An Efficient and Cost-Effective Approach to Manage Network Elements in a Campus like Environment

Srinivas.M.N
PG Student, Dept of PGS-CEA
National Institute of Engineering
Mysore, Karnataka, India
cena10.td@gmail.com

Girish
Asst. Professor, Dept of PGS-CEA
National Institute of Engineering
Mysore, Karnataka, India
hpgirish@yahoo.com

Marx.R
Manager, Software Group
Bharat Electronics Limited
Bangalore, Karnataka, India
marxr@bel.co.in

***Abstract*— with the changing trends and new developments, designing an Element Management System (EMS) for efficient management of network elements is a challenging task. Therefore a good design is essential to make such a system. The primary issues of element management are taken into consideration in this work and the main functions of an EMS are implemented. The implementation is web based, provides ease of access and is robust in nature. The implementation uses open source technologies, open source API's and lower cost tools. In case of client machines (hosts), the SNMP service facilitated by the OS is used, resulting in enhanced efficiency. In this implementation, when a new element is added, a separate instance of the functional module is created for that element. This improves scalability as any number of new elements can be added easily. The solution is implemented using Java and is highly portable, i.e., the solution can be used without modification on multiple platforms. Data integrity is preserved by the solution implementation. This implementation exposes its functionalities through a well-defined API and can be extended to be compatible with systems from different vendors. Interactive User Interface (UI) and customized event viewing provided by this implementation enable the user to use the deployed system effectively.**

Keywords*— *IP, ICMP, TCP, UDP, SNMP, Element Management System, API, Operating System, Java, C++, URL

I. INTRODUCTION

EMS is concerned with managing network elements of the same type. In the current trends and ever changing technologies, providing efficiency and meeting the demands of availability, reliability and scalability is a difficult task and it requires a lot of effort and dedication from the developer. At present, a number of challenges exist in element management of campus like environment which are addressed in this paper. The challenges faced earlier in developing a management system for campus like environment are; EMS for campus was developed in object oriented languages such as C++ which offered faster execution but lacked portability and state of the art features. In addition, memory leaks and usage of Operating System-dependent threads caused problems. Other problems included less focus on alarm generation and alarm diagnosing [1], less attractive interfaces developed, and insufficient statistics and reports generated for future analysis. Element management has a high priority in campus like environment but the development is limited due to lack of investments and dedicated personnel. In this paper, the management system described to manage the campus like environment addresses the above mentioned issues. The Java platform is used, as it's simple, easy to code, portable, avoids memory leaks, multithreaded, has a good feature set and a rich set of API's and is an open source technology. Many development software (IDE, Runtime environments, etc.,) used with java are open source and are available free or at a very low cost. Other advantages of the implementation described in this paper are as follows: The deployment is web based, which allows remote access to the application. Web technologies bring valuable cost improvements, flexibility, and security enhancement to the configuration management and performance management [9]. Data is centralized and secure. Information is accessible to a wide audience, available 24/7. The management protocol used is SNMPv2C. This version is chosen as it provides improved error handling and improved set of commands than the previous versions. SNMP4J API is used as it is an open source API, it has a good set of functions and ample examples are available on web. Security is provided through authentication and authorization. Data integrity is maintained with the help of MySql database server. MySql is open source, fast and robust. It is also easy to install on any operating system. The essential features required to manage a system efficiently in a campus are:

- Multi level thresholding and advanced alarm diagnosing, which describes alarms in full detail, severity of the alarms with suggestions for probable causes and remedies.
- Real time management data is reflected in the form of statistics and graphs which can be helpful in future analysis and research. They are available to the user even when the application is not running. A vital role is played by the statistics and data collected in future enhancements.
- EMS is deployed redundantly and can scale to support hundreds of concurrent users by simply adding processing power.
- A help document that provides overview of EMS and brief instructions on installation of the EMS has been

created. Functionality of the EMS is explained visually with the help of graphical data.

This article is structured as follows:
The background and concise description of the EMS is presented in section I. The Structural design and technical framework of the EMS is presented in section II and III. The implementation of the EMS is done in section IV. In addition the results are also presented in section IV. The work is concluded in section V .Enhancement to the work is also proposed in section V.

