Enabling Mobile Network Managers

James Reilly, Petri Niska
Nokia Research Center
PO Box 422, FIN-00045 Nokia Group, Finland
{James.Reilly, Petri.Niska}@research.nokia.com

Luca Deri, Dieter Gantenbein
IBM Zurich Research Laboratory
Säumerstrasse 4, CH-8803 Rüschlikon, Switzerland
{lde, dga}@zurich.ibm.com

Abstract

Network and system maintenance personnel are increasingly mobile. This creates a potential market for a network, system and service management terminal that is highly mobile, which would supplement existing network and system management solutions. This paper presents a generic architectural solution for this problem based on a highly scalable and network-centric approach to development of network management applications. Although the specific focus is on network management solutions, the results are generally applicable to many other types of applications as well. Some details and experiences from an actual implementation are described, using the Nokia 9000 Communicator and IBM Webbin' CMIP as the enabling technologies. Areas for future research are also explored.

Keywords: Network Management, Internet Technologies, Mobile Devices.

1. The Need for Mobile Network Management Terminals

Network Maintenance Operation Center (NMOC) personnel generally perform their duties by using sophisticated network management applications (windowing based), on relatively high powered network management consoles with large displays in the NMOC itself. Simpler network management applications also exist. For example *network node manager* applications are used to manage just a single node, and have reduced functionality and simpler user interfaces.

This paper assumes that network and system managers and maintenance personnel of telecom operators, internet (and other) service providers, as well as large private data networks are becoming increasingly mobile, and this creates the need for a simple highly mobile management terminal. There are many reasons for this, and several potential user profiles can be identified:

- *The warm fuzzy feeling*: When network managers are away from their office (due to work meetings, travelling, holidays, etc.), they may wish to be able to still perform some amount of their duties. For example, when very serious network or equipment failures happen, a network manager may like to instantaneously receive related alarms in spite of being away from his or her office.
- *Mobility as a part of the job description*: In this case, a primary function of the network manager's job is that he/she is mobile; and may be moving between physically separate locations (sites, cities, even countries) in order to perform his/her duties as a network and/or system administrator. It seems likely, at least in private data networks, that the role of network manager and system administrator may be performed by the same staff. It is probable that the need for more mobile dual-role sysadmin/network managers will grow along with the expansion of corporate intranets; and with out sourcing of management tasks by small to medium companies to their service providers and/or network operators.

- *Field installers and other craft*: When field personnel are installing new equipment in the field, they need some kind of terminal(s) to receive new work orders, configure parameters or perform simple tests on the equipment, and perhaps inform some centralized network planning agency of their organization when the equipment has been put into service. A hand-held mobile information appliance which can use the land-line or mobile telephone networks is quite desirable.
- *Customers of network and service providers*: The previous user roles assumed personnel employed by a telecoms operator (PNO), internet service provider (ISP), etc. Such organizations may wish to provide mobile or data network access to service subscription, management, billing, and customer care services for their best customers, in order to provide the best service in a highly competitive telecoms marketplace. For example a network manager of a private network (e.g. bank, hospital, or other organization) may wish to receive an alarm from his network operator or service provider in case of a very serious network outage. The capability for an operator to provide such information for their customers, or as an added value service, practically at anytime or anywhere within the GSM and ISP coverage is attractive.

