Management Model for Measurement Infrastructure

Gerson Battisti[1], Liane Tarouco[2], Lisandro Granville[2]

[1]Department of Technology - Regional University of the Northwest of RS (UNIJUI)
CEP 98.700-000 – Ijuí – RS – Brazil

[2]Institute of Informatics - Federal University of Rio Grande do Sul (UFRGS)
CEP 91.501-970 – Porto Alegre – RS – Brazil

{gerson, granville}@inf.ufrgs.br, liane@penta.ufrgs.br

ABSTRACT

The evaluation of metric of performance is useful in the evaluation of protocols, in the improvement of applications, in the content choice, among others. The obtaining of these metrics, it continues being a complex task because most of the time there are not appropriate tools for such. The measurement infrastructures appear as a form of endowing the networks of computers with instruments that facilitate the acquisition of these metrics. This work evaluates a group of measurement infrastructures, where it is possible to observe that the treatment showed to the management of such infrastructures is varied. They do not possess a group of standardized management functions. Based on that, this article presents a proposal of a management model for measurement infrastructures. The proposed model maintains the administrative independence of the present points in the infrastructure, but it allows the interaction among them by means of specific management functions.

Keywords: Measurement Infrastructure, Management Network;

1 INTRODUCTION

The performance evaluation of a flow is a difficult task when equipments belonging to different administrative domains influence in the performance. That difficulty happens because the access providers do not usually supply data on the load and performance of its networks. Also, because they do not possess appropriate tools or these procedures represents a security risk.

A solution is to endow the providers of a measurement infrastructure, adding a special hardware or software that enables to analyze the performance among those points. A measurement infrastructure permits to evaluate the performance and it can facilitate the quick detection and diagnosis of problems.

In spite of the existence of favorable points for the use of measurement infrastructures, they are still restricted to testing or small groups with common interests, because its use can become quite complex when done in wide scale.

Usually, the measurement infrastructures are not concerned with the conditions of the equipment in which it is installed. They do not monitor the hardware and software. The lack of monitoring of features such as: CPU use, available disk, execution status of test programs (normal, loop and fault) allows that the occurrence of errors compromises the work of the equipment and probably the results of the tests.

In a wide scale measurement infrastructure the changes in the installed tools or the installation of new tools are frequent. In both cases, a long time is necessary so that all the members are configured correctly and with the same software versions. Still, considering a measurement infrastructure, in wide scale, it will be inevitably present in multiple administrative domains, which means, the components of the infrastructure are heterogeneous and administered by different institutions. This situation involves security concerns among the administrators. To summarize, the measurement infrastructures do not possess a group of specific functions for the management of its components.

The main objective of this work is to propose a management model for measurement infrastructures. This model presents some functionality of network management to be used in a safe way, which facilitates the maintenance and use of the measurement infrastructures.

The remainder of the paper is organized as it follows: In section 2 a group of public measurement infrastructures is introduced. In section 3, it is presented the management model for measurement infrastructures. The components and the functions foreseen in the model are detailed in along of the section. The final considerations of the work are presented in section 4.

2 MEASUREMENT INFRASTRUCTURES

A measurement infrastructure allows to evaluate the network performance and it can facilitate the fast detection and diagnosis of several problems. The measurement infrastructure goals are [1]: to establish a pattern of behavior of the network, to differentiate and to detect abnormal behaviors; to detect and to locate specific problems in the Internet (DoS attacks, router configuration problems, link fault or hardware fault); to identify bottlenecks; to maintain the data collected by a long period (historical) for post analyses (tendencies); to facilitate the collection of information for specific events (an experiment). The evaluated measurement infrastructures were AMP, BW, MonALISA, NIMI, NWS, PingER, E2E piPES, RIPE TTM, Scriptroute and Surveyor.

2.1 AMP

AMP is NLANR's Active Measurement Project. The focus is on making site to site measurements of round trip time (RTT), packet loss, topology and throughput across the National Science Foundation (NSF) approved HPC networks. Each monitors sends a single ICMP packet to each of the others every minute and records the time to (or absence of) the reply. In addition, every 10 minutes the route to each other monitor is recorded using traceroute. Throughput tests can also be run between any pair of the monitors using a web based throughput test request. (Throughput tests are only run on demand because of the high cost, in terms of network traffic, of running these tests.) The following throughput tests are available: Bulk TCP data transfer, Bulk UDP data transfer, ping –F, treno [2].

