

Location Based Network Management System

Carlos André
Instituto Superior Técnico/UTL, Portugal
carlos.andre@ist.utl.pt

December 3, 2009

Abstract Today, companies increasingly rely on their computer networks to support their business operations, but in many cases the network infrastructure is large and complex. In order to achieve high performance, NMS (Network Management System) are installed to perform constant monitoring and surveillance of the network. These systems increasingly use the georeferenced location of network elements in order to add new management capabilities, thus resulting in another type of systems, the LBNMS (Locations Based Network Management System). This concept was explored in this thesis to develop a georeferenced NMS based on the integration of the Nagios system and Google Earth. The developed NMS was applied to the monitoring of IP networks with fixed equipment, and a P2P network over MANET (Mobile Ad-hoc Network).

Key words: LBNMS, Nagios, Google Earth, KML, GWT, P2P over MANET, SNMP

1 Introduction

In the current scenario, modern computer networks are large heterogeneous collections of computers, switches, routers and other devices. The growth of such networks tends to be ad-hoc and based on the current and perceived future need of the users. In larger networks, the job of monitoring and managing becomes a complex task. Without any tools to deal with this complexity, network administrators cannot ensure the desired levels of performance and availability offered by the network. So, it is essential

that companies invest in Network Management System (NMS) solutions, providing network administrators with tools to analyze, identify and solve problems efficiently. These solutions are becoming more important, because companies increasingly depend on computer networks to support their business operations. Today, a network downtime or performance degradation can result in large financial losses.

For these reasons, extensive research on the area of network management has been made in order to develop new tools and standards (e.g. SNMP [1] and CMIP [2]). These works and standards contribute to a more simplified management of the entire network infrastructure, covering scenarios from multiple areas (performance, fault, configuration, accounting and security) on the OSI network management model.

Over time, the NMS tools have undergone a process of specialization that seeks to deal with various types of network, such as WAN (Wide Area Network), WLAN (Wireless Local Area Network) and Mesh. This approach, allows NMS specialization, providing more appropriate features. A common trend in this tools, is the inclusion of the georeferenced information on the network elements, providing new management capabilities, and creating NMS based on location which are the core concept of the study developed in this thesis.

1.1 Motivation

The traditional NMS typically use network diagrams without any geographical information from their equipments, allowing only a topological view of the

network and how the equipments are connected together. Therefore, we cannot know in detail where such equipment is located, for example, in which room, building or street.

With the addition of geographical information, the network analysis is improved, since it gets a clear view of how the connections between the equipments are physically made (see picture 1). With this approach, it is possible to identify the dispersion of the network based on geographic distances and to improve the analysis of congestion problems and faults occurring in equipment and services. With the information shown in figure 1, for example, the administrator can check different points of the network, and observe a great activity in the building of router1, with yellow links between the AP and the router. From this geographical information, like the floors, GPS coordinates and some statistical data, the administrator may choose adding another AP to a specific building area.

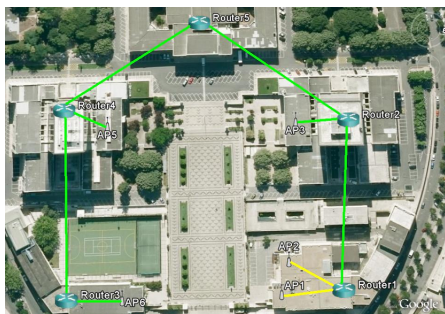


Figure 1: Example of a map with device locations and their connections

With this location concept, it is possible to provide a great flexibility on network maintenance, such as distribution of tasks to repair/assistance carried out by maintenance teams, based on geographical location(s) and the failure location. Some examples could be found at some Telecommunications Operator, when some BSC (Base Station Controller), DSLAM (Digital Subscriber Line Access Multiplexer) or router fail. Maintenance teams having a PDA running a NMS module are notified to move to the failure location, because they are the nearest or only avail-

able technical in that zone.

There are also advantages for network planning and design by combining the physical location of equipments and their performance statistics, helping to make better planning decisions. These are useful features, for outdoor environments, like celular, mesh and fixed networks. For the wireless networks, we can apply theoretical models of signal propagation and check the radio coverage in geographical maps. For complex models it is possible to use buildings and objects existent in that zone. For fixed networks, details of cabling can be represented in the map, like the real path taken by the fiber/copper through the cities and villages.

