Effectiveness of Autonomous Network Monitoring Based on Intelligent-Agent-Mediated Status Information

Susumu Konno, Sameer Abar, Yukio Iwaya, and Tetsuo Kinoshita

Tohoku University, 2–1–1 Katahira, Aoba-ku, Sendai, Miyagi, 980–8577, Japan
skonno@isc.tohoku.ac.jp
http://www.ka.riec.tohoku.ac.jp/ka/menber.index.en.html

Abstract. The growing complexity of communication networks and their associated information overhead have made network management considerably difficult. This paper presents a novel Network Management Scheme based on the novel concept of Active Information Resources (AIRs). Many types of information are distributed in the complex network, and they are changed dynamically. Under the AIR scheme, each piece of information in a network is activated as an intelligent agent: an I-AIR. An I-AIR has knowledge and functionality related to its information. The I-AIRs autonomously detect run-time operational obstacles occurring in the network system and specify the failures' causes to the network administrator with their cooperation. Thereby, some network management tasks are supported. The proposed prototype system (AIR-NMS) was implemented. Experimental results indicate that it markedly reduces the network administrator workload, compared to conventional network management methods.

1 Introduction

In recent years, computer communication networks have grown dramatically both in size and complexity. Moreover, they comprise heterogeneous multi-vendor environments. Traditionally, network management activities have been performed by network managers. However, these activities are becoming more demanding and data-intensive because of the rapid growth of modern networks. For those reasons, automation of network management activities has become necessary. For managing these huge distributed network systems, manual procedures have become tedious.

A typical approach to network management is centralized, static, polling-based management that involves high-capacity computing resources at the centralized platform including commercially available management tools. As managed components become more numerous, the amount of network traffic, which should be managed, have increased accordingly. Consequently, in centralized management systems, the management traffic might eventually oppress the network bandwidth. Even where the management platform uses several distributed

H.G. Okuno and M. Ali (Eds.): IEA/AIE 2007, LNAI 4570, pp. 1078–1087, 2007.

management stations, the huge bulk of management traffic remains concentrated around those stations [1]. The overwhelming volume and complexity of the information involved in network management imparts a terrible load [2].

Furthermore, in view of the dynamic nature of evolving networks, future network management solutions need to be flexible, adaptable, and intelligent without increasing the burden on network resources. The rapid of network systems has posed the issues of flexibility, scalability, and interoperability for the centralized paradigm. Even though failures in large communication networks are unavoidable, quick detection and identification of the causes of failure can fortify these systems, making them more robust, with more reliable operations, thereby ultimately increasing the level of confidence in the services they provide [3]. Motivated by these considerations, the proposed approach is intended to provide an intelligent, adaptive and autonomous network monitoring support paradigm for communication network systems.

A network monitoring support method based on the activated information is proposed in this paper. In this method, the distributed information in a computer network is activated using the concept of Active Information Resource (AIR). In the AIR scheme, each unit of distributed information has knowledge and functionalities related to utilization of the information resource as well as its information. In our experiment network system, each activated information AIR (I–AIR) is developed as an intelligent agent. The proposed framework simplifies network monitoring for the administrator. Experiments were performed to investigate the effectiveness of the proposed method.

The remainder of the paper is organized as follows. Section 2 presents an overview of the AIR concept and conversion of the dynamic status information as I-AIRs. The detailed design and implementation considerations of I-AIRs in the proposed prototype system are discussed in Section 3. Experimental results, along with the system's performance evaluations are outlined in Section 4. Finally, the conclusions and future issues are presented in Section 5.

2 Automated Network Monitoring Based on Activated Information

For monitoring of communication network by an administrator, much status information distributed in a network is required, such as network traffic, conditions of service processes, and application server logs. Commonly, the information is static; furthermore, an administrator must investigate them one by one, which places a necessary physical and mental load on the administrator.

