# Agent-based Model of a Responsive Network Monitoring System for Internet Service Providers

Osongi Felix Agik
Department of Computing and Informatics
University of Nairobi
Nairobi, Kenya
Email: fosongi [AT] gmail.com

Elisha T.O.Opiyo
Department of Computing and Informatics
University of Nairobi
Nairobi, Kenya

William Okelo-Odongo
Department of Computing and Informatics
University of Nairobi
Nairobi, Kenya

Bernard Manderick
Artificial Intelligence Lab
Vrije Universiteit Brussel
Brussels, Belgium

*Abstract*— **The widely used protocol based Simple Network Management Protocol (SNMP) is today particularly limiting due to the client-server centralized and static paradigm. The approach is centralized as it relies on a limited set of capabilities at network nodes while management processing has to be performed at the network management station. This requires transferring of large amounts of management data between the manager and the managed nodes. In this paradigm, the system does not auto respond to changes and modifications in the network as hard coding is required to make the necessary changes making the system inflexible. Significant attempts have been made to develop solutions like Remote Monitoring (RMON), Cisco Discovery Protocol (CDP), Cisco Net flow Accounting and Monitoring by Delegation (MbD), as an attempt to address the underlying challenges faced. Distributed Object based approaches like the Common Object Request Broker Architecture (CORBA) and later by Remote Method Invocation (Java-RMI) have been developed to address some of the underlying problems of the SNMP protocol [17]. These solutions do not however address dynamism, interoperability and responsiveness required of current networks [13]. The MOMENT (Monitoring and Measurement in the Next Generation Technologies) makes bold contributions about the things that ensure that network researchers will gain a better understanding of current networks as well as of The Network of the Future. But in our approach we use software agents to collect and propagate statistics from different Points of Presence to the Network Operations Center; Update changes in the Network; process and present results. With the agent paradigm in our system, the approach is able to increase system's responsiveness; to increase system flexibility; to take advantage of distributed computing resources, and to introduce interoperability, modularity and dynamism into the monitoring system. In this paper, we describe the system architecture and implementation of the responsive model. In our MAS model, the system is seen to reduce the time taken to initiate and update client parameters in the system; releases time for the administrators that can be used in other productive activities; eliminates mistakes that could be done by the administrators as the system has controls that ensure the agents only use the exact information as provided from the database; the system optimizes responsiveness by ensuring changes done on the database are automatically updated to the agents database which makes the relevant changes on the graphs and monitored parameters of the clients. This ensures the system is responsive to changes, increases flexibility and dynamism.**

***Keywords- MAS; SNMP; ISP; SLA; QoS; POP; NOC***

## I. INTRODUCTION

Information Communications and Technology industry has been very dynamic and has experienced dramatic growth in the last couple of decades. The ICT Service Providers have been tasked by different companies to develop and deliver connectivity using different methodologies and technologies. Different applications are being integrated which require strict Service Level Agreements and are hence prompting for very responsive methods of monitoring and reporting. The Internet Service Providers (ISPs) in the industry use different methodologies to monitor and respond to requirements in their supported Networks. Network monitoring and maintenance was traditionally not taken so seriously but with new developments and technologies, organizations are requiring strict Service Level Agreements (SLA's) that prompts the need for advanced monitoring and management systems by the providers [2]. Network users are increasingly having high expectations of reliability and quality of service (QoS) that the networks offer. Most of today's organizations rely on the Internet for their day to day activities and thus lay a huge reliance on it through the ISPs hence a few minutes of network breakdown would incur huge losses to the organization. There are different tools that have been developed to try to solve the network monitoring problems that has rendered most networks unreliable. However, these tools do not turn out to be optimal as they cannot solve the flexibility requirements of a complex network system. Multi-agents system is characterized by intelligence, abstraction, interoperability, modularity and dynamism [8]. These qualities are particularly useful to promote open and more robust system. The agents' modularity allows a complex system to be broken down into simpler subsystems, so that processes inside the subsystems are simple, and that the processes between subsystems are manageable. The agents are also intelligent and can learn changes and update automatically which increases responsiveness [5]. The Internet Service Providers is used as a case study to implement multi-agents approach in this study as they are majorly rolling out networks and hence engaged in monitoring them. In this case study, different agents will be used to represent different functional areas in the framework. The intelligence, reactivity, mobility and proactive characteristics of multi-agents achieve responsiveness and accomplish the ISP network monitoring requirements. Nonetheless, there are still interesting research issues surrounding the end to end monitoring of provisioned links. The problem of distributed networks can easily be solved by using software agents which can be mobile across the networks [1]. Hence, we developed and implemented an agent-based platform and a management and user application for responsive network monitoring [18]. The main features of our system are: capable to make fast and accurate updates to the monitoring system; to fasten the process of initiating new monitoring parameters; to allow efficient communications between agents due to the reduced overheads during communication. The objectives of this paper is to find out existing network monitoring methods and their associated challenges; to investigate how software agents can be used to implement Network Monitoring; to Simulate a MAS Model of a responsive Network Monitoring system for experimentation.