Another application, can be found in Ad-Hoc networks, updating in real-time the nodes location in the map. Ad-Hoc networks are very volatile, with node addition and deletion at any instance, making its monitoring a challenge. This type of network can be used in critical scenarios like rescue or even social environments. In this thesis, a P2P network over MANET (Mobile Ad-Hoc Network) was monitored.

1.2 Objectives

The goal of this thesis is the development of a location based NMS that uses the physical location of the network elements to provide additional information to manage networks. To accomplish this goal, the NMS will use Google Earth as a graphical interface, taking advantage of its technological potential, as the visualization layers, coverage of the globe with high image quality and also all features related to 2D/3D design.

The solution must be as possible agnostic to the network, regardless of its scale and settings. It is important to emphasize that the NMS will not be created from scratch, by selecting one with great popularity, online documentation and integration flexibility for Google Earth, easing the transition and conversion of data between these two worlds, i.e., NMS data to Google Earth. The NMS that fulfill these requirements was the open source Nagios [3].

The developed NMS was tested using the network data of IST (Instituto Superior Técnico) and a network mesh of 6 netbooks (Asus EeePC). The first

environment was used to monitor a general fixed IP network, and the other environment to monitor a P2P network over MANET [4].

1.3 Organization

This thesis is structured in five chapters. The second chapter presents the architecture solution, describes its main requirements and components description. The third chapter contains the implementation details. Chapter four describes the performed tests to validate the developed solution. Finally, the last chapter presents conclusions and future work.

2 Architecture Solution

This chapter describes the main requirements and the proposed architecture to implement the LBNMS.

2.1 Requirements

At this point the essential requirements that must be included in the solution, are the following:

- Embedding Google Earth on the graphical environment of the NMS
- Ability to add/edit/remove equipment and network links in Google Earth
- Creating visualization layers to filter information
- Add graphics generated by the NMS into Google Earth
- View the status of equipment and links in Google Earth, according to the color of its icon
- Persistent storage of geographic information
- Concurrency control between users and ensure the data consistency
- SNMP agent for monitoring the P2P network over MANET

2.2 Architecture

The overall architecture is presented in figure 2.

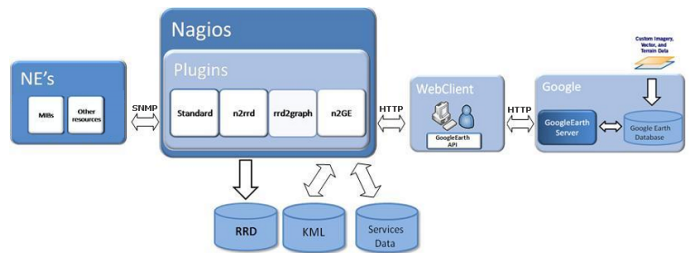


Figure 2: Architecture

2.3 Components

The architecture consists in 4 main components: network elements; NMS Nagios; Web Clients; Google Earth service.

2.3.1 Network Elements

The monitored network elements are desktops, laptops (Asus EeePC), switches, routers and APs. The management protocol used was the SNMP, usually the version 2c, to collect status informations from MIB tables. Other protocols were used too, like ICMP (Internet Control Message Protocol).

2.3.2 Nagios

The Nagios was the chosen NMS to build the geo-referenced system. This open source application includes a large number of plugins and add-ons, with many potential to monitor the resources on the network. The work presented in this thesis corresponds to an additional Nagios plugin, providing location aware behavior.

Nagios is a very complete tool in the alarmistic field, but doesn't provide native mechanisms for graph generation, unlike other monitoring tools, namely Cacti [5] and Zabbix [6]. To overcome this limitation, the plugins n2rrd and rr2graph are used to reproduce the graphics, based on the collected data from network elements when Nagios does its periodic pollings.

The plugin n2rrd is responsible to store the collected data from the network elements into a RRD database,

while the plugin `rrd2graph` is responsible to generate graphs from it.

The add-on N2GE, corresponds to a web service which provide the integration between the world of Nagios and Google Earth, by processing data from Nagios, typically images of statistical graphics and other information to monitor, and convert this into the KML format. When these KML file is loaded by a network administrator on a web browser featuring the Google Earth plugin API, all this information will be interpreted and displayed.

2.3.3 Web Client

This component corresponds to the network administrators, carrying out the access to NMS by a web browser. To achieve this access, it is required to have the Google Earth API plugin installed in order to view the Google Earth in the web browser. Currently, this plugin is only available for MacOSX and Windows platforms, which have support for most browsers (IE, Firefox, Safari and Chrome), and will be available for Linux in a near future.