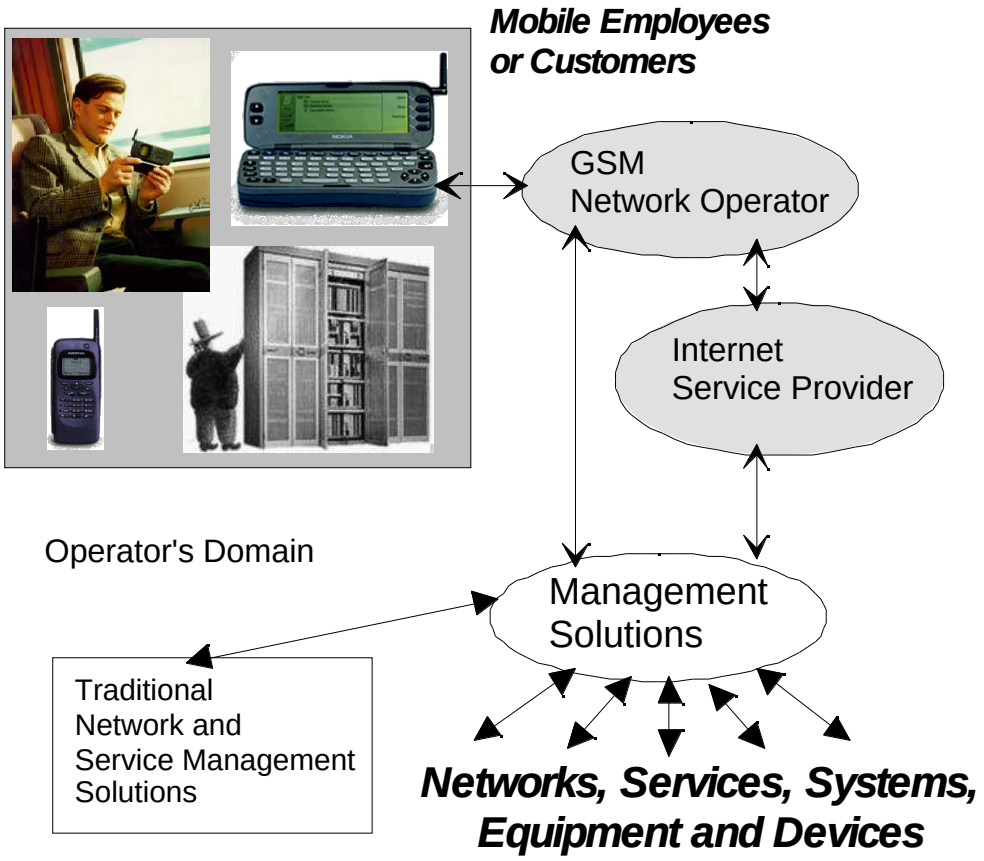

Figure 1. Mobile Web Access to Management Services.

Of course, these types of service can also be implemented using alternatives such as application specific software, or a laptop and modem. This paper instead concentrates on emerging handheld information mobile appliances like the Nokia 9000 Communicator [Nok1] [Nok2]. The Communicator is an integrated dual-mode GSM telephone handset and palmtop computer which supports Internet applications such as a Web-browser. Use of a device like the Communicator has additional benefits which make it attractive on its own: its small size and weight make it easier for even occasional use, it is one less thing (the laptop) to lug through the airport, and it may be easier to use in certain situations

(e.g. while hanging from the top of a radio mast or utility pole) than other alternatives. The particular requirements of a task will definitely dictate the choice of preferred device, according to the classification into being a more voice-centric or more data-centric task. In any event, the option of being able to participate in both worlds is becoming a common desire.

2. Network Centric Management Solutions

2.1. Overview

The traditional approach to the development of large complicated network management systems (NMS) and applications was to build them using a uniform platform-specific architecture, based on openly defined management interfaces such as defined by the TMN [Itu1] or SNMP [Snmp1] management frameworks. In the traditional model, a network management application's user interface is provided by an appropriate data-driven protocol to a windowing or other display (the F-interface shown in Figure 2.1). Recent advances in technology such as the World Wide Web (WWW) browsers and services, corporate intranets, and "executable content" such as provided by Java; introduce the possibility for a more network-centric approach to the development of management applications, that allows organizations to leverage their existing NMS solutions with generic, highly scalable and customizable Web-based front ends. Pavlou [Pav96] presents related results concerning the TMN F-interface.

The benefits for a network-centric management application architecture are the scalability (from the smallest terminal to the largest NMS application), portability, and flexibility of the solutions which can be provided in a generic way. Choosing a client-server distribution of the TMN manager role as an implementation architecture, it is possible to reach a high degree of platform and location independence. To factor out resource-intensive or location-dependent functions, the described portable clients communicate with *middleware servers* which are reached across the Intranet or Internet. It is expected that the Internet is going to substantially change the way we will look at network and systems management, including the management of carrier networks [Bogl97][Rei96]. In particular, Customer Network Management solutions will make use of Internet technologies, including aspects like directory and security. A uniform end-to-end integration of the overall management architectures will be one of the enabling factors for providing cost-effective offerings.