## II. RELATED WORK

Mobile Agents as an approach has been investigated by many researchers and this paper analyses substantive findings, as well as theoretical and methodological contributions to Network monitoring [11].

Some of the previous work that has been proposed by various authors is a client-server Agent-based intelligent network monitoring system. This framework consists of a responsive network monitoring system and an agent-based system that handles the information flow and propagation of usage and uptime statistics. In this study we do not require more administrative work in initializing and updating network monitoring parameters as the Agents get details from the database and automatically does the work hence increasing flexibility, responsiveness and saves time. The users do not need to install any agent execution environment on their computers to be able to use the system.

Another important distinguishing feature of our sys tem is that we are capable to develop as many agents as possible to be able to work together to achieve the same goal hence reducing time taken to execute a task especially while searching a large database. Sathya (2008), made bold contributions about the things we know to make sure that network researchers will gain a better understanding of current networks as well as of The Network of the Future [9]. By bringing together existing pan-European network monitoring infrastructures, MOMENT (**Monitoring and Measurement in the Next generation Technologies)** tends to have mobilized the European key stakeholders who can make a difference and enhance our understanding of the network. The MOMENT project was meant to integrate the existing measurement and monitoring infrastructures towards a common and open, pan-European platform. The project was to achieve a semantic representation and retrieval of measurement and monitoring information. It was also to develop and demonstrate a set of tools and applications for the future Internet taking advantage of the integrated approach. The key issues addressed by the project can be summarized as follows: The protocol that serves for the applications to perform semantic queries to the mediation engine through the query interface using web services; the monitoring services to subscribe through subscription interface; the interface or wrapper that the measurement infrastructures should use in order to register the offered service and data. For example, XML could be used to specify the data; configuration interface for communicating with management tasks. Liotta (2001) discusses "One necessary step towards the realization of active distributed monitoring its implementation and experimentation on prototype networks or on a real networked system. Real measurements will enlighten the actual behavior of the proposed approach with regard to overheads, stability and complexity. Cisco Network Management System; Best Practices White Paper analyzes The International Organization for Standardization (ISO) network management model. It defines five functional areas of network management [3]. The document describes the functional areas. The overall purpose of the document is to provide practical recommendations on each functional area to increase the overall effectiveness of current management tools and practices. It also provides design guidelines for future implementation of network management tools and technologies. The network management architecture includes the following: Simple Network Management Protocol **(SNMP)** platform for fault management; Performance monitoring platform for long term performance management and trending; CiscoWorks2000 server for configuration management, syslog collection, and hardware and software inventory management [4]. The architecture is designed on Client Server architecture with SNMP server sending traps to the clients. A central point of failure is seen to be created if the server malfunction rendering monitoring impossible in the whole network.