2.2 IEPM - BW

The Internet End-to-End Performance Monitoring – Bandwidth goal was to develop a simple, robust infrastructure to enable making network and applications measurements for links capable of high throughput. There are 2 types of hosts [3]: a measurement or “monitoring host” that runs the measurement tools ("sensors"), logs the data from the sensors, extract, analyses and reports on the information via the web and the “remote host” that run the servers and responds to the probes from the monitoring host sensors.

The monitoring host starts a set of measurements at regular intervals driven by a Unix cron table. Typical intervals are of the order of an hour or two. The actual intervals depend on the load acceptable on the monitoring hosts link, and the amount of time it takes to make a set of measurements to all remote hosts. The actual start of the measurements is randomized with a flat

distribution over a 15 minute interval. For each set of measurements, the measurement host selects each remote host in turn and runs ping for 10 seconds, does a traceroute (with one measurement per hop) followed by running the iperf. TCP transfer tool, secure file copy using the peer-to-peer bbcp tool with both memory to memory (bbcpmem) and disk to disk copies (bbcpdisk), followed by the bbftp file transfer tool and the packet dispersion bandwidth estimator pipechar.

2.3 MonALISA

The MonaLISA (Monitoring Agents in A Large Integrated Services Architecture) system provides a distributed service for monitoring, control and global optimization of complex systems. MonALISA is based on a scalable Dynamic Distributed Services Architecture (DDSA) implemented using Java / JINI and Web Services technologies. The scalability of the system derives from the use of a multithreaded execution engine to host a variety of loosely-coupled self-describing dynamic services or agents, and the ability of each service to register itself and then to be discovered and used by other services, or clients that require such information.

The system monitors and tracks site computing farms and network links, routers and switches using SNMP, and it dynamically loads modules that make it capable of interfacing existing monitoring applications and tools. The core of the monitoring service is based on a multithreaded system used to perform the many data collection tasks in parallel, independently. The modules used for collecting different sets of information, or interfacing with other monitoring tools, are dynamically loaded and executed in independent threads [4].

2.4 NIMI

NIMI (National Internet Measurement Infrastructure) is a software system for building network measurement infrastructures. A NIMI infrastructure consists of a set of measurement servers (termed NIMI “probes” or “platforms”) running on a number of hosts in a network, and measurement configuration and control software, which runs on separate hosts. NIMI is not a measurement tool, but a command and control system for managing measurement tools.
The design of NIMI emphasizes are [5]:
(i) Infrastructures composed from diversely-administered hosts, rather than an infrastructure controlled by a single entity, and
(ii) Facilitating diverse types of measurements by diverse parties, some of whom are allowed richer access to certain portions of the infrastructure than others.

2.5 NWS

The goal of the Network Weather Service is to provide accurate forecasts of dynamically changing performance characteristics from a distributed set of metacomputing resources. The NWS was designed and implemented a system that takes periodic measurements of the currently deliverable performance (in the presence of contention) from each resource and uses numerical models to generate forecasts of future performance levels dynamically. Forecast data is continually updated and distributed so that resource allocation and scheduling decisions may be made at run time based on expected levels of deliverable performance. To the extent that network performance conditions can be thought of as the “network weather” this functionality is roughly analogous to weather forecasting and hence we term the system the Network Weather Service (NWS) [6].

2.6 IEPM – PingER

As its name indicates, the framework of the PingER project is based on the ping program familiar to network administrators and widely used for network troubleshooting. A ping involves sending an Internet Control Messages Protocol (ICMP) echo request to a specified remote node which responds with an ICMP echo reply. It is also optional to send a data payload in the request which will be

returned in the reply. The round-trip time (RTT) is reported; if multiple pings are dispatched, most implementations provide statistical summaries [7].

The PingER methodology is sends 11 pings with a 100-byte payload, at 1 s intervals, followed by 10 pings with a 1000-byte payload, also at 1 s intervals, to each of a set of specified remote nodes listed in a configuration file. The first ping is discarded because it is assumed that it is slow due to priming caches.