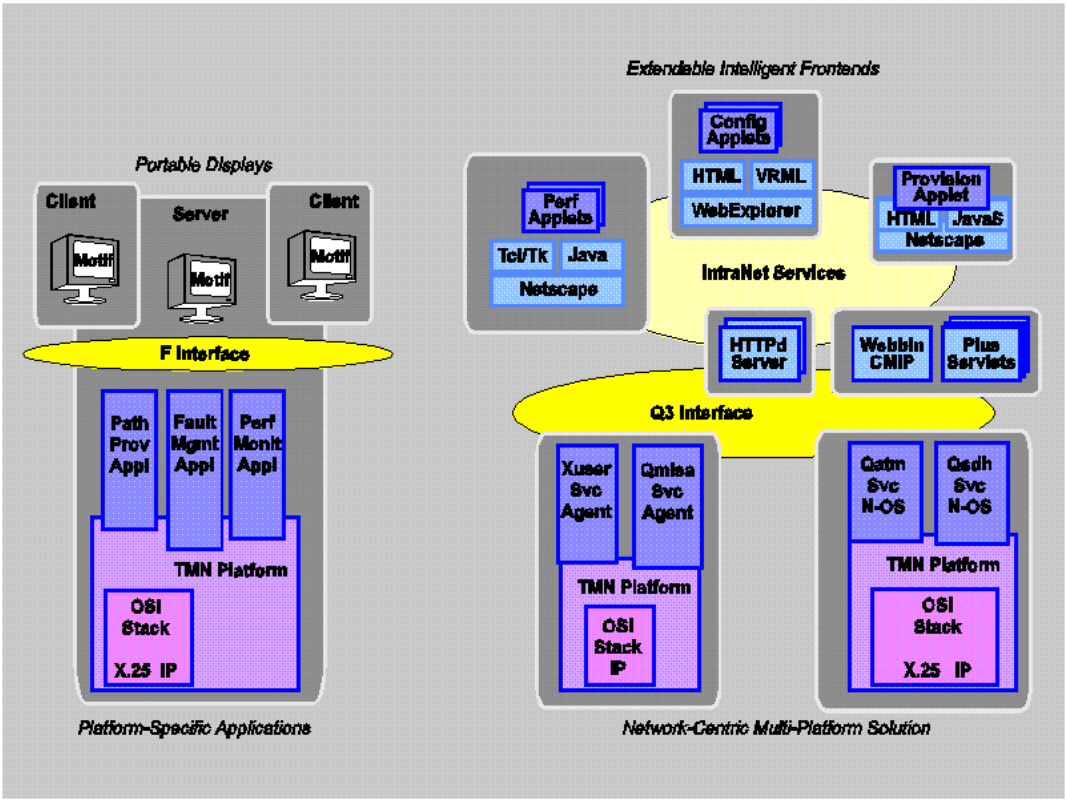

Figure 2.1. Network-Centric Approach To Management Applications.

2.2 IBM Webbin' CMIP as Middleware Server Component

IBM Webbin' CMIP [Deri95] is a research project which aims to simplify the way network management is performed. The core element of the project is a software application called *Liaison* [Deri96a] which allows for management of CMIP/SNMP resources through the Web. The core element of Liaison is the "Proxy server" which is based on a special type of software components called *droplets* that have the ability to be replaced and added at runtime allowing to dynamically modify and extend the behaviour of the application that contains them.

Each droplet, built upon shared libraries, implements one or more services. These components cooperate through the Proxy that takes care of the communication with the outside world. Proxy implements the HTTP protocol [Http1] hence remote web users can access it directly without the need to have, for instance, Common Gateway Interface (CGI) applications that interface the HTTP server with the Proxy itself (the CGI was defined by NCSA to interface applications to HTTP servers). This solution presents several advantages in terms of performance and configuration. The proxy comes with droplets that implement all the required CMIP and SNMP operations, a basic directory service and a metadata repository for SNMP. Additionally there are a couple of droplets that have the ability to query the metadata information contained inside the OSI stack. The idea is to implement a droplet for each management CMIP/SNMP operation and then cooperate with the existing droplets in order to reuse the services they provide especially with respect to the metadata access. This demonstrates how powerful software components are and how they allow to reuse existing services and then to incrementally build applications instead of starting from scratch every time.
The basic Liaison configuration contains droplets for:

- browsing CMIP and SNMP resources using a Web-browser;

- performing batch operations which are used by *external bindings*, which are some Java and C++ classes which enables the creation of simple management applications by exploiting Liaison's services [Deri96b];
- displaying network topology in 3D using VRML (Virtual Reality Markup Language) [Deri97b].

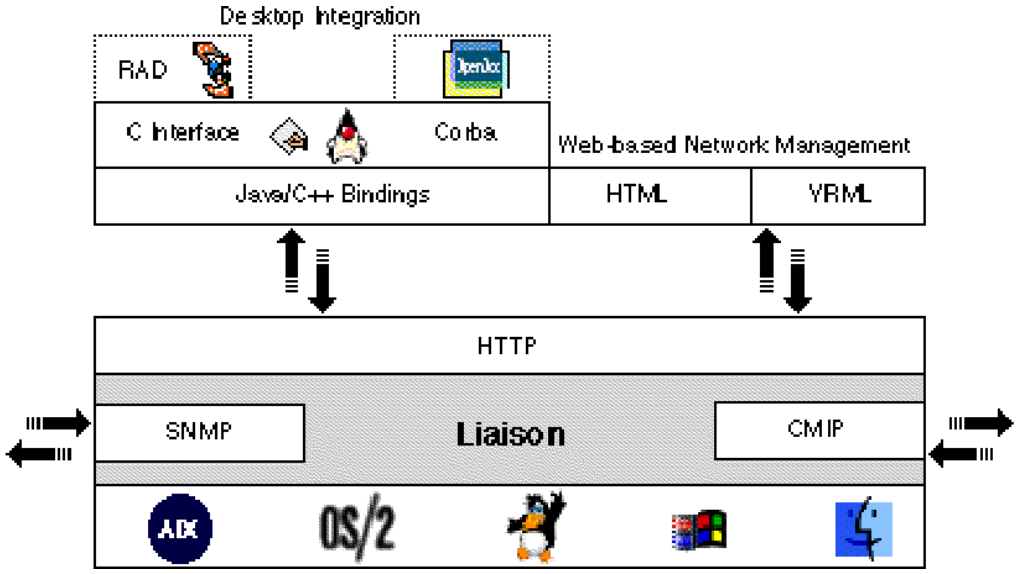

Figure 2.2. Overview of Webbin'

Additionally, Liaison provides *CORBA Bindings* which allow users to manage CMIP and SNMP instances from CORBA, exploiting the services provided by Liaison [Deri97a].