II. OVERALL STRUCTURAL DESIGN OF EMS

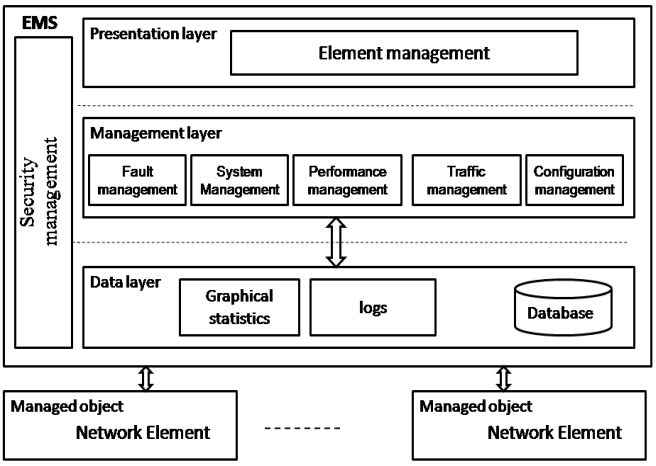

Figure 1. Structure of Element Management System

A good architecture should provide conceptual integrity for a system. The overall structure depicts n-tier client/server architecture, where the EMS is depicted as server and managed objects as clients. In the proposed structure, any number of managed objects can be handled by the server. As exposed in Figure 1, the EMS structure is split into presentation layer, management layer, data layer.

1) Presentation layer: The main function of this layer is to obtain user inputs, create tasks for the system, and translate tasks and results to something that a user can understand.

2) Management layer: The core functionality of the EMS exists in the form of Plain Old Java Classes in the management layer. The EMS is based on a centralized structure which facilitates improved security, consistency and accuracy of management data. The management functions which are required for the efficient functioning of the EMS are complemented by the ISO defined network management tasks as shown in Figure 1.The sub modules in the management layer are: fault management, system management, performance management, traffic Management, configuration management and security management [1].
a) *Fault management:* Fault Management can be acknowledged as the most vital part of any management system. The system stability depends on how well the faults are handled. The EMS deployed facilitates efficient fault management, by providing real time fault information which is collected from each individual element present in the network. Real time alarms and traps are generated if a threshold value for errors is crossed (or) flaws in the network are discovered. Multi-level thresholding and advance alarm diagnosis provides relief to the user by describing alarms and severity of the alarms in full detail with suggestions for probable causes and remedies. A separate thread/process is created for receiving traps. The important OID'S in Management Information Base (MIB) which play a vital role for in-depth fault analysis are listed in the form of table and their brief description is given for better understanding and future analysis. Real time monitoring for faults is done by polling the network every 30 seconds. This time interval is user configurable. Real-time fault information collection manages to receive a variety of fault alarm information from managed network via Trap, Sys log and active polling [1]. The fault information is reflected in the form of statistics and attractive graphs to the user.

b) Performance management: performance management should optimize the network to carry traffic with speed and reliability. Capacity planning is an important aspect in performance management. Real-time performance measurement is done based on packet in-flow, out-flow and error rates of packets [10], [1]. In the EMS implemented, IP, ICMP, TCP, UDP and SNMP packets are monitored in-detail in a campus like environment. Packet rate and error rate of each element helps us to identify the congestion point and the overloaded system in the network. Real-time performance data are collected and correlated and are used for report generation. Detailed reports and attractive graphs are generated based on detailed statistics of packet information and error information. Packet loss is a useful metric in performance measurement. Alarms are generated based on the packet loss and error rate of packets. Health status monitoring of all elements in the network is done at constant intervals.

c) Traffic management: Real-time IP, ICMP, TCP, UDP and SNMP traffic are managed with the help of packet handler sub module [1], [10]. The packet handler plays a packet analyzer role. Traffic statistics are reflected with the help of logs and graphs. Dynamic updating helps in effective traffic report generation and future analysis. The traffic error rate is supplemented by the inflow and outflow of IP, ICMP, TCP, UDP and SNMP packets. Traffic analysis helps in solving congestion problems. Alarms are generated based on error rate of traffic and suggestions and probable remedies are provided to the user to overcome the situation. This time interval is user configurable.

d) System management: System manager module gathers detailed information about client's system properties and settings and displays it in an extremely comprehensible manner to the user.