2.7 E2E piPES

The main objective of the piPEs project is to enable end-users and network operators to determine end-to-end (E2E) performance capabilities, locate E2E problems, and contact the right person (with evidence) to get an E2E problem resolved. For example, an end-user who seeks to determine whether an E2E performance problem exists, identify the appropriate person to contact to get the problem resolved, and supply sufficient documentation to convince the remote network engineer that a problem within their sphere of responsibility does, indeed, exist [8].

The piPEs architecture considers the E2E path to be composed of a series of partial paths composed of layer 3 network components (e.g. end-hosts and routers). A remote machine can then initiate regularly scheduled or on-demand tests between any two such network measurement node (NMN), subject to authorization and policy restrictions. The results of such tests are stored in a database of performance results, and a remote machine can request such results from said database. The currently tools in pipes framework is: BWCTL, OWAMP, traceroute, and NDT.

2.8 RIPE NCC - TTM

The Test Traffic Measurement is a project in RIPE NCC (Reseaux IP Europeen - Network Coordination Center). The main goal of the TTM is to do performance measurement according to well-defined and scientifically defendable standards set forth by standards bodies such as the IETF [9]. The metrics used in TTM is one-way delay and one-way packet loss. The measurement device should be a black box in order to avoid tampering with the device by sites using it. The measurement device was called “test-box”. The test-box is equipped with a GPS card for clock synchronization.

2.9 SCRIPTROUTE

The SCRIPTROUTE goal is to combine the flexibility to run a wide variety of different measurement tools with the general availability of traceroute servers. We begin with the safety properties of traceroute servers: we design the system to prevent misuse, even at the cost of disallowing some kinds of useful measurements. Our thesis is that even within the context of a carefully controlled interface, we can provide more functionality than is currently provided by traceroute servers [10].

To use Scriptroute, clients use DNS to discover measurement servers and then submit a measurement script for execution in a sandboxed, resource-limited environment. The servers ensure that the script does not expose the network to attack by applying source- and destination-specific filters and security checks, and by rate-limiting traffic.

2.10 SURVEYOR

The Surveyor project consistently measures end-to-end unidirectional delay, loss, and routing among a diverse set of measurement probes throughout the Internet. The goal of the project is to create technology and infrastructure to allow users and service providers (at all levels) to have an accurate common understanding of the performance and reliability of paths through the Internet. Surveyor measures the one-way delay and one-way loss] metrics being developed by the Internet Protocol Performance Metrics (IPPM) working group of the Internet Engineering Task Force (IETF) [11].

Each Surveyor measurement machine is a desktop PC. In addition to the basic PC, each machine has an appropriate network interface card, and a GPS card.

2.11 Measurement Infrastructure Comparative

This section shows a comparison among measurement infrastructures that has public information MonALISA, piPES, NIMI, Surveyor, Scriptroute, PingER, BW, AMP, NWS and RIPE.

The measurement infrastructure can be classifying in three test types groups: based on round trip delay (rtt), based in one-way delay and dynamics (tests can be built dynamic). The infrastructures that use tests based in round trip delay they are PingER, BW, AMP and NWS. The tests used by these infrastructures use the ping program or variations. A ping is the send of a package *ICMP echo request* for remote equipment that should respond with a package *ICMP echo reply*. Additional bits (payload) can be sent in the package. The packet round trip time (rtt) it measured.

The infrastructures Surveyor, RIPE and piPES are based on one-way delay and use the OWAMP protocol. In this case it synchronized clocks is requires. The infrastructures that permit inclusion of dynamic tests are MonALISA, NIMI and Scriptroute. The inclusion of a test can be in the scripts form or as a tool that is executed by the infrastructure. Tests inconsistency can cause problems for these infrastructures.

For one-way measures, the equipments clocks synchronization is a decisive factor. To maintain synchronized clocks the infrastructures Surveyor and RIPE using external equipment such as GPS. The other infrastructures use the NTP protocol. The argument for not using GPS is installation complexity (software, antennas and cost) and no need of high accuracy. In addition, these measures just will be used to indicate trends or fault detect.