## III. AGENT BASED MODEL

Back in 1998, emerging agent paradigms and enabling technologies were considered a key for the implementation of highly responsive and scalable solutions that add a degree of openness to the telecommunications industry. In retrospect, agent technology suffered from different terminology and heterogeneity of technical approaches due to the lack of standards [12]. In the recent times, agent-based technologies are however considered the most promising means to deploy enterprise wide and world wide applications. This is because they offer the high-level software abstractions needed to manage complex applications and because they were invented to cope with distribution and interoperability. Networks are becoming widely distributed and complex hence the need for multi-agent systems that can be used to monitor and manage complex systems [7]. Usage data received and uptime statistics from various Points of Presence are propagated to the network Operations Center autonomously. The use of multi-agent system will promote flexibility hence reduce the time always taken by network administrators in making manual ad-hoc configurations. The integration of different Points of Presence of a service provider with different routing matrix makes the system to be complicated hence the suitability of multi-agent systems. Automatic initiation and updating of monitoring parameters can be best achieved by agents that have the ability to communicate information in a timely manner.

## IV. ANALYSIS

In this paper we started by analyzing the current mostly used monitoring tools, SNMP and Nagios NPM and then analyzed the MAS monitoring system before drawing any conclusions. There are requirements needed in the current system to enable an administrator initiate monitoring of any link, these are as below:

*a)* *SNMP server Software*, This can be licensed or free software .

*b)* Web server for Monitoring Client links froma web browser.

*c)* SNMP enabled Client Premise Equipment (CPE).

*d)* SNMP string configuration for Monitoring.

Figure 1 below shows how most of the Internet Service provider (ISP) monitoring systems are designed. The Network Monitoring System (NMS) has to communicate with all nodes in the network in order for it to report the required parameters.

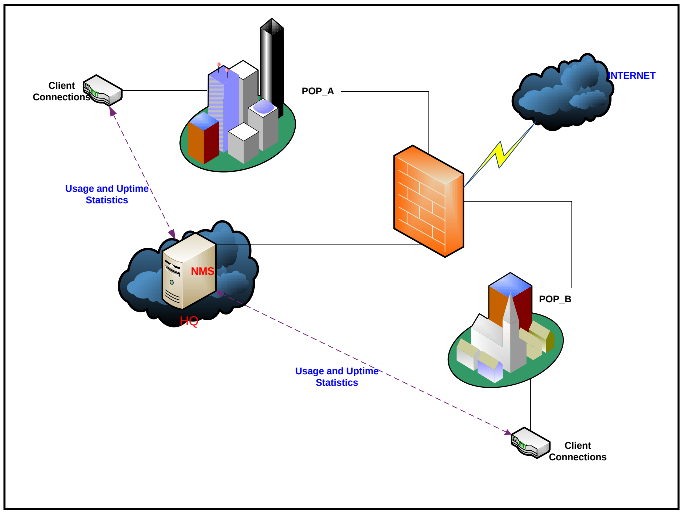


***Figure 1**: Architecture of A Typical Network monitoring System*

## V. ARCHITECTURE OF RESPONSIVE MONITORING SYSTEM

Our study takes into consideration certain assumptions to be able to design the desired model. These include: The service providers have more than one Point of Presence (POP) in the country and are logically interconnected; Internet and inter branch connectivity is provided to the users of the network using different technologies like Static routing, MPLS VPN, GRE etc; There is an existing Network Operations Center (NoC) for Network Monitoring and management; The available Internet Protocol (IP) block assigned to the Service Provider is submitted to the different POPs appropriately; There is an existing Database of Corporate clients with the required parameters to initiate monitoring; There is an existing monitoring tool used by the service providers which can be used in parallel to the Agent Based system if need be for evaluation.