In this study, therefore, this static information is activated to reduce the administrator's workload. For activation of status information, a concept of an active information resource (AIR) [4] [5] [6] is employed. Each unit of status information is wrapped as an AIR for activation; it is called I-AIR. An I-AIR has its original information resources along with related knowledge and functionalities. Several I-AIRs can cooperate autonomously based on their status information and knowledge. Consequently, our scheme can reduce network management loads

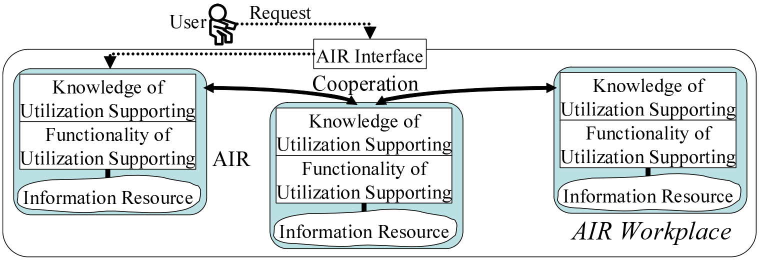
User
Request
AIR Interface
Cooperation
Knowledge of Utilization Supporting
Functionality of Utilization Supporting
Information Resource
AIR
Knowledge of Utilization Supporting
Functionality of Utilization Supporting
Information Resource
Knowledge of Utilization Supporting
Functionality of Utilization Supporting
Information Resource
AIR Workplace

Fig. 1. Active Information Resource

by presenting the dynamic status information of the network resources during automatic detection and specification of network failures.

2.1 AIR Concept

An AIR is defined as the distributed information resource enhanced with its knowledge as well as functionality to facilitate its resources. Figure 1 shows a conceptual model of an AIR with its support knowledge and functionality. The knowledge of an AIR typically consists of metadata of the information contents and their processing descriptions. The functionality of AIR is about how to analyze and process the users' query as well as defining the cooperation strategy among the multiple AIRs.

An AIR can be implemented using the multi-agent-based approach. Agent-based computing is known as a complementary way to manage the resources of distributed systems because of the increased flexibility in adapting to the dynamically changing requirements of such systems [7].

Essential features of AIRs include:

- To extract and process the information contents in response to the query from user (or another AIR) in a knowledge-based manner.
- To interact actively and mutually to make full use of the information contents, the embedded support knowledge, and functionality.

The effectiveness of AIR has been employed in the context of diverse web-based information retrieval techniques. The prototype systems have exhibited very promising results.

2.2 Applying the AIR Concept to Network Monitoring

Generally, the status information of the communication network is classifiable into two types: static information and dynamic information. For example, the relationship between IP addresses and Mac addresses, host names, domain names, IP-routing, etc., are included as static network information. On the other hand, the dynamic information includes number of packet traffic, RMON-MIB, SNMPv2-MIB, logs of network services, etc. To apply the concept of AIR to both types of information for network monitoring, each unit of information is converted to an AIR to form a so-called I-AIR.

Conventionally, an administrator collects various status information through periodical polling. She aggregates the data and decides the status of the network system using her know-how. This task can be disaggregated into three stages, such as detection, recognition, and specification of the failure. This task requires much experience as a network manager; therefore, a beginner cannot be employed as an administrator.

To support the empirical task of the administrator, an I-AIR includes diverse knowledge and functionality in addition to its original data. For example:

- meta-knowledge about information resources
- knowledge about failure condition (threshold)
- knowledge about cooperation with another I-AIR
- functionality to handle original data

Using this additional knowledge and functionality, I-AIRs can mutually cooperate. The following tasks can be partially supported by AIR:

- distributed and effective monitoring of network system
- detection of network failure using a threshold
- processing of information resources according to the failure with its functionality
- improvement of reliability of detection, recognition, and specification of the failure through cooperation among AIRs

These features can reduce the overall workload of the administrator.

3 Design and Implementation of I-AIR

In this section, the design of an I-AIR is discussed. The design comprises three vital ingredients: internal support knowledge, functionality for sharing the information contents, and specifications of the information resource itself.

3.1 Design of Knowledge in I-AIR

The support knowledge for sharing information contents is the empirical knowledge of network management which inspects the status information of the network for occurring faults. Essential components of this knowledge are as follows:

- I-AIR Identification Knowledge (ID):
  The ID includes an identification number, task number of I-AIR, etc.
- Knowledge about Information Resource (IR):
  The IR includes a type, an update-time, a format type, etc.

- Knowledge about Failure Inspection (FI)
  The FI includes two types of knowledge to inspect the failure: text information to be detected in logs, and a threshold of packets, etc.
- Knowledge about Network Periodic Investigation - Control Method (CM):
  The CM includes the polling time and other conditions for updating of the information resource.
- Knowledge about Cooperation Protocol (CP):
  The CP includes protocol sequences for cooperation with other AIRs.

The knowledge contained in an I-AIR as ID, IR, and CP is required mainly in order to operate on the information resource and facilitate communication and cooperation among the I-AIRs. The preeminent characteristic of I-AIR is its autonomous monitoring mechanism, which is supported via FI and CM for the inspection and investigation of the obstacles that hinder the normal network operation. Thus, the performance of I-AIRs in the proposed technique relies heavily on the design of various types of internal support knowledge.