The most important point for this work is the features of management existent in the infrastructures. Some proposals such as AMP and PingER simply ignore the management need and just define a minimum group of hardware and software to compose the infrastructure. The infrastructure that more includes management foresees a small group of functions that are usually used for users and test tools control. Same management functions found in infrastructure are: security connections, automatic updates, user access control list, program access control list, status test monitoring, memory and processor monitoring, disk utilization limits and automatic restart service.

3 MEASUREMENT INFRASTRUCTURES MANAGEMENT

The efficient use of wide scale measurement infrastructures is a objective difficult to reach. Some proposals already anticipated the problems and they have informed that, in the way the infrastructure is being proposed it is not scalable [2]. The biggest problem found in those proposals is the management of the components in different administrative domains. The complexity already begins in the inclusion of a new element in the infrastructure, being necessary to establish a group of procedures and authorizations among administrators.

3.1 Management Requirements

The management requirements are presented in agreement which as the functional areas foreseen in the OSI Management Model, which are still accepted for any management model.

3.1.1 Fault Management

The Fault Management has the responsibility for monitoring the status of the resources, for the maintenance of each one of the objects and for the decisions that should be taken to reestablish the units of the system that can bring problems. The components that should be monitored are:

Hardware: All hardware components should be monitored (CPU, disks and network interface). In the case of the network interface one of the objectives is to avoid that an interface which is working with problems can compromise the results of the tests.
Operating System: There is a report of experiments of infrastructures which shows simple problems as out of disk space compromise the operation of the tests [12]. In addition, it is necessary to monitor the applications of the operating system that are necessary for the measurement infrastructure, such as, daemon ssh, database, cron and others.
Test Tools: A measurement infrastructure has a group of applications that is responsible for the tests. For example, the one-way delay test using the protocol OWAMP needs interaction between the points of measurement. That interaction is made based on the client-server model therefore the monitoring of the execution of server application is essential to guarantee the operation of the tests.

3.1.2 Configuration Management

The functions of the configuration management allow to collect, to identify and to control information of configuration of the managed objects. In this area it is necessary to monitor the following points:
Test Tools: The management of this component should register and verify the configuration of the tools. The registration is important to generate "configuration disks" which store the custom configurations, facilitating new installations or recovery in case of fault. Another important task of the configuration management is tool updating. For example, the administrator can define that the updating will be manual and it just wants to be notified when a new version is available.
Operating System: The main task of the configuration management in the operating system is to register and to store the configuration of the necessary components for the operation of the infrastructure. For example, the measurement infrastructures usually use the openssh as a security component. The registration and storage of its version and configuration can avoid compatibility problems.

3.1.3 Account Management

The functions of the account management are measure and collect information regarding the use of the resources and services of a network. The knowledge of the use of the resources is necessary network capacity planning. In this area the following factors can be accounted:
Users: The accountancy of the users' activities can avoid abuses in the use of the measurement infrastructure. Some bandwidth tests generate a lot of traffic in the network and a user can use this mechanism as a denial of service (DoS) attack, trying to congest the connection to the measurement point.
Test Tools: The monitoring of the amount of request of each test, associated to the consumption of resources for each one of them is important for the capacity planning of the measurement point.

3.1.4 Performance Management

The performance management is the group of functions responsible for the maintenance and analysis of the performance registrations. The objective is to be used in the analysis of tendencies and in the consequent system planning. The components that should be evaluated are:
Test Tools: The monitoring of a certain test execution implies in evaluating the time of execution, the amount of used memory, processor and amount of generated traffic. This monitoring seeks to identify abnormal behaviors of the tools, for example, programming loops or other flaws. Another important factor is to establish a relationship between the hardware consumed by the tests and the existing hardware. The administrator needs this relationship to be notified to surpass predetermined values. This management allows to evaluate and to define which are and which number of simultaneous tests that can be made.

3.1.5 Security Management

The functions of security management are to give subsidies to the application of security polices, protecting the managed objects and the system of inappropriate accesses. Security management in the measurement infrastructures can be applied in:

Users: In a measurement infrastructure there are different tools of tests, with different functionalities and different requirements. It is necessary to manage the relationship between users and test tools. The tools of evaluation of band width, for example, generate a lot of traffic in the network and, therefore, its use should be rational. It would be normal to believe that these tools have its restricted use for the operator.