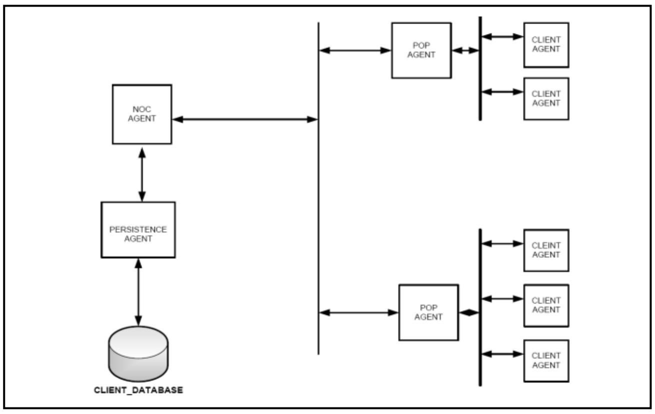


***Fig. 2 Architecture of the Responsive Agent Based System***

## VI. AGENT ROLES

The platform is a multi-agent based system in which agents interact to perform network monitoring. They are: **Persistence Agent** which handles database operations for the platform; **Platform Monitor Agent** which initializes the platform and the agents. It creates a link to the platform database and it ensures that all the required parts of the platform are running, these may include database and all the required agents; **NOC Agent** , This agent handles new sign ups and updates to and deregistration from the platform by external system. It also receives propagated results from the POP agents; **POP Agent**, This agent receives information about new clients and updates from the NOC agent. It receives the usage and uptime information from the clients and propagates the results to the NOC Agent; **Client Agents**, These agents are at the client end and are responsible for communicating usage and uptime statistics to the relevant POP agent; **RMA, DF and AMS agents**, These are system agents, shipped with the JADE distribution; JADE implements a Directory Facilitator (DF) agent as specified by FIPA [10]. The DF is often compared to the "Yellow Pages" phone book. Agents wishing to advertize their services register with the DF. Each platform must have a **Main Container** which holds two specialized agents called the AMS agent and the DF agent. The **AMS** (Agent Management System) agent is the authority in the platform. It is the only agent that can create and kill other agents, kill containers, and shut down the platform. The **DF** (Directory Facilitator) agent implements a yellow pages service which advertises the services of agents in the platform so other agents requiring those services can find them and; **Sniffer Agent** , It's a tool to sniff message exchange between agents. This tool is useful to debug a conversation between agents. It allows also saving the conversation to a file and load from a file.

## VII. SYSTEM COMPONENTS

**MySQL;** This is an open-source relational database management system (RDBMS). The SQL phrase stands for Structured Query Language. MySQL provides a suite of tools for developing and managing MySQL-based business critical applications on Windows [16].

Data is stored in separate tables rather than putting all the data in one big store. This adds speed and flexibility. The tables are linked by defined relations making it possible to combine data from several tables on request. The reason of choosing MySQL is that MySQL is very fast, reliable, and easy to use. MySQL also has a very practical set of features developed in very close cooperation with the users. The most important, it is open source and free of charge. Open source means that it is possible for anyone to use and modify. Anybody can download MySQL from the Internet and use it without purchasing the soft ware. Also, any one can study the source code and change it to fit their needs. Cross Platform compatibility of MySQL is another advantage. It can be installed in all major Operating Systems as UNIX, Solaris, LINUX in addition to Windows without a loss of performance [6]. It also runs without compromise with various development interfaces such as JDBC, ODBC, Pearl, Python, PHP, C++ and many others. This is primarily due to the development APIs that are integrated with it. MySQL also prevents memory leaks thus enabling efficient solutions for information storage. A single database can hold up to 8 Tera Byte of data, although the default limit is 4 GB.

**SQLyog Community;** It provides a powerful means to manage MySQL databases. It Runs on all Windows version from Win XP to Win 8.x (desktop systems) as well as "Windows Server" systems of same generations (Windows Server 2008) [15]. It allows Create/Drop/Alter Tables, Stored Procedures, Functions, Views, Triggers and Events.

The SQLyog Community Edition and is available as an open source project.

**Netbeans IDE;** This is an integrated development environment (IDE) for developing primarily with Java, but also with other languages, in particular PHP, C/C++, and HTML5. It is also an application platform framework for Java desktop applications and others. The NetBeans IDE is written in Java and can run on Windows, OS X, Linux, Solaris and other platforms supporting a compatible JVM.

We use Netbeans in this research study because; NetBeans offers best support for latest Java technologies; It's easy and efficient Project management software; NetBeans offers fast and smart Code editing much more than a text editor; Ability to write bug free code as NetBeans provides static analysis tools, especially integration with the widely used Find Bugs tool, for identifying and fixing common problems in Java code; Provides rapid User Interface Development; Design GUIs for Java SE, HTML5, Java EE, PHP, C/C++, and Java ME applications quickly and smoothly by using editors and drag-and-drop tools in the IDE; Support for multiple languages; Cross platform support; Rich set of community provided plug-in.