The main categories of information captured are described as follows:

- A textual description of the entity value which includes the full name and version identification of the system's hardware type, software operating-system, and networking software.

- An administratively-assigned name for this managed node.

- The physical location of this node, etc.

e) Configuration management: Configuration management includes functions to exercise control over, identify, collect data from and provide data to Network Elements. Auto discovery of clients connected and periodic health status monitoring of all elements connected in the network is performed. Automatic alarms are generated when elements are unreachable.

f) Security management: security is provided in the form of authentication. Access is denied to unauthorized users. Confidentiality and integrity of network information is maintained by storing all the confidential data in a database and access to the database is not possible without the administrative password. The administrator can assign access permissions to various users based on their requirement at that instance.

3) Data layer: Data layer includes functions to effectively store and retrieve data from the database. Data integrity is also maintained. The administrator is the sole person to modify the data. Access to database by other users is restricted. The data stored in the database includes, access permissions IP addresses and port numbers of all the elements in the network. Statistics and graphs on performance data, packet loss, element's system description, traffic data, alarms and traps received are also accumulated in the data layer.

III. SYSTEM TECHNICAL ARCHITECTURE

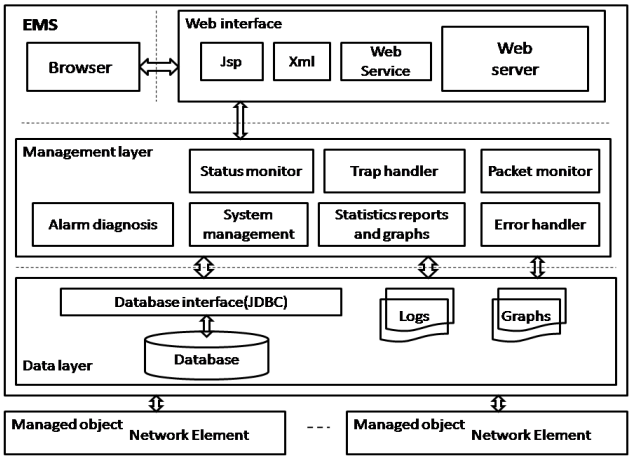

Figure 2. Technical architecture of EMS

The EMS put into operation is web based [9], [12], providing convenience and combined power of desktop and server applications with the accessibility of 24 hrs. The web application functions uniformly. As shown in Figure 2, the application runs on Apache web server. The user can manage the clients remotely with the help of a browser and http protocol [4]. Open source technologies and APIs are used for the implementation. The web interface is built using JSP, XML and web service technology [1]. Using Web services, the same core functionality can be used to build a console application .The data collected may be network data, performance data, fault data , logs graphs, etc., [1]. The management layer provides the core functionality of the EMS. Customized event viewing and help document prove valuable to a new user. The confidential data is stored in the database. In the application, complex data types can be passed to and from web services without losing data integrity. When passing objects to web services the data is passed for the object's fields, but the code is not. The interface used between the core functionality and the database is JDBC, which is an open source technology. The web application was made attractive by using buttons and images developed using Flash CS4 .The image galleries were developed using CSS. The embedding of flash developed buttons and images in the application increased look and feel of the application to a great extent. SNMP4J is the API used by the EMS to communicate with the managed object.