2.3. Web-based Network Management User Interfaces

The work presented in this paper grew out work performed in the ACTS MISA project [Galis96]. A single application (ATM Topology) is presented later in Chapter 3 as an example for a mobile web terminal. It is important to first explain how and why this type of application came into being in the first place. MISA concentrates on the management of integrated ATM and SDH management. The project has created web based (HTML and Java) network and service management applications. Some reasons for the use of web technologies to make the user interfaces of the management applications were:

- loose integration :- A network operator has the possibility for a loosely integrated interface, with a similar look and feel for many types of network management activities
- *familiar* :-Many people already use Web-browsers and web technology for other purposes
- *cost* :- a simple, low-cost management terminal that is available wherever a Web-browser can run; hardware and software costs are affordable
- *platform independent* :- because the application can run wherever a browser can, this gives more freedom
- *software versioning* :- use of executable content can allow new software versions to uploaded from the network transparently to the users
- *scalability* :- Web-browsers may run on very small devices to very complex ones.
- *correlated information* :- the possibility to access and correlate results with other types of information (e.g. from other types of databases, legacy systems) using IIOP and CORBA also seem very promising.

There are some disadvantages, mainly based on the growing pains of the Internet itself. Security of information traversing a public internet is still an important issue. Network management activities are often time critical (e.g. to resolve a fault), and are adversely affected by network latency, which may cause unsuitable response times for loading or execution of HTML/Java-based applications. HTML provides for a fairly static model of a user interface. Also, the functionality of user interfaces that can currently be efficiently implemented, as well as the tools to graphically design their layout, are somewhat limited at the moment. However, the situation for most of these issues is rapidly improving.

The Generic MISA Management Console (GMMC) is an example of a web based (HTML + Java) GUI/manager application from MISA. The goal of the GMMC is to provide "loose integration" of ATM and SDH network and service management applications. This means management applets have been implemented, using a common look and feel, as well as a common starting point for a human manager (in the employ of a network operator) to begin the execution of any applet (i.e. a common HTML starting page). The GMMC's philosophy is to "divide and conquer" - to split the management tasks into small portions (applet threads) which will together cover all the required functionality with a common look and feel. One task is to display and configure an ATM network Topology. This example is shown in Figure 2.3. where both the initial GMMC HTML/Java page is started, and subsequent ATM network topology task can be seen.

The same example of an ATM Topology application is used again on a mobile web terminal in Section 3.2. The main point to be remembered here, is that an existing web-based user interface for a management application can be made scalable and customized to the needs of the end-user (even a mobile one) by using generic web technologies.

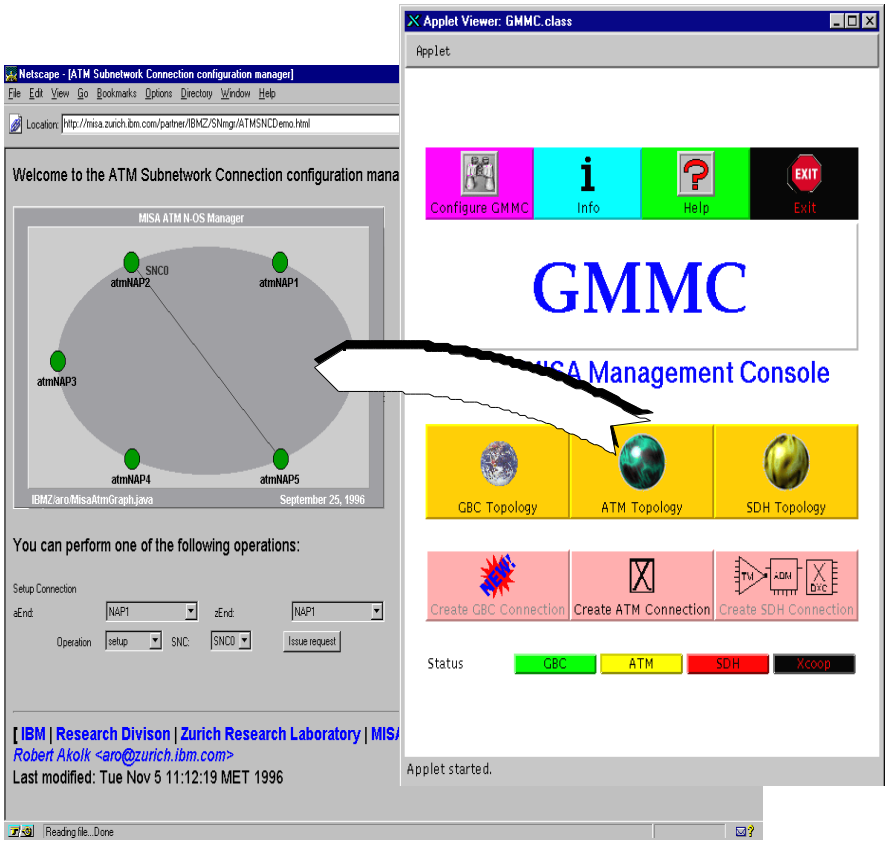

Figure 2.3: A Web-Based "ATM Topology" Network Management Application.

3. A Portable Network Management Terminal Implementation

3.1. The Nokia 9000 Communicator

The Nokia 9000 Communicator [Nok1] [Nok2] combines the benefits, portability and functionality of today's digital, data capable cellular phones and palmtop computers. It enables truly mobile connection to electronic media such as e-mail, FAX and the Internet (World Wide Web) from almost everywhere. The Communicator provides the following features:

- Full featured GSM Phone with the same data functionality (e.g. GSM short message service and data calls) as traditional digital cellular phones.
- Versatile Messaging Device: faxes, e-mails and short messages can be sent and received easily.
- Portable Access Terminal: Internet access (9600 baud) with Web-browser, Telnet and VT100 terminal emulation.
- Compact Personal Organizer: address book, calendar, note-editor, calculator and world clock.