## VIII. AGENT INTERACTIONS

The transfer of usage and uptime statistics data from the various Points of Presence to the Network Operations Center requires a server to support large bandwidth requirements. The growth of networks complicates the effort to allocate server and network resources to ensure continuous storage of data over a long period of time [14]. Our plan is to have data storage server separate from the database so as to ensure fast and effective functionality of the system. The system also eliminates transmission of information and Data from the NOC to the client hence limiting communication to the NOC to be done by the POPs, this ensures all clients only communicate to the relevant POPs eliminating transfer of data directly to the NOC by the client.

The diagram below illustrates how the Agents interact using our model to accomplish the required tasks.

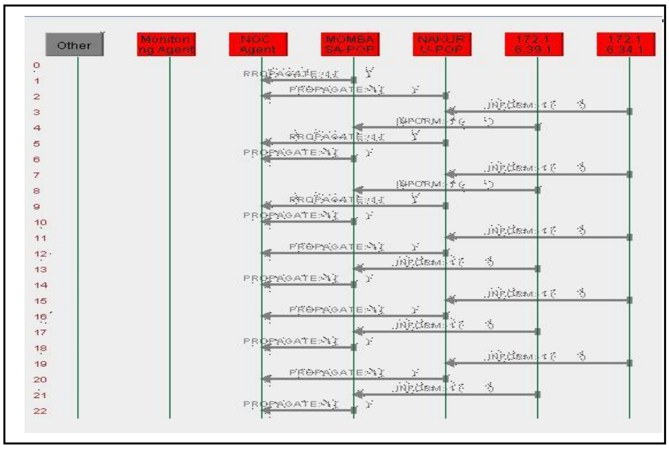


***Fig 3. Agent Interactions***

## IX. EXPERIMENTAL RESULTS ON CLIENT SESSIONS

Our Model took couple of scenarios to experiment the model; given a requirement to initiate a new monitoring for a newly subscribed corporate client, we are able to measure the average time taken to complete the exercise. This experiment was done with different network administrators at different intervals and the time taken to complete noted. The time was also noted for making changes to the existing clients and noting the time to complete.

The below table indicates tabulated results from the model.

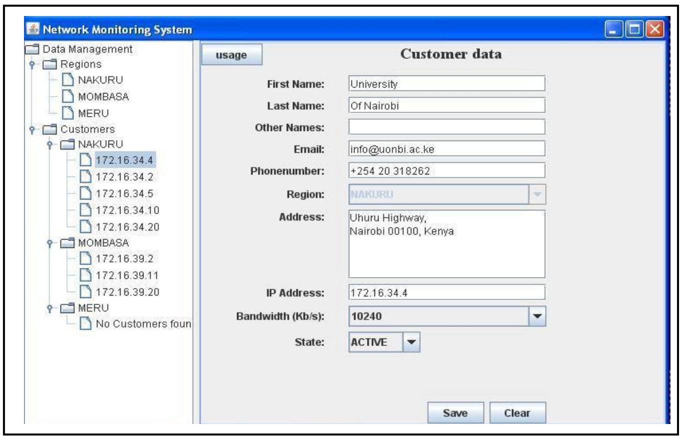
| Admin | Number of sessions | Time taken to Initiate | Time taken to Modify | Av.Time to Initiate (mins) | Av.Time to Modify (mins) |
|---|---|---|---|---|---|
| 1 | 5 | [4.5,4,3,2.5,2] | [2,2,1.5,1.5,1] | 3.2 | 1.6 |
| 2 | 6 | [4,3.5,3,3,2,2] | [3,3,2,2,1.5,1.5] | 2.9 | 2.1 |
| 3 | 8 | [4,4,4,3.6,3.5,3,2.5,2.5] | [2,2,2,2,1.5,1,1,1] | 3.4 | 1.6 |