IV. EXPERIMENTS AND RESULTS

A. *Experimental Environment*

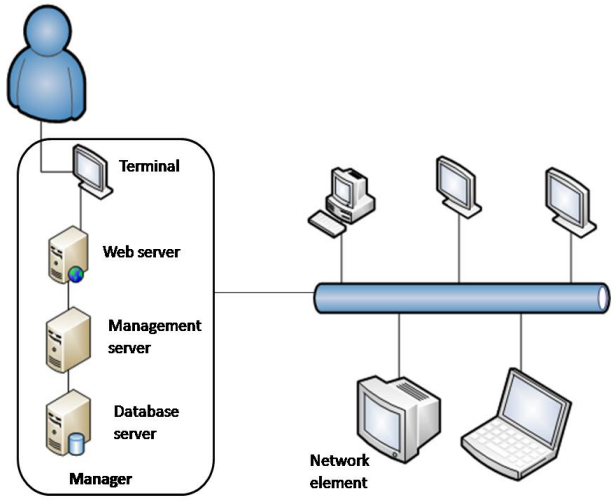

Figure 3. EMS implementation

For the experimental evaluation, we have deployed the EMS in a small campus as shown in Figure 3. A total of six nodes are interconnected through100 Mbps Ethernet using RS232 cable. Windows Xp is the operating system installed in all the elements. SNMP service is activated in all the elements and the default community name of the SNMP agent is set to 'public' with read-write access permissions. SNMP service is restarted before further proceedings. The latest version of a browser and flash plug-in, preferably opera 11 and flash 10 is

installed on the management server. The management server collects network management data, performance data, traffics data, and fault data etc., and stores the data in a database or a file [1]. During the installation of MySql server, the new URL, database name and password of the database are noted down and the new values are updated in 'config.properties' file. Exchange of management information between the management server and agent takes place with the help of SNMP4J API.

B. Results

The application is started with the help of a browser. At the first glance, we can observe the look and feel of the Graphical User Interface (GUI) is considerably increased by employing Flash and CSS technology. After the login page is opened, the correct username and password with administrator privileges is entered and the 'login' button is clicked. If the authentication is successful, the user can use the management functionality. In case of failure, user can retry with a different username and password as shown in Figure 4.

Figure 4. Login page of EMS

After successful authentication, the user can add all the IP addresses and port no's of the elements present in the network. Error messages are displayed in case of duplicate and invalid entries. '+' button can be used to add an element as shown in Figure 5 and the form to add an element is displayed as in Figure 6. The blank menus in Figure 5 can be used to add new devices to the network as part of future enhancement. The elements displayed in 'red' are unreachable. All the different types of elements present in the network can be seen by clicking the' NE'S PRESENT' button as shown in Figure 5 and the elements present in the network are displayed as in Figure 8.

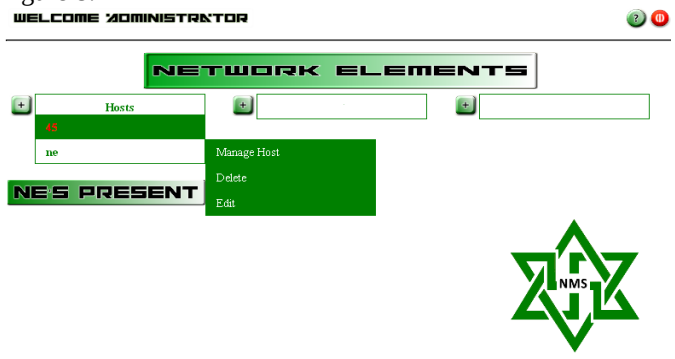

Figure 5. Menu page after successful login

The user can edit the values of an element by clicking the 'edit' submenu shown in Figure 5 and the form to edit the elements values is displayed as in Figure 7. The element can be permanently deleted from the network by clicking the 'delete' button as shown in Figure 5. 'Manage host' tab can be clicked as shown in Figure 5, to get brief information about the element as in Figure 9. Detailed management information can be viewed as in Figure 10 by clicking "Detailed Analysis" button as shown in Figure 9.