One feature of particular interest in the context of this paper is
the Web-browser, which supports HTML 2.0 and has been used with Webbin'
CMIP to the management application described in the next section. As explained in Chapter 2, the MISA project has developed several HTML and Java based applications for network management. This general solution, additionally gave the opportunity to run and optimize some of applications on the Web-browser of the N9000, for use by mobile personnel.

3.2. A Design

One of the most useful attributes of a portable network management console would be the ability to register for and receive the most important or critical alarms about failures of network equipment, in order to be able to respond to them. This is shown in Figure 3.1. It is assumed that a real manager process exists somewhere to manage the network, receive alarms, set EFDs, allow specific human managers to register to receive certain types of alarms etc.

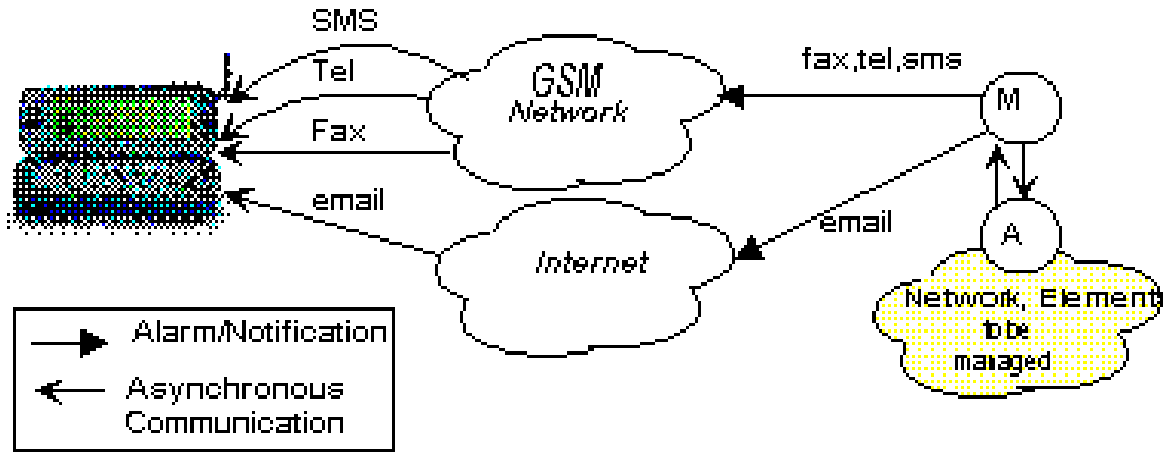

Figure 3.1. Reception of Asynchronous Alarms/Notifications

One basic metaphor for contacting the human manager could be similar to paging. Another is to pro-actively establish an HTTP connection, and use HTML2.0 to poll appropriate information. It is possible to use the periodic updating features of HTML 2.0 for polling purposes (i.e. create an HTML page that periodically gets updates/polls from an appropriate Manager/Agent). Using a paging metaphor, a piece of manager software has several possibilities to contact the human network manager: voice telephone call, fax, GSM short messages, or e-mail. These can be used to transmit alarms, traps or other appropriate event reports/notifications. With Fax, Short Message Service (SMS), or e-mail

instructions can be given to the human manager where to get more information (e.g. a URL to connect to with their WWW browser).

A paging metaphor (as shown in Figure 3.1) is an important mechanism to receive asynchronous alarms, as it would be expensive to maintain a permanent IP over GSM connection from the terminal shown to a remote manager. A mechanism is needed at the remote manager to allow one to register for certain types of asynchronous alarms and/or events. The remote manager may keep a permanent connection to the agent across a LAN or WAN. When events that are important to a registered mobile user occur, then the manager must forward the appropriate event to the user's mobile terminal using a FAX, telephone, SMS or e-mail connection. Using any of these mechanisms will cause a mobile terminal (like a Communicator) to ring or beep appropriately, and inform the user that a FAX, e-mail or SMS has been received. It is up to the remote manager to fill in the contents of the message sent, it may be as simple as informing the remote user to contact their HTML homepage, which can be updated automatically (behind the scenes) with the details of the appropriate events or possible actions which could be taken.

In response to reception of an alarm, the manager can respond in a variety of ways:

- do nothing
- call his office (similar to paging)
- take a telnet connection to an appropriate VT100 manager application
- take a Web-browser connection to an appropriate manager application