**Table: 1 Experimental results from the MAS Model**

In this experiment, we get new client monitoring parameters and initiate monitoring from the Graphical User Interface (GUI) of the network monitoring page. We find that the time taken to either initiate or modify is affected by the number of times the admin uses the system. The trend is seen that the initial set up takes more time but as the admin get more conversant with the system, less time is taken to perform the function and then remains constant for a longer time. We believe that if the networks grow, the NOC agents can be increased and the query time can be shorten even further. However, the delay is still acceptable in this experiment. It can be shown that the use of agent paradigm allows us to increase responsiveness and flexibility and hence of scaling up the system.

## X. INITIATING MONITORING FOR A NEW CORPORATE CLIENT

The below results show how a newly subscribed client details are keyed into the database via a web console and the agents pick up the relevant details. Figure 4 below is an illustration of how the end user interacts with the system while initiating monitoring for a specific node in the network.



Customer data

First Name: University
Last Name: Of Nairobi
Other Names:
Email: info@uonbi.ac.ke
Phonenumber: +254 20 318262
Region: NAKURU
Address: Uhuru Highway,
Nairobi 00100, Kenya
IP Address: 172.16.34.4
Bandwidth (Kb/s): 10240
State: ACTIVE

***Fig. 4 Initiating Monitoring using the MAS model***

## XI. CONCLUSION

In this paper, we have introduced an agent platform to perform network monitoring. We discussed the system architecture and the role of agents. We also implemented a simple model to demonstrate agents performing network monitoring. Experimental results show that the time taken to initiate and modify details for monitoring can be highly reduced by using agents. The results also show that our system increases responsiveness and flexibility to network monitoring keeping the results more accurate as Ad-hoc method is eliminated. The system also improves scalability as many agents can be introduced into the system as the network grows.

## REFERENCES

[1] Bohoris Christos (2003); Network performance management Using Mobile Software Agents; submitted for the Degree of Doctor of Philosophy from the University of Surrey.

[2] Bianco Philip, Lewis A Grace, Merson Paulo (2008); Service Level Agreements in Service-Oriented Architecture Environments; A report prepared for the SEI Administrative Agent; ESC/XPK; 5 Eglin Street; Hanscom; AFB, MA 01731-2100.

[3] Cisco Systems, Inc (2001); Cisco IOS IP Configuration Guide, release 12.2.

[4] Cisco Systems, Inc(2012); Ciscoworks 2000 Installation and Configuration Guide.

[5] Dr. K L Shunmuganathan (2013); Multi-Agent

Based Network Monitoring and Management using

Jade.

[6] Ivonne Andrade Herrera (2012); Proactive Safety Performance Indicators, ISBN 978-82-471-3589-1.

[7] Giuseppe Aceto, Alessio Botta (2013); A survey of cloud Monitoring.

[8] Charles M. Macal and Michael J. North (2006); Center for Complex Adaptive Agent Systems Simulation (CAS2): Introduction to Agent-based Modeling and Simulation.

[9] Rao Sathya (2008); Project Coordinator MOMENT - Monitoring and Measurement in the Next generation Technologies.

[10] Fabio Bellifemine, Giovanni Caire (2010); JADE Programmer's GUIDE.

[11] D.Gavalas (2001); Mobile Agents for Network Monitoring and performance management.

[12] Wooldridge Michael (2002); An Introduction to Multi Agent Systems published by John Willey & Sons 2002.

[13] G. Aceto et al (2013); Cloud monitoring: A survey, Computer Networks.

[14] Mary Queen (2007); Specifying Protocols for Multi-Agent Systems Interaction.

[15] Microsoft (2008); Guide to installing Microsoft Windows Server 2008.

[16] Microsoft (2008); Guide to installing Microsoft SQL Server 2008.

[17] SolarWinds, Inc. (2011); Orion Simple Network Performance Monitor (SNMP) Administrator Guide, Version 10.2.1.

[18] Stefan Wallin (2012); Rethinking Network Management Models, Data-Mining and Self-Learning, ISBN 978-91-7439-390-3.