Test Tools: One of the administrators concerns is the possibility attacking of the equipment used for the measurement infrastructure and the after use to attack the other equipments. The monitoring of the permissions and of the identification of the test tools seeks to detect a possible abnormal operation of the program.

Data: The results of the tests are stored and they can be published in different levels, in agreement with the users. For example, the data measurement brutes are of restricted access and consolidated data are of public access. It is a task for the operator to identify and to define these differences.

3.2 Management Model for Measurement Infrastructure

An infrastructure of measurement of global scope is distributed unavoidably between different administrative domains and the management model it should attend to this feature. The model considers that each domain administrative participant of the measurement infrastructure should turn available at least an element to participate in the tests. This element will be identified now in as **Measurement Point**. A Measurement Point is constituted by hardware (dedicated or not) and a group of software's that it permit: to elaborate acting tests among the participants of the infrastructure; store and to turn available the results of the tests. Each administrative domain should possess a manager that answers for the domain and that will make interaction with the managers of the other domains. The relationships among the managers are presented in the sequence.

3.2.1 Managers Interaction

The model presumes the existence of at least a manager in each administrative domain that integrates the measurement infrastructure. From now on, the responsible local manager for the domain is called **Domain Manager (DM)**. A DM interacts with managers of other domains and with a top-level manager called **Infrastructure Manager (IM)**.

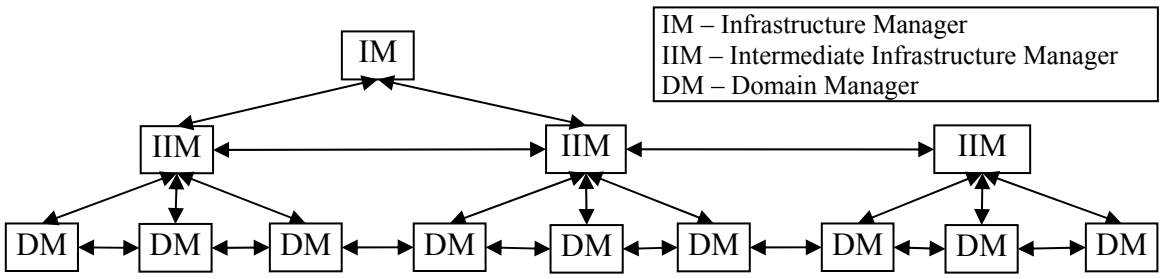

Figure 1: Relation among managers

Considering a measurement infrastructure in global scale, the centralization in a single manager is possible, but not advisable due to the scalability problems. To guarantee the scalability, the interaction among managers is recursive, which means, an IM can interact with a new top-level

manager, becoming a mid-level manager, called **Intermediate Infrastructure Manager (IIM)** and so forth.

The relations among managers are presented in figure 1. The managers are organized in hierarchical form where, in base has DMs that communicate with other managers in same level and with managers of superior level. In theory, the number of DMs, IIMs e IMs is unlimited. The communication among managers should guarantee the domain administrative independence. Each domain operator determines which interactions are possible with its equipment.

3.3 Model Components

As presented in the previous section, the model of proposed management is distributed and its components are: **Infrastructure Manager (IM)**, **Intermediate Infrastructure Manager (IIM)** and **Domain Manager (DM)**. In a new detail level, the model presents the components: **Domain Agent (DA)** and **Infrastructure Agent (IA)**. The figure 2 presents the components of the model and to its interactions. The numbers in figure are used in sequence to detail components and its interaction.