3.2 Design of Functionality of I–AIR

I-AIRs' functionality deals with the sharing and processing of the information resource for cooperative problem solving during the active fault monitoring and detection phases. In this regard, the design of some essential features is crucial as follows:

- Functionality as an Interface to I-AIR internal support knowledge
- Functionality for processing the information resource
- Functionality for transmitting the processed results to other I-AIRs
- Functionality for inspecting the obstacle with respect to the pre-defined threshold

3.3 Design of Information Resource

Two I-AIR information resource types are described here.

- Simple text format
- RDF/XML syntax specification

The RDF/XML language is a W3C-recommended framework for describing information resources using machine-readable metadata, which brings about an unprecedented level of automation for the representation and retrieval of information. The plain-text format consists of log-information that is acquired through the syslog (a standard logging solution on Unix and Linux systems). In the proposed approach, the I-AIR functionality extracts a diverse type of log-information during operational management scenarios and converts it to RDF/XML format specifications.

Table 1. Examples of implemented I-AIRs for network monitoring

I-AIR No.	Function	I-AIR No.	Function
1	Network Disconnection detector	11	DNS server process checker
2	NIC config failure detector	12	SMTP server process checker
3	SPAM mail detector	13	POP server process checker
4	MSBlaster attack detector	14	DNS connection checker
5	Mail send/receive error detector	15	Network route to host checker
6	TCP/IP stack failure checker	16	Kernel information checker
7	NIC config failure checker	17	Lease IP address checker
8	HUB failure checker	18	Mail server error checker
9	Ruter failure checker	19	Number of SPAM mail
10	Communication failure checker		

3.4 Implementation of I-AIR

A multi-agent-based approach was adopted for implementation of I-AIRs in the proposed technique. For the effective realization of I-AIR support knowledge and functionality, the multi-agent system is a highly pragmatic choice. The I-AIRs realized with the software agents render the I-AIRs active, which, after being invoked by an outside event, can autonomously perform the task of cooperative problem-solving. The proposed system architecture is supported by an Agent-based Distributed Information Processing System (ADIPS) framework [8], which is a flexible computing environment for designing multi-agent systems. Table 1 illustrates the I-AIRs developed in this study.

4 Evaluate the Effectiveness of I-AIR in Actual Monitoring Task

To evaluate the prototype system's effectiveness, an experimental NMS system, called AIR-NMS, was set up in the laboratory. The network administrator per-

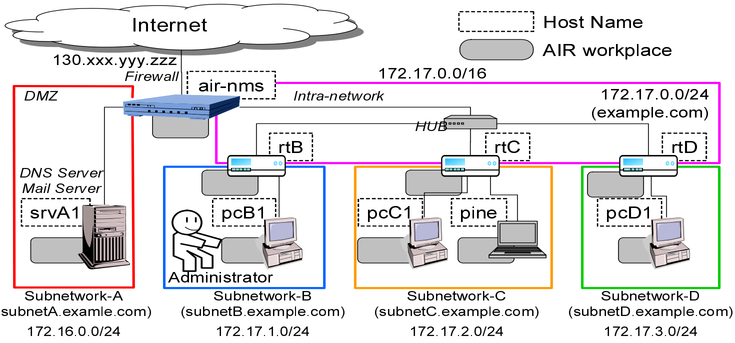

Fig. 2. Construction of Network Systems and AIR-NMS

forms the management task according to the conventional manual method, as well as with the I-AIRs based proposed system. He also measures the performance of the proposed approach adopted for the automation of network functions. In the experiment, the time and the number of procedures executed to correct the obstacle were measured after a network obstacle was reported to a subject.

4.1 Experimental Network

Figure 2 demonstrates the practical setup of the environment for experimenting with the I-AIRs. The network system comprises a 100BASE-TX Ethernet with a firewall configured as a Network Address Translation (NAT) firewall, a router, and various personal computers (PCs) arranged in four subnetworks. Subnetwork A is configured as a DeMilitarized Zone range 172.16.0.0/24. The server (sevA1) DNS and Mail application settings are configured. The other three subnetworks (B, C, D) have IP-addresses in the order given as 172.17.1.0/24, 172.17.2.0/24, and 172.17.3.0/24. Moreover, the network management console for managing the whole setup resides in pcB1 of subnetwork B. In subnetwork C, there is a desktop-type PC system (pcC1) with a fixed IP address from the DNS server, and a notebook computer (pine) which acquires the IP-addresses through the DHCP. In addition, Fig. 2 depicts the nodes (PCs, routers, firewall etc.) of the experimental network system. Each node shows the corresponding AIR workplace where the I-AIRs operate actively. For each node, about 15 AIRs were implemented. This implies that nearly 140 I-AIRs were incorporated within the experimental setup. A Linux operating system was used in each PC.