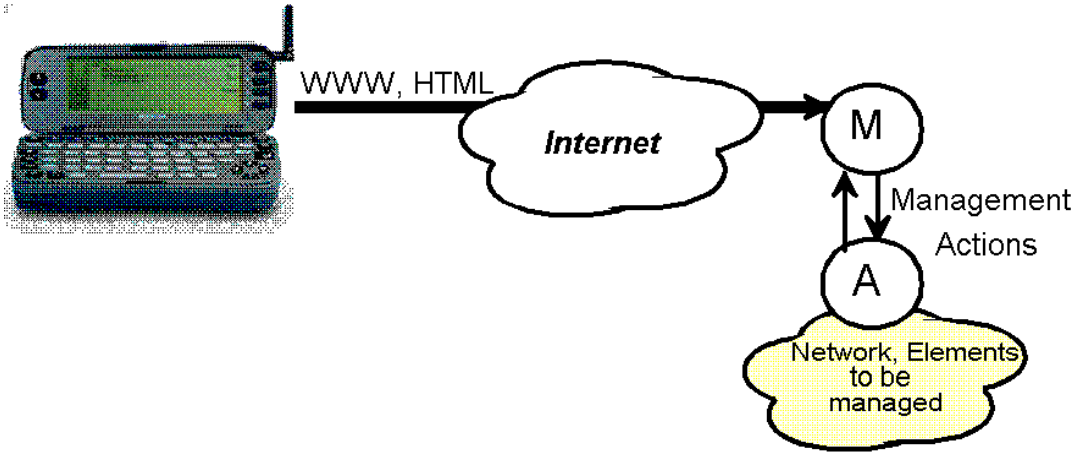

Figure 3.2. Using a Web-browser GUI for Network Management activities.

This document primarily concentrates on the last option (WWW browser as a GUI front end for manager applications) shown in Figure 3.2, and experiments on the reception of events using GSM short messages have also been performed. This is the architecture used for the following section on the implementation using Webbin' CMIP.

An important issue may also be the security of the Internet for receiving such alarms (e.g. via e-mail) or for interacting with remote managers. In this respect, use of GSM SMSs to receive alarms may be more secure and also more reliable :- only a GSM network connection is needed to send and receive SMSs. In addition, either a public or private ISP may be used. A private ISP means that one does not need to use a public Internet Service Provider, but could equally well provide their own private Internet services (HTTP daemons, etc.), on top of appropriate private dial-in IP over GSM connections. More

work is needed to study the potential security issues involved, but these are extremely important issues in practise. This is shown in Figure 3.3. It is important to note that pricing can also be affected. Using a public ISP allows one to make a less expensive local call to the ISP's nearest point-of-presence (POP) to gain IP connectivity, but one must accept the level of security provided by that ISP. Using dial-up access to one's own IP services, may be more secure but may also be more expensive if long-distance calls (typically the dominating price factor) are needed.

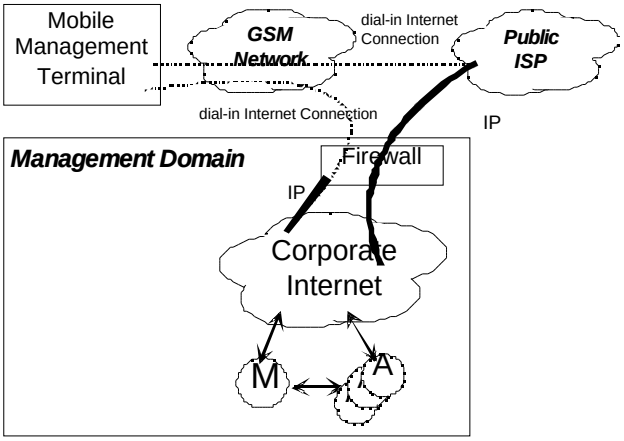

Figure 3.3. Security Issues Involved.

3.3. An Implementation

An experimental implementation of the ATM Topology application previously shown in Figure 2.3 has been created using a Communicator and Webbin' CMIP technologies. Such an application could be used, for example, by a mobile end-user to check the status of the access end-points of different operator's WANs. This prototype application is based on the design shown in Figures 3.1 and 3.2. The implementation to handle alarms is still an area for future improvements, as several design issues must be resolved in a practical implementation, and is not dealt with further in this paper. An important restriction for the implementation work, is that the Communicator does not currently support Java. Ideally, it would have been desirable to have directly ported the graphical and interactive applets shown in Figure 2.3. However, HTML and Webbin' CMIP offered alternative possibilities to implement an ATM Topology application in a similar manner. Although they may not be as graphical and user-friendly, they can still be just as useful to a mobile end-user. It is important to note, that many network management applications are basically textually oriented anyways (especially service subscription and billing).