This plugin will allow embbed Google Earth inside the Nagios administrative webpages, becoming unnecessary the usage of Google Earth as a standalone application.

2.3.4 Google Earth

This component is the service provided by Google, which offers a set of technologies for creation of GIS and LBS. Running the Google Earth plugin on the client side, the administrators can interact with this geographical platform, an export the information of the network state, both software and hardware to this environment and perform their monitoring.

There are many similar tools such as NASA World Wind, Microsoft Live Earth, but Google it's more mature, providing practically all the rivals features. Another advantage appears from the many operation system support, such as Linux, Mac and Windows.

3 Implementation

The N2GE is composed by a single webservice, which is responsible to implement the integration of Nagios and Google Earth. This webservice is accessed by a single url placed on the Nagios administrative webpages.

The N2GE was developed using the GWT framework on the Eclipse IDE. The GWT provided the tools to develop the client and server side code. For the client side, it generates Javascript and for the server side, a java servlet. The client side code, is loaded by web clients, and it includes the Google Earth interface and a set of HTML forms to export Nagios data to it. The server side code runs typically on the Nagios machine and it implements a set of methods to receive the RPC's invoked by web clients, such that they may perform some reads and write operations, resulting in accesses to database and KML files.

3.1 GUI Objects

To represent the network in the Google Earth graphical environment, four types of objects were defined: the network equipments, network links, cameras and building structures. These objects are sufficient, since we can design the network topology with the network equipments and their links, and additionally, using 3D buildings to better understand its location in indoor locations. The camera object allows to move the user viewpoint to places of interest in the map.

3.1.1 Network equipments

The network equipments were represented in the geographical map using Google Earth's placemarks. Each placemark represents a position on the earth's surface, with some icon and description in HTML. These features, enable to construct a rich object with many vital information about the current status of the equipment.

The icon image was used to represent the equipment's type. The supported icons in this work are listed on the table 1. Using different icons, it is pos-






Icon	Equipment type
	router
	switch
	access point
	server
	laptop

Table 1: Available icons

sible to better differentiate the equipment functions, such as access or core nodes.

The icon's color correspond to the equipment's status, which can be four (ok, warning, critical and unknown), as shown in table 2. This color is updated in real-time. For example, if a equipment or service becomes inactive, the icon becomes red, providing an intuitive way to view the network state.

These icons, when selected, can display an HTML page. This page, displays informations about the equipment, like its geographical information, monitoring graphics, IP address and the equipment status (up or down).

The included geographical information are referenced to equipments floor, and must be provided by the administrators.

For the monitoring graphics, each graphic has associated status, that are obtained by Nagios plugins, when collecting network information. The Nagios plugins are configured with some thresholds, that allows determine the state of each collected sample.

3.1.2 Network Link

The network link between the equipments were represented using 2D lines. A line is created with two GPS coordinates of both network elements. This line only shows the link status, with a different color for each network status.

To determine a status line, it is required to known the NICs (Network Interface Card) status, which can

be obtained by Nagios.

3.1.3 Camera

The camera provides a way to define the observer's viewpoint in terms of position and viewing direction, only requiring 2D coordinates, longitude and latitude. The third coordinate (altitude), is not available in the Google Earth API, so it is reset to the default value.

This object is important to save all relevant locations in the globe, becoming easy and fast to look at any place, avoiding manual navigation to the desired location.

3.1.4 3D Buildings

The creation of 3D building is performed using the Google SketchUp. After created, they can be uploaded to the N2GE server, and placed in a 3D models folder, to further be loaded on demand by users.

3.2 Implemented Features

3.2.1 Create/Edit/Remove N2GE Objects

This feature is absolutely a must-have on every LB-NMS, enabling the manipulation of every GUI Object on Google Earth. With this feature, administrators can start building the virtual environment with all information present in Nagios.

3.2.2 Real-time data

Nagios monitors the network in real-time, keeping all the administrators informed about the network health. On the Google Earth we want to accomplish the same goal, displaying the network information updated in real time. The adopted solution focus on simply updating the KML files, which are associated with every N2GE Object. When some information carried by the object changes, the respective KML is updated. This can happen, for example, when some service reaches its threshold and switches to a warning state. Another very common case is when the graphics associated to some equipment is updated. This change must be reflected on the equipment KML. The

KML files are on serverside, and loaded initially on the clienside, which periodically checks for new updates. This periodic checks are performed by KML's Network Links.