Figure 6. Form to add a new element into the network

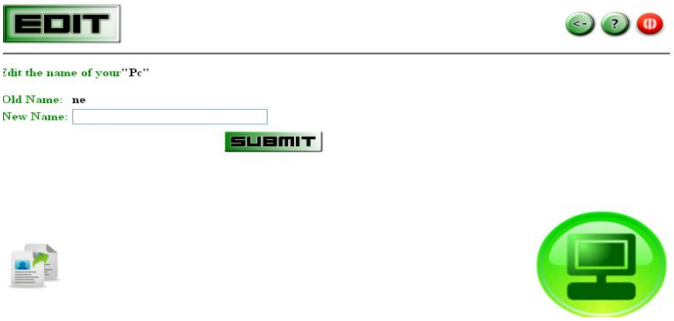

Figure 7. Form to edit an element's name

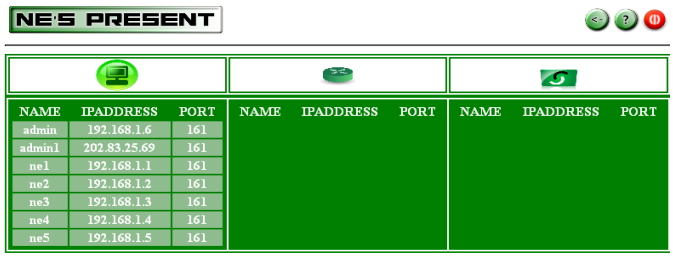

Figure 8. All the network elements present in the network

The information shown in Figure 9 includes parameters and statistics which reflect the performance of the system. The information displayed is as follows;

- Uptime, name, IP address and port number of the device.
- The packets inflow and outflow. Error information and packet loss of packets captured.
- Graphical Statistics and logs.
- Alarms, severity of the alarms supplemented by probable remedies for the problems, etc.

The image gallery shown in Figure 9 and 10 is built using CSS. The image can be viewed in full size by clicking on the image. The logs related to 'system information' can be viewed by clicking on the 'system information button. Similarly other

related logs can be viewed by clicking on their respective buttons. The replica of graphs and logs is shown in Figure 12.The Traps received can be viewed as in Figure 11 by clicking the 'Trap receiver' button as shown in Figure 9. At any point of time if the user wants to exit the application then he can click the logout button as shown in Figure 5. In case help is required during the course of time the user can click the '?' button as shown in Figure 5 to view the help document.