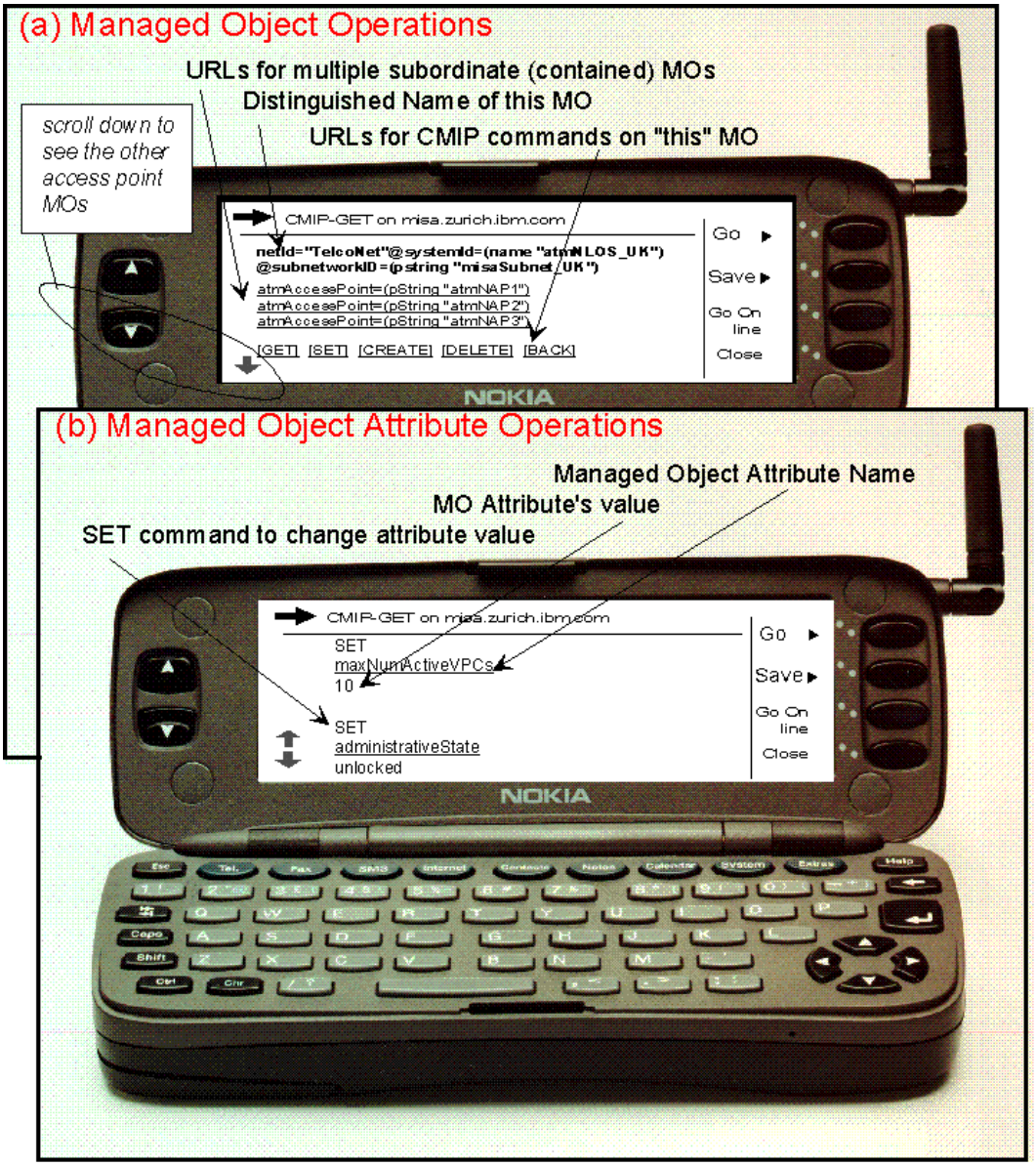

Figure 3.4. Managed Objects, Attributes and CMIP operations displayed using HTML.

Using the Webbin' CMIP environment, droplets are created for the desired management service. Figure 3.4 shows the displays from a Communicator browsing HTML pages written by a Webbin' CMIP droplet which represents managed objects (MOs) contained in a corresponding real MIB. The ATM topology can be viewed by contacting the appropriate agent for the subnetwork, and interrogating it (using the Communicator's web browser). The items displayed (from a remote agent) are the actual MOs of interest, and they are displayed in a structure similar to an ordinary file system. The distinguished name of the *currently selected* MO is shown as a plain-text string at the top, and contained MOs (ATM access points) are shown as URLs which can be selected. The MOs in this MIB represent ATM access points of an ATM subnetwork. It is possible to traverse to a contained MO (using the **GO** soft-key of the Communicator) by choosing the appropriate URL for it. CMIP commands which are allowed on the currently selected MO (**GET, SET, CREATE, DELETE**) are also displayed as URLs and can also be performed using the **GO** button.
It is possible to browse a *live* demo (CMIP agents for an SDH network) using a Communicator (or other Web-browser) by pointing it at the following currently public URL:

`http://misa.zurich.ibm.com:1998/CMIP/GET/MIBCTL/?detail=low"`

The command "`?detail=low`" at the end of the URL indicates that the HTML produced by the droplet is optimized for display on PDAs with limited display and bandwidth resources. Although the demo is customized for use with small-screen, bandwidth-limited devices, it can be viewed using any Web-browser. A more graphical layout (using tables and graphics) is created by the droplet if one does NOT specify the *?detail* command, or uses "`?detail=high`" in the URL. The *?detail=high* option would be appropriate for larger displays, for example those found on typical PC and LAN based Web-browsers, that are not limited by bandwidth and have greater display resolution than a hand-held device.

The following features are used in the HTML generated by a droplet for the "*?detail=low*". They have been based on our experimentation:

- use a very simple textual layout due to limited display size (telephone handset) to improve usability :- i.e. no forms or tables (although these are supported on the Communicator)
- use small font sizes for displayed text
- limited use of graphics, as large complex images take time and money to download using an IP over GSM connection
- do not use *internal-** images (this is quite common for folder icons, etc.) as these were originally Netscape specific and are not true HTML and can cause problems if a browser does not support them

By making the implementation in this way, the existing Webbin' CMIP implementation has been extended in a generic way (using the *?detail=low* parameter). More experimentation on the final look and feel of the application is needed, to improve usability aspects as much as possible.