3.2.3 Web SSH

This feature takes advantage of the web-based interaction between the client and the N2GE, integrating in the browser an SSH shell-client to perform remote access. The displayed network elements may be targeted of some problems or maintenance tasks with some updates, and having a tool to perform quick remote access is very useful. The software used was the applet Mindterm, which supports the first and second SSH versions and the ciphers RSA, DES, 3DES, Blowfish, IDEA and RC4. This software is also lightweight, taking small time to download it.

3.2.4 Location Search

In this feature the Google location search service is used to look out for a particular address. This service will return the GPS coordinates of the specified address and translates the camera viewpoint to that address. Using this service, it is avoided to waste time making a manual search for each place on the earth globe.

3.2.5 Visualization Layers

There is often irrelevant information being displayed that can be turned off, helping the administrator to focus this attention on certain information of the network. The visualization layers, are the concept used to provide this feature. In this thesis, we try to explore this feature, creating different layers types, and providing options to make layer combinations. The five layers that were implemented are the following:

- Equipment type (router, switch, AP, server, or laptop)
- Object type (network element or link)
- Status type (up, down, warning, unknown)
- Floor

- IP address

For the floor layer, it was possible to add several rules, such equal, above or below a certain floor value. For example, we can create two rules to filter all the network elements above the third floor, and those in the six floor. This feature is important to represent each floor separately, because Google Earth does not support the creation of placemark above the ground, i.e., it can not create network elements for different altitudes, other than zero.

Other useful layer is the selection of network elements based on the IP address. This layer also supports rules, including a filter by a specific IP address or subnet, using the CIDR notation for the last one. This is a powerful layer, because it is possible to show only individual equipments and specific parts of the network.

The other layers provide filtering according to the selected type.

3.2.6 Client Concurrency

In order to deal with client concurrent accesses, a single exclusive lock for destructive operations was implemented, as edit and delete. When multiple users compete for these updates, the first winning access to the critical zone will gain the lock and perform their operation successfully. Others users attempting the same operations will fail to accomplish it, and they will be notified about the changes occurred in the server, by loading these changes.

3.2.7 Client Update System

Nagios may have simultaneous user access, and the information will be changed over time, is extremely important to implement a feature to update the client current system view. By logging all the operations, and knowing the actual system view and client view, it is possible to determine the steps required to make the client updated.

3.2.8 P2P SNMP Agent

To monitor the P2P network, another SNMP agent was defined using the TwistedSNMP Project [7], pro-

viding access to all v1 and v2 SNMP functionalities. The SNMP agent runs in port 20000, for each p2p node. The created MIB tables, are grouped in four main tables: routing service, storage services, peer selection service and service registration service. All the variables data type are defined as integer, easing the create of graphical templates for each variable, because they share the same base template.

4 Tests

This chapter presents the tests performed, in order to ensure proper implementation of the developed NMS. The performed tests were applied in two different scenarios, in order to test both static and mobile networks. The IST network was used to test a network with fixed equipments, while the ad-hoc over Manet was used to test the mobility component, made by mobile terminals.

4.1 IST Network

4.1.1 Scenario

To test the implemented NMS in fixed networks, a special test access was made available by CIIST to both the Taguspark and Alameda networks. To perform monitoring of the network elements of the two campus, the NMS server had access to the management VLAN.

This network provides an important test bed, because it is a complex network, composed by a large number of heterogeneous equipments, such as AP's, switches, servers and routers from many vendors and different series.

4.1.2 Network Mapping

This test aims to represent some N2GE objects in both IST campi, and observe the correct operation of them. The first objects created were the 3D models, representing the buildings structure. After this step all network elements and links on both campus were created. To finish this test, two cameras were created, one for each campi.

The Google Earth has some problems concerning indoor representation. For this reason, only some network elements for each zones of the map were represented, avoiding a confusing map with too many equipments together.

After launching the N2GE server and execute all test phases, the outcome results are illustrated in figure 3.