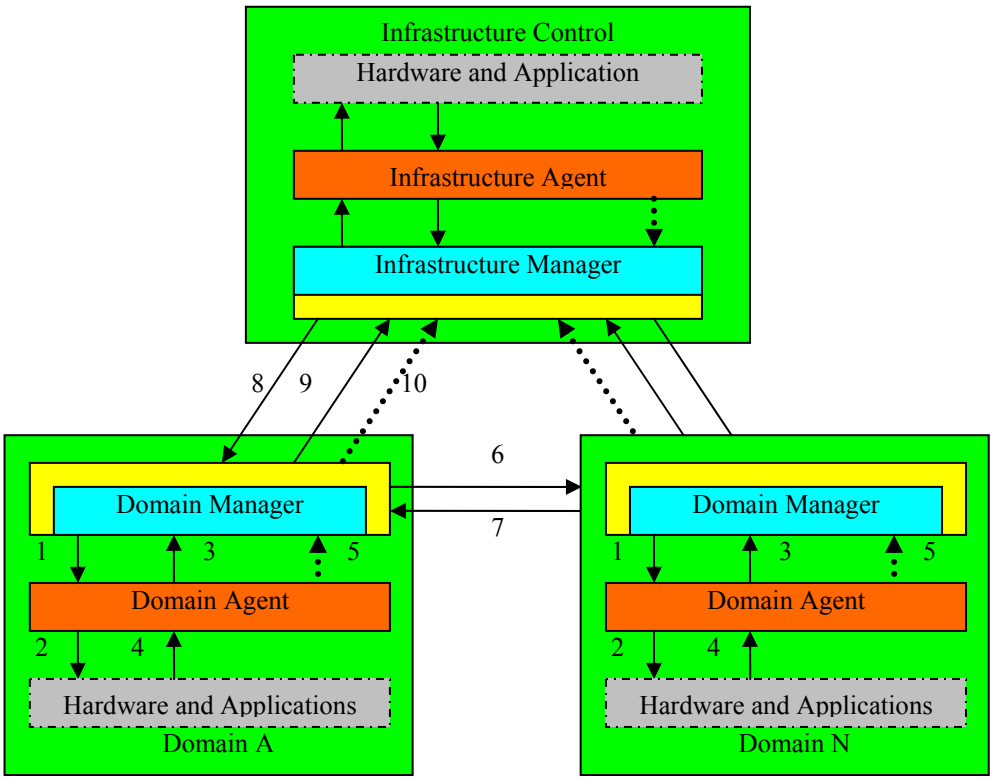

Figure 2 – Management Model for Measurement Infrastructures

3.3.1 Infrastructure Manager

The IM is the responsible for the management of the all measurement infrastructure through IIM or DM. The main functions are: keep consolidated information about infrastructure management; to keep information on the active members; monitor its local components (for sample, databases and test tools); notify IIM or DM about components updates. As presented in the figure 1, an IM can interact with others IMs, IIMs and DMs. An IM or an IIM are extensions of DM and any DM can become an IM or IIM.

3.3.2 Domain Manager

The DM is responsible for the management tasks in its administrative domain. The DM consults its agents periodically and takes actions, if necessary, to guarantee the good operation of the measurement point. The DM interact with the agents of its domain in agreement with the centralized management model, i.e., the manager is responsible for the consults (1) to all the DAs and receives answers (3), besides that, it receives all notifications of the DA (5). The DMs tasks are not restricted to its domain. It also interacts with managers of other domains (6)(7) and with the IMs (8)(9). Interaction among managers of different domains allow for information exchanging without the need of an IM intervention (which is also possible).

When a DM interacts with the IM, it functions as an agent, called IIM. As IIM, it should answer the IM consults (9) and receives information about alterations in the infrastructure components such as updating or changing in the configuration. When the DM functions as an agent, it should notify the superior manager in cases of critical information, even if that has not been requested (10). Whenever a DM is initiated it uses the same mechanism (10) to register in the IM and consequently to integrate the infrastructure.

3.3.3 Domain Agent

The DA is responsible for the monitoring of all necessary components for the operation of the measurement point. These components include the specific applications of the measurement point, the operating system tools and the hardware. The agent should interact with the applications and the hardware of the measurement point (2)(4) to answer the requests of the manager(3). Besides that, the agent should notify the DM (5) about abnormal behaviors of the monitored components. Each administrative domain can integrate a measurement infrastructure with more than a measurement point and in this case to possess more than one DA.

3.3.4 Infrastructure Agent

An IA has the same features of a DA, including the interaction with the IM. Also, the infrastructure agent should monitor the specific components found in the measurement infrastructure control. It is considered specific components of the control as applications to receive, to consolidate and to keep the measurement data.