4. Areas for Future Work

It is probable that the MISA project will implement other types of applications for mobile web users in the future. Usability aspects are also generally interesting, there is still room to improve the general layout and design customized for small screen devices. The implementation so far has concentrated mostly on the metaphor where the user contacts the system to be managed. A more sophisticated implementation for fault management, especially for asynchronous alarms (as described in section 3.2) would be a valuable addition.

The implementation of the application presented in this paper was based on the Nokia 9000 Communicator, which currently supports HTML v2.0. It is probable someday in the future, that mobile PDAs which additionally support the use of executable content, such as Java, in their Web-browsers will become available. This would give the opportunity to create more powerful managers/GUIs to be loaded and executed on mobile PDAs, such as the GMMC application presented in Chapter 2. In addition, the use of applets or servlets might someday become an interesting possibility (if mobile Java-enabled devices become available), to dispatch executable work to be performed (and status monitored) to one's mobile PDA. This could act as an enabling technology for *workflow* applications.

5. Conclusions

Many human network managers are increasingly mobile and will require improved mobile data access to network management and other services. A web-based, network centric design gives the flexibility to offer such a solution, and additionally allows one to scale up the solution because of its genericity. This paper examined the use of emerging web and mobile PDA technologies in this respect. We

believe the solution presented is one of the first such applications of its kind. An initial design was presented, as well as results based on an example implementation. The example was based on existing web-based network management applications created for management of broadband ATM and SDH networks. The results are generally applicable to a wide range of other applications, and alternate vendor solutions.

The work presented should be viewed as a *supplement* (and not substitute !) for existing network management systems and applications, which addresses the needs of certain types of mobile end-users. It can be used to complement and extend existing management services in a variety of ways. An overview of potential user profiles and services which could be offered was also presented. Based on our experiences, we believe that this is a new and very promising area for continued research and development.

The work presented in this paper was a result from research performed in the ACTS MISA project, and is not a commercial product offering from either Nokia or IBM. The authors gratefully acknowledge the support and funding of the European Commission EU ACTS project AC080 MISA (Management of Integrated SDH and ATM Networks), and the work of the other members of the MISA consortium whose project contributions we recognize and appreciate. This work is also supported by grants from the Swiss Federal Office for Education and Science (BBW).

6. References

[Nok1] Nokia 9000 Communicator specs, http://www.club.nokia.com/support/9000/specifi.html.

[Nok2] Nokia 9000 Communicator homepage, http://www.nokia.com/com9000/n9000.html.

[Java1] Sun Microsystems, Java Homepage, http://java.sun.com/, 1995-97.

[Snmp1] J.Case, M. Fedor, M.Shoffstall, J. Davin, *A Simple Network Management Protocol (SNMP)*, Internet RFC 1157, May 1990.

[Itu1] ITU-T Recommendation M.3010, *Principles for a Telecommunications Management Network*, Geneva, Nov 1991.

[Html1] Hypertext Markup Language (HTML): Working and background Material, http://www.w3.org/pub/WWW/MarkUp/MarkUp.html.

[Http1] HTTP: A Protocol for Networked Information, http://www.w3.org/pub/WWW/Protocols/HTTP/HTTP2.html.

[Bogl97] G. Bogler, *Internet - New inspiration for Telecommunications Management Network*, accepted for publication by IS&N '97, Como, Italy, 27-29 May 1997.

[Galis96] A.Galis, et.al., *Towards Multi-Domain Integrated Network Management for ATM and SDH Networks*, Proceeding of International Symposium on Advanced Imaging and Network Technologies, Conference on Broadband Strategies and Technologies for Wide Area and Local Access Networks (BSTW'96), Berlin, Germany, Oct 7-11 1996 (published by IEEE), http://misa.zurich.ibm.com/World/Gali96b/finalPaper.ps.

[Deri95] L. Deri and A. Weder, *Webbin' CMIP*, Proc. of 3rd Int. WWW Conference, Darmstadt Germany, 10-14 April 1995, http://misa.zurich.ibm.com/Webbin/.

[Deri96a] L. Deri, *Surfin' Network Resources across the Web*, Intl. IEEE Workshop on Systems Management", Toronto, June 1996, pp.158-167, http://www.misa.zurich.ibm.com/~lde/Surfin.html.

[Deri96b] L. Deri, *Network Management for the 90's*, ECOOP'96, Workshop on OO Technology for Service and Network Management, Linz Austria, 8 July 1996, http://misa.zurich.ibm.com/~lde/SimpleNM.ps.

[Deri97a] L. Deri and B. Ban, *Static vs. Dynamic CMIP/SNMP Network Management Using CORBA*, accepted for publication by IS&N '97, Como Italy, 27-29 May 1997.

[Deri97b] L. Deri and D. Manikis, *VRML: Adding 3D to Network Management*, accepted for publication by IS&N '97, Como Italy, 27-29 May 1997.

[Pav96] G. Pavlou, Th. Tin, *A CMIS-capable Scripting Language and Associated Lightweight Protocol for TMN Applications*, IEEE Communications, September 1996.

[Rei96] J. Reilly, *The World Wide Web and Programming Future Broadband Network and Service Management Applications*, Seventh bi-annual Nordic Workshop on Programming Environment Research (NWPER'96), Aalborg Denmark, May 1996, http://www.iesd.auc.dk/~normark/NWPER96/proceedings/paperM.html.