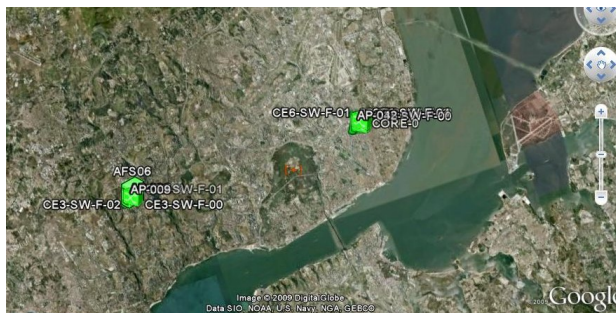


Figure 3: Complete IST-Network View

4.1.3 WebSSH

To test the ssh applet, a simple remote access to the server raquel was established. Figure 4 shows the ssh applet working in the N2GE environment.



Figure 4: Remote access using mindterm ssh applet

4.1.4 Visualization Layers

The N2GE has several visualization layers. To cover all of them, three tests were performed. On the first test, a combination of three layers were used, related to the type of equipment and the state of each one. This layer will filter other equipments that are not server or AP type, and different from Up state.

On the second test, the floor feature was tested, trying to filter all the equipments above the ground floor. All the AP's were positioned at the ground floor on every class room, while the mapped switches and the server are on the first floor, at CIIST premises. Applying the floor rule, i.e., higher than zero, all the AP's should be filtered, and only the equipments in higher floors should remain. After the test was performed, the obtained results prove that the layer works correctly.

For the third and last test, the IP feature was tested, its aim was to test the subnet filter. For the Alameda network, the mapped switches are the only equipments in the 172.128.2.0/24 subnet. Performing the filter based on this information, we can demonstrate the correct operation of this layer.

4.1.5 Concurrency

This test pretends to launch simultaneous destructive operations (write/delete) by multiple users, and prove that only one user can be successful. The selected number for concurrent access were three users, with the current system view, because a client cannot perform destructive operation, unless he is updated, seeing all the changes made on the NMS. After the test, all went well because two of them received an popup message saying that the operation failed, and only one submitted the operation successfully.

4.1.6 Consistency

The consistency tests are important to avoid an inconsistent NMS. These inconsistencies could appear when some user tries to make an update, and he/she is not aware of the last submitted operation. Another and very frequent event for this kind of application, results from changes on the Nagios configuration files, in order to add another equipment/service to monitor, or even to delete.

As described on the implementation chapter, all user operations include the system view. When it is received by the server side, it is compared to the current system view, and from this action we can check if the user did not lost any destructive operation (update, insert or delete). To test this issue, two users

were used. User A starts to make some operations, as the creation of two network elements, and after that, the user B logs into Nagios and receives the current view. Immediately afterwards, the user A creates another network element and a network link. If the user B tries some destructive operation, he/she should not succeed. This test run as expected.

4.2 P2P Network

4.2.1 Scenario

To accomplish this test, a P2P network running over a MANET implemented by 6 netbooks (Asus EeePC) was used, with identical software and hardware configuration, except for the special node, responsible to run the Nagios with the N2GE instance and monitor the entire P2P network.

In this network, there are two different networks operating simultaneously, one cabled and another one mobile. The cabled network is used to receive a WAN connection from the gateway (Linksys wireless-G), to access Google Earth service and facilitate the configuration of the P2P nodes using SSH. The attributed IP ranges distributed to cabled and mobile networks were 192.168.2.0/24 and 192.168.1.0/24, respectively. The middleware nodes includes an option to block the desired peers neighbours, dropping their packets, this way refusing peer links with them (at link layer). Using this option, we can create an overlay without to move the mobiles positions, remaining on the same place (testbed table), and forcing the desired overlay. The selected overlay enables to test multi-hop search throughout the network, with multiple possible paths. Figure 5 illustrate the selected network topology.

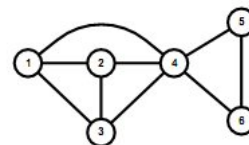


Figure 5: Network Topology

4.2.2 Network Mapping

This test validates the correct operation of the N2GE basic operation when applied to P2P over MANET networks. The procedure consists in the creation of all peer nodes and the gateway. Furthermore, a network link was added between the monitor node and the gateway. The obtained result, is illustrated in figure 6.



Figure 6: Ad-Hoc network mapped to Nagios

The green icon color of each equipment suggests the correct operation of each equipment. In order to perform a more detailed analysis of each object, all the extra information displayed in the HTML balloons were analysed. All the IP's were correct, and the graphics were correctly extracted from Nagios.