4 FINAL CONSIDERATIONS

It is known that a measurement infrastructure is useful for a better understanding of the behavior of the network. With a measurement infrastructure it is possible to build a network baseline and this can be used to detect anomalies. Moreover, the behavior historical registrations allow to evaluate tendencies, facilitating the network capacity planning. The inability of the current measurement infrastructures towards the management of its components restricts its use.

To tackle the existing management lack in the measurement infrastructures, it was presented a distributed and autonomous management model. Such model tries to minimize the main difficulty of the use of measurement infrastructures in different administrative domains. In this case, the model enables each local operator to administrate its part of the infrastructure and make functions available for the other managers. The functions allow an integrated and collaborative management of the whole measurement infrastructure.

The next stage of the work will be validating the model, to build a management prototype. The preliminary analyses on the methodology to be used in the building of the prototype indicate that the use of a Service Oriented Architecture (SOA) is appropriate to the needs of the model. A SOA is not a technology in itself, but a group of principles and methodology for the development of

"services" that can be used through the network. The services will be the functions presented in the Management Model for Measurement Infrastructure.

REFERENCES

[1] MURRAY, M. CLAFFY, K. **Measurement the Immeasurable:** Global Internet Measurement Infrastructure. Cooperative Association for Internet Data Analysis, CAIDA, San Diego Supercomputer Center, University of California, San Diego. Disponível em: <http://www.caida.org/outreach/papers/2001/MeasInfra/measurement.pdf> Acesso em: mar. 2005.

[2] McGREGOR, T. BRAUN, H-W. BROWN, J. The NLAMR network analysis infrastructure. **IEEE Communications Magazine**, New York, v. 38, n. 5, p. 122-128, maio 2000.

[3] IEPM. **BW:** Methodology. Disponível em: < http://www-iepm.slac.stanford.edu/bw/methodology.html>. Acesso em: Mar, 2005.

[4] TOARTA, M. et al. **MonALISA:** Monitoring Agents in A Large Integrated Services Architecture. Disponível em: <http://monalisa.cacr.caltech.edu/> Acesso em: fev. 2005.

[5] PAXSON, V.; MAHDAVI, J.; ADAMS, A.; MATHIS, M. An architecture for large scale Internet measurement. **IEEE Communications Magazine,** New York, v.38, n.8, p.48-54, Aug 1998.

[6] WOLSKI, R. Dynamically forecasting network performance using the Network Weather Service. **Cluster Computing**, Hingham, v. 1, n. 1, p. 119-132, mar.1998.

[7] MATTHEWS, W. COTTRELL, L. The PingER Project: Active Internet Performance Monitoring for the HENP Community. **IEEE Communications Magazine**, New York, v.38, n.5, p.130-136, maio 2000.

[8] E2EPI. **piPES:** Internet2 E2E Performance Initiative Performance Environment System. Disponível em: <http://e2epi.internet2.edu/E2EpiPEs/index.html> Acesso em: Out. 2004.

[9] GEORGATOS, F. at all. Providing Active Measurement as a Regular Service for ISP's. In: A WORKSHOP ON PASSIVE AND ACTIVE MEASUREMENTS, 2001, Amsterdam, April 2001. **Proceedings…** Disponível em: <http://www.ripe.net/projects/ttm/Documents/Papers/PAM2001.pdf>. Acesso em: Mar. 2005.

[10] SPRING, N. WETHERALL, D. ANDERSON, T. Scriptroute: A Public Internet Measurement Facility. USENIX SYMPOSIUM ON INTERNET TECHNOLOGIES AND SYSTEMS, 4., 2003, Seattle, Washington, USA. **Proceedings…** Disponível em: <http://www.usenix.org/events/usits03/tech/spring.html>. Acesso em: abr. 2005.

[11] KALIDINDI, S. ZEKAUSKAS, M. J. Surveyor: An Infrastructure for Internet Performance Measurements. In: INET Internet Network, 1999, San Jose, California, USA. **Proceedings…** Disponível em: <http://www.isoc.org/inet99/proceedings/4h/4h_2.htm>. Acesso em: abr. 2005.

[12] PAXSON, V.; ADAMS, A.; MATHIS, M. Experiences with NIMI. In: SYMPOSIUM APPLICATIONS AND THE INTERNET (SAINT) WORKSHOPS, 2002. Nara, Japan. **Proceeding…** p.108-118, Jan. 2002.