Ambient Intelligence in Network Management

Mary Luz Mouronte[1,2], Pilar Cano[2], and Miguel Ángel Fernández[1,2]

[1] Universidad Carlos III, Universidad 30,
28911 Madrid, Spain
[2] Telefónica Investigación y Desarrollo, Emilio Vargas 6,
28043 Madrid, Spain
mmouront@it.uc3m.es, mlml@tid.es

Abstract. This paper presents a technical method for solving the main provisioning problems on transmission networks automatically: communications, naming, misalignments, etc. This solution incorporates users' experience and business knowledge in expert agents which execute specific actions on the Network Management System (NMS) when an error occurs. The human intervention is reduced so that OPEX and network management are improved.

This paper gives an overview of the NMS of Telefónica España (GEISER), where the described method is applied. The framework has been verified in the actual network scenario while new features have been validated with simulated requests and tested on a real testbed.

Keywords: Network Element (NE), Network Element Manager (NEM), Network Management System (NMS); Network failures, Ambient Intelligence (AmI), OPerational EXpenditure (OPEX).

1 Introduction

A NMS of the transmission network is a combination of hardware and software used to monitor and control all the equipment in the network. In particular, the provision function is responsible for setting up the circuits or path routes. This paper proposes a method to improve the provision from a NMS whose main problems are the communication failures and the errors by inconsistency of entities.

Specifically, this paper applies the technical procedure to NMS of Telefónica España (GEISER). This NMS is a unified network management solution which provides end to end view and homogenous functions across different vendors. Furthermore, it executes all business processes related to the transmission network and its services: network creation, circuit provisioning, network supervision and performance monitoring. This NMS manages Synchronous Digital Hierarchy (SDH), Ethernet over SDH and Wavelength Division Multiplexing (WDM) networks. The operation is highly complex: it handles over 36.000 Network Elements (NES), which belong to different providers [1] [2], so it has a large number of procedures which are used by over 500 operators.

For all these reasons, when a failure occurs, the operators spend a vast amount of time to solve it. This paper presents a technical method to solve the main provisioning problems. This solution incorporates users' experience in expert agents which execute specific actions on the NMS without reducing its performance, achieving less human

I. Tomkos et al. (Eds.): BROADNETS 2010, LNICST 66, pp. 385–397, 2012.

intervention, OPEX reduction and reliability improvement [3]. The described method is based on the Ambient Intelligence (AmI) issue.

AmI represents an intelligent service system to enhance operations. It provides a context aware system, which uses unobtrusive computing devices that improve people's life and work quality by acknowledging their needs, requirements and preferences and thus acting in some way on their behalf [4].

The rest of the paper is organized as follows: section 2 discusses relevant related work, section 3 gives an overview about the NMS of Telefónica España (GEISER) and also about the most important problems in the network provisioning, section 4 goes on to describe the approach to solve these problems automatically, section 5 summarizes the results while section 6 describes main conclusions and evolution areas.

2 Related Works

The architecture of a NMS has several management layers. Each layer has its own functions and is connected with its upper or lower layer through interfaces [5]. Users control an specific network domain [6]. and the NMS must have all required services for its management [7], during the execution of these services different errors can happen.

There is no previous works on applications using expert agents to solve provision problems on the transmission network from a NMS. Nevertheless, similar methods have been developed in others areas such as:

- Manufacturing and supplying chains where the solution consists of a community of autonomous, intelligent, and goal oriented units which cooperate and coordinate their decisions to reach a global goal [8].
- Smart home where the AmI concept is implemented to show how simple devices may be networked and how several tasks may be automated to get an intelligent health management [9].

3 Overview

The Transmission Network is a strategic Telefónica España asset. It is a multi-vendor network and carries all kinds of services:

- Imagenio (IPTV)
- ADSL/FTTH, IP Services, Ethernet VLAN
- POTS, GSM, GPRS, etc
- Leased Circuits.

The NMS of Telefónica España manages SDH, Ethernet over SDH and WDM networks. Its main features are:

- Network model based on standards like ITU-T G.803, ITU-T G.805, ITU-T G.709, complemented with the extensions required to support Telefónica's business.
- It provides unified network management, independent of vendors.
- It connects to the vendor Network Element Managers (NEMs). The interaction with the plant is carried out using the northbound interfaces that the NEMS offers. The NEMs interact with the Network Equipments (NE).