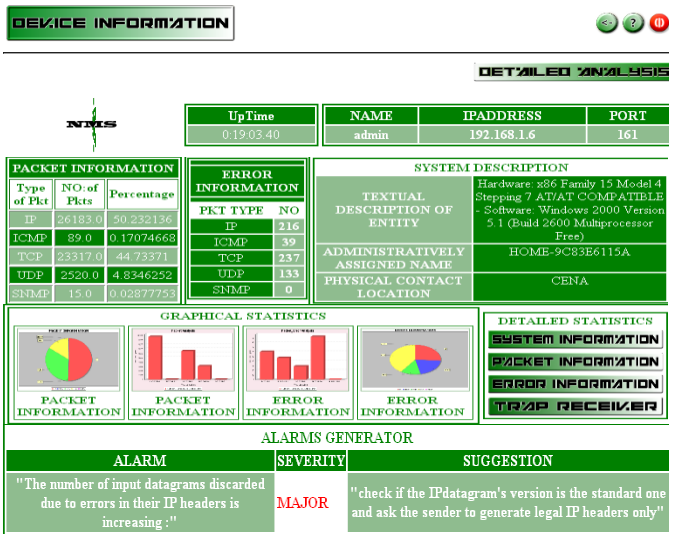

Figure 9. Brief Managed information of a particular element

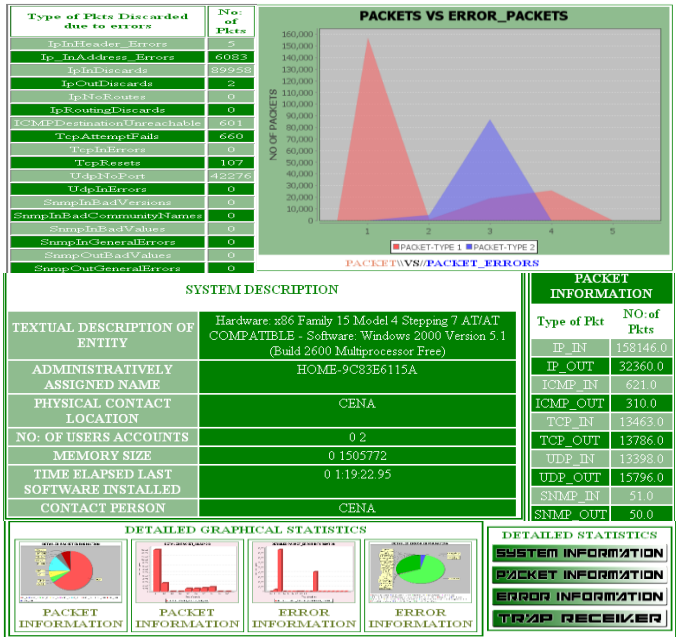

Figure 10. Detailed management information of a particular element

Figure 11. Traps received and its severity

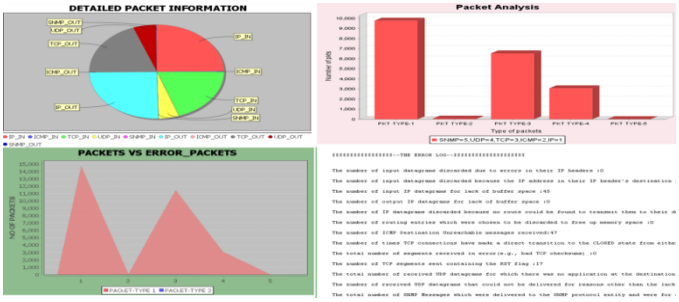

Figure 12. Graphical statistics and logs of a particular element

V. CONCLUSION AND FUTURE SCOPE

In this effort we have focused on developing an efficient EMS for campus like environment. This work presents a good design and framework for EMS resulting in reduces cost. The complete implementation is done using lower cost tools. The code is built using Java technology and is portable. The use of JSP, CSS and Flash web technology helped in creating an attractive web interface. In future the author is trying to enhance the security related features by upgrading SNMP V2c to SNMP V3. Further the author is planning to develop EMS for managing other types of network devices and integrate all the EMS's to form a NMS. In addition the author is interested in developing an NMS in a wireless environment.

REFERENCES

[1] Xiaosong Wang, "Studies on Network Management System Framework of Campus Network", 2nd International Asia Conference on Informatics in Control, Automation and Robotics, pp. 285-289, Car 2010.

[2] Chien-Chung Shen,"A Network Management Architecture for Battlefield Networks" ATIRP, pp. 1226-1231, IEEE 1997.

[3] G. Mansfield, M. Murtha, K. Higuchi, K. Jayanthi, B.Chakraborty, Y. Nemoto and S.Noguchi," Network Management In a Large-scale OSI-based Campus Network using SNMP", IEEE 1992.

[4] Hwa-Chun Lin and Chien-Hsing Wang,"Distributed Network Management by HTTP-based remote invocation", Global Telecomrnunications Conference – Globecorn '99.

[5] Jae-Kyu Chun, Ki-Yong Cho, Seok-Hyung Cho, Young-Woo Lee and Young-Il Kim," Network Management Based on PC Communication Platform with SNMP and mobile agents" Proceeding of the 22nd international on Distributed systems workshop,(1CDCSW'02),IEEE 2002.

[6] Annie Ibrahim rana,"New Roles of Policy –based Management in Home Area Networks-Concepts, Constraints and Challenges",IEEE 2009.

[7] Jong-Wook Beak," ATM Customer Network Management Using WWW and CORBA Technologies". IEEE 1998.

[8] Marcus Burner,"Probabilistic Decentralized Network Management " , IEEE 2009.

[9] Md. Jakir Hossen, Abd Rahman Ramli, and Mohd. Khazani Abdullah,"Web-based Network Device Management Using SNMP Servlet",IEEE2003.

[10] Harry Li and Guangjing Chen," Wireless LAN Network Management System",IEEE 2004.

[11] Panos Trimintzios, "DiMAPI: An Application Programming Interface for Distributed Network Monitoring", IEEE 2006.

[12] M. F Zhang, "Design and Research of the Network Management Model based on Web," Computer Engineering, 2003, 29 (20), pp. 104-106.