4.2 Experiment I: Various Application Scenarios

In this experimentation technique, several obstacle circumstances are generated and then inspected with and without I-AIR based system. These obstacles might occur by various causes. The task of a subject is to determine only one cause of a failure.

1. Cannot Connect to the Specific Host: In this case, file-transfer from pcD1 to pcB1 is not possible. A rare cause has been presumed, that is, a problem with the settings of Network Interface Card (NIC) of the host computer (pcB1).
2. Transmission of Spam Mail: In this case, a spam mail is transmitted from pcD1. However, the originating location of spam is concealed, so it is required to detect accurately the host that sends out the illicit message.
3. Slow Network: This delinquency is reported in the case of accessing World Wide Web (WWW) connection. The notebook PC (pine) was infected through an attack (from MSBlaster from outside source) at the port 135, thereby hindering its access to the Internet.
4. Mail Sending/Receiving Error: Here, the client network encounters the problem in sending/receiving email because the reason that the SMTP server process is down.

Table 2. Experimental results (Exp.1)

1. Cannot Connect to the Specific Host

	A		B		C		D		E		Average	
	Time	Step	Time	Step	Time	Step	Time	Step	Time	Step	Time	Step
no I-AIR	1056	20	756	20	680	22	771	20	282	40	709.0	24.4
I-AIR	99	5	51	2	125	4	226	5	52	2	110.6	3.6
$\frac{\text{I-AIR}}{\text{no I-AIR}}$(%)	9.4	25.0	6.7	10.0	18.4	18.2	29.3	25.0	18.4	5.0	15.6	14.8

2. Transmission of SPAM Mail

	A		B		C		D		E		Average	
	Time	Step	Time	Step	Time	Step	Time	Step	Time	Step	Time	Step
no I-AIR	1096	24	221	4	901	23	1155	26	92	5	693.0	16.4
I-AIR	49	3	93	3	83	4	129	2	40	2	78.8	2.8
$\frac{\text{I-AIR}}{\text{no I-AIR}}$(%)	4.5	12.5	42.1	75.0	9.2	17.4	11.2	7.7	43.5	40.0	11.4	17.1

3. Slow Network

	A		B		C		D		E		Average	
	Time	Step	Time	Step	Time	Step	Time	Step	Time	Step	Time	Step
no I-AIR	208	3	205	3	330	9	323	3	682	35	349.6	10.6
I-AIR	528	4	53	1	61	1	63	1	94	1	159.8	1.6
$\frac{\text{I-AIR}}{\text{no I-AIR}}$(%)	253.8	133.3	25.9	33.3	18.5	11.1	19.5	33.3	13.8	2.6	45.7	15.1

4. Mail Sending / Receiving Error

	A		B		C		D		E		Average	
	Time	Step	Time	Step	Time	Step	Time	Step	Time	Step	Time	Step
no I-AIR	996	31	369	16	680	22	565	7	1499	49	821.8	25.0
I-AIR	98	4	59	2	125	4	81	2	73	2	87.2	2.8
$\frac{\text{I-AIR}}{\text{no I-AIR}}$(%)	9.8	12.9	16.0	12.5	18.4	18.2	14.3	28.6	4.9	4.1	10.6	11.2

Management experience: A. 1 year, B. 2 year, C. 2 year, D. 3 year, E. 7 year

Results. The experimental results were compiled into Table 2. The results show that, for each failure situation, with the inclusion of I-AIRs, the management load related to the time taken to resolve a certain fault as well as the number of steps necessary to locate the cause of failure was reduced to an average 20%.

4.3 Experiment II: One Obstacle from Various Causes

An application scenario is tested against various causes for the occurrence of a specific failure condition to demonstrate the flexibility of the proposed approach using I-AIRs. Furthermore, these causes do not occur necessarily in any fixed pattern. The checks to detect these causes are performed randomly. However, using I-AIRs is advantageous because every check is done only once during the course of the fault-localizing process. The failure cause is detected and the main cause behind the failure is reported to the network operator actively.