After the network mapping was created, it was tested an external access, by a laptop connected to the cabled network. Everything run as expected, since the laptop logged into Nagios and loaded the N2GE from the monitor node, and it observed the current mapping network. This access indicates an important feature, that the network can be monitored remotely.

4.2.3 P2P operations monitoring

This test tries to reproduce some typical operations in an P2P network, such as a search for a given document and other resources. This operations will affect the MIB tables, and the goal of this test is to use the N2GE to observe this events, by analysing some graphics generated for some MIB variables.

Before starting the test, it was used the P2P resource search service to find which P2P SNMP agents

are running, and from the nodes that reply to this query, it was checked that all of them were running the SNMP agent.

With the guarantee about the running agents, several requests were generated from the monitor node (other node could be used). These requests have modified the nodes MIB tables. The used syntax for the search command was the following, searching for debian packages.

```
./search.py -a file "extension:deb"
```

After several queries were generated, some traffic was generated. By the graphics generated, we can conclude that all nodes received approximately the same number of queries, which result in this traffic. Other graphics were observed, but remained static, e.g., the number of documents and database shared by every node. Other interesting test was related with the monitoring of the current number of peers neighbours. However, this variable had some operational problems and it was not possible to populate it, remaining empty. If all run as expected sending down node4, node01, node02 and node03 would have two neighbours, while node05 and node06 only one.

4.2.4 Mobility Test

For this test, the desired goal was to test a mobility situation that can occur in a mobile network. The test scenario consists on the separation of two nodes relative to the overlay, creating two islands when they become unreachable from the monitoring node.

The developed NMS does not include any mechanism to perform an automatic mapping of network elements to Google Earth, only manual, implying intervention of the network administrators to specify the precise location. To resolve this issue, it could be used a new variable on the MIB tables indicating the current equipment's GPS coordinates (longitude and latitude), collected from a GPS receptor attached to the laptop. But since no GPS equipment was available, and the tests were performed indoor, one python script was used to simulate mobility. This script inject GPS positions into the database, for nodes 5 and 6. These positions were collected in multiple campus

positions, from a certain path that increase its distance from the overlay. At certain point the distance between nodes 5 and 6 became unreachable, and this will result in another island.



Figure 7: Network status after node 5 and 6 signal faded away

After the test, the generated graphics on the monitor node were analysed, it was observed that the connections failed slowly, with the RTT increasing with time, until complete loss of the source signal.

5 Conclusion

This thesis proposes the development of a georeferenced NMS, based on the integration of the Nagios and the Google Earth systems. This solution explores the Google Earth technological potential to implement a rich NMS, displaying in real-time all the network information in a very interactive way.

One desired feature of this project is associated with the support of different scenarios. For this purpose, the developed NMS was tested using the IST network, and also a P2P network over a MANET. With these tests, it was possible to prove how important these systems can be, providing extra information and tools to monitor the network, helping even more network administrators.

The test results allowed to validate all the NMS features, namely the network mapping to Google Earth, WebSSH and Visualization Layers. The mobility test gave some positive notes about the NMS potential to create an automatic update for each node location.

We expect that this work represents a valuable contribution to the open source community, providing a suitable alternative to existing proprietary software.

5.1 Future Work

For future work, this application can include support for GPS receivers, and perform an automatic location update, like the project BelView. To implement this feature, it is only required to update the N2GE servlet to receive data from the GPS receiver and inject the coordinates into the database or directly into the KML files.

Other important feature is the representation of huge indoor networks. To solve this problem, the support of Flash [8] in the HTML ballons could help. With Flash programs, it is possible to create all possible building scales, and watch every place on it. The idea would be to use regular Google Earth for outdoor environment, and to develop a Flash-based 3D model for indoor environment when required.

References

- [1] J. Case, M. Fedor, M. Schoffstall, and J. Davin. A simple network management protocol (snmp).
- [2] U. Warrior, L. Besaw, L. LaBarre, and B. Hand-spicker. Common Management Information Services and Protocols for the Internet (CMOT and CMIP), October 1990.
- [3] <http://www.nagios.org/> Accessed in 28 September 2009.
- [4] Ant3nio Marques. P2P over mobile ad-hoc network, September 2009.
- [5] www.cacti.net Accessed in 17 October 2009.
- [6] www.zabbix.com Accessed in 17 October 2009.
- [7] <http://twistedsnmp.sourceforge.net/> Accessed in 28 September 2009.
- [8] <http://pt.wikipedia.org/wiki/AdobeFlash> Accessed in 17 October 2009.