- End to end control of the whole transmission network.
- Complete functional support in the following areas: fulfillment (network creation and circuit provisioning), network supervision and performance monitoring.
- Simple and intuitive user interface, based on well-known web technologies, which allows accessing the NMS from any point of the Telefónica corporate network, defines operating profiles adapted to each user and enforces a strict security policy. An essential requirement of the system is to reduce the training time of new users.
- Connection to corporate systems of Telefónica España in order to automate most routine tasks for creation and provisioning of the network. Its main goals are: to shorten the time needed to solve network issues and to reduce the frequency of human interventions in operational processes. The interconnection with the Telefónica corporate uses MQ-Series.

The NMS processes:

- 130.000 alarms per day and correlates to 4.000 root alarms/300 network troubles,
- 1000 fulfillment requests (network creation, circuit provisioning) per day.

The NMS is used by 500 users.

Different functions are integrated in the NMS:

- Circuit provisioning: this function sets up the circuits or path routes: SDH (2 Mbps and 34 Mbps), Ethernet: (2 Mbps, 10 Mbps and 100 Mbps). It also maintains the consistency in the inventory, the alignment with the corporate repositories and with the NEMS. The NEMS belong to different vendors.
- Network creation: it executes operations over the NEMS regarding NES, cards, circuits, paths and physical links. In particular: SDH Path (155 Mbps, 622 Mbps, 2, 5 Gbps, 10 Gbps, 40 Gbps, 100 Gbps); SDH Circuits (155 Mbps, 622 Mbps, 2, 5 Gbps, 10 Gbps and 40 Gbps) and Ethernet circuits (1 Gbps, 10 Gbps and physical links: 155 Mbps, 622 Mbps, 2, 5 Gbps, 10 Gbps and 40 Gbps). This function maintains the consistency in the inventory, the alignment with the corporate repositories and with the NEMs.
- Inventory: this function maintains the network repository in NMS and carries out the tasks of auditing and discovery.
- QoS: this utility executes operation about network quality assurance and the Service Level Agreements SLAS.
- Surveillance: this function carries out automatic alarm correlation, root cause identification, end to end view and protection against "alarms floods".

The NMS is a system constructed around standards based network model, supported by an ORACLE DBMS, with a business logic layer that allows interacting with the core applications through a CORBA bus. The NMS works as a centralized system, with a primary machine of 16 CPUs, another one of 4 for the mediation with the plant and three more for users' access.

In this paper the proposed solution is applied to the main provisioning problems although it may be extended to all troubles that happen in the fulfillment.

The circuit provisioning requires high performance and efficiency. The NMS must process a large number of requests with high success percentage at the first attempt and understand the different kinds of requests: setting up, elimination, modifying, restoration and rerouting.

NMS receives the provisioning requests from a corporate repository where allocation tasks are performed. It is joined to different NEMs which interact with the NES to build the SubNetwork Connections (SNC) which are necessary to set up the circuits. Fig.1 represents the architecture of the system.

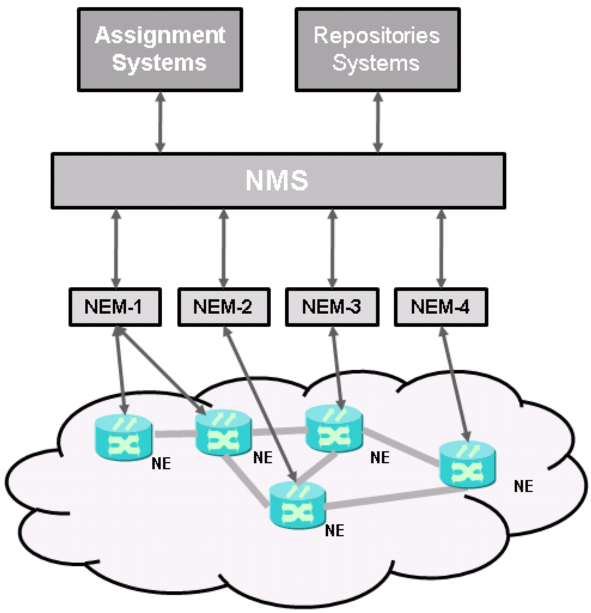

Fig. 1. NMS Architecture

The NMS deals with an average of 520 provision requests daily, with a 70% success rate at first attempt. Each request is split in different operations over the NEs which be included in the route (build/elimination of SNC, build/elimination of Subnetwork Connection Protections (SNCP), build/elimination of Virtual Circuits (VC), configurations of Link Capacity Adjustment Scheme (LCAS), etc.). These operations are executed on the NEMs.

Different problems have been identified in network provisioning, they are described bellow: Internal error, NE communication, Entity does not exist, Entity already exists, Unknown owner identifier, Busy entity, Processing failure, Functionality not supported, NMS communications and others.

3.1 Current Status of Network Errors Management

Vendors offer catalogues about the errors in the NEMs and the NEs. The failures in the operations have been analyzed monthly and a small subset of errors is