Table 3. Assumed failure causes: Mail Sending/Receiving Error (Exp.2)

Problem	causes
Cable problem	a. Cable was disconnected.
Port problem	b. The 25th port was closed.
	c. The 110th port was closed.
DNS Server problem	d. DNS Server process was downed.
	e. Config was not available.
Mail Server problem	f. Mail Server process was down.

Table 4. Experimental results among individual administrators (Exp.2)

	F			G			H			I			J		
		Time	Step		Time	Step		Time	Step		Time	Step		Time	Step
no I-AIR	d	158	9	b	566	8	e	929	23	f	235	5	a	655	19
	e	743	24	d	871	12	b	339	9	c	615	9	f	182	5
I-AIR	a	51	1	f	104	2	c	82	3	a	40	1	b	86	2
	f	85	4	c	106	2	d	52	3	e	74	2	e	128	6
$\frac{\text{I-AIR}}{\text{no I-AIR}}$(%)		151.1	15.2		14.6	20.0		10.6	18.8		13.4	21.4		25.6	33.3

Management experience: F. 2 year, G. 2 year, H. 3 year, I. 3 year, J. 7 year

Table 5. Experimental results among individual failures (Exp.2)

	a		b		c		d		e		f	
	Time	Step	Time	Step	Time	Step	Time	Step	Time	Step	Time	Step
no I-AIR	655	19	566	8	615	9	158	9	743	24	235	5
	–	–	339	9	–	–	871	12	929	23	182	5
I-AIR	51	1	86	2	106	2	52	3	74	2	85	4
	40	1	–	–	82	3	–	–	128	6	104	2
$\frac{\text{I-AIR}}{\text{non I-AIR}}$(%)	6.9	5.3	19.0	23.5	15.3	27.8	10.1	28.6	12.1	17.0	45.3	60.0

Table 3 depicts the failure situation "Mail Sending/Receiving Error" with some possible causes underlying the occurrence of this anomaly. The task of the subject is to determine the cause of this error.

Results. Experimental results computed by each manager while resolving the mail sending/receiving anomaly were compiled into Table 4. Additionally, the results corresponding to each failure cause were accumulated into Table 5. The results demonstrate that the network management overhead regarding the time taken to resolve a certain fault, along with the number of steps necessary to locate the cause of failure, were reduced to 20% on average, which concurs exactly with the results of Experiment 1.

5 Summary

This paper presented a novel technique for the automation of management tasks for communication network systems. The foundation of the proposed framework is the use of I-AIRs, which, through active mutual interaction and with the functional network system, can resolve various network-failure situations. A part of the I-AIR knowledge is modified dynamically and frequently according to the operational characteristics of the network. Moreover, experimental results demonstrated a marked reduction in the administrator workload through the use of the proposed automated network monitoring and fault detection functions.

Acknowledgement. The authors thank Mr. Satoshi YOSHIMURA for his contribution to this study.

References

1. Stephan, R., Ray, P., Paramesh, N.: Network management platform based on mobile agent. International Journal of Network Management 14, 59–73 (2003)
2. Consens, M., Hasan, M.: Supporting network management through declaratively specified data visualizations. In: IEEE/IFIP 3rd International Symposium on Integrated Network Management, pp. 725–738 (1993)
3. Bouloutas, A.T., Calo, S., Finkel, A.: Alarm correlation and fault identification in communication networks. IEEE Transactions on Communications 42(2,3,4), 523–533 (1994)
4. Kinoshita, T.: A method for utilizing distributed information resources effectively: Design of active information resource (in japanese). In: Technical Report of IEICE (Japan) AI99-54, 13–19 (1999)
5. Li, B., Abe, T., Sugawara, K., Kinoshita, T.: Active information resource: Design concept and example. In: The 17th International Conference on Advanced Information Networking and Applications, pp. 274–277 (2003)
6. Li, B., Abe, T., Kinoshita, T.: Design of agent-based active information resource. In: The 1st International Conference on Agent-Based Technologies and Systems, pp. 233–244 (2003)
7. Papavassiliou, S., Puliafito, A., Tomarchio, O., Ye, J.: Mobile agent-based approach for efficient network management and resource allocation: Framework and applications. IEEE Journal on Selected Areas in Communication 20(4), 858–872 (2002)
8. Fujita, S., Hara, H., Sugawara, K., Kinoshita, T., Shiratori, N.: Agent-based design model of adaptive distributed systems. Journal of Applied Intelligence 9(1), 57–70 (1998)