types Counter, two from INTEGER)
Gauge objects	3
TimeTicks	2
Descriptive	35 (MIB types OBJECT IDENTIFIER, DisplayString,Networkaddress & INTEGER)

[2] source: RFC 1213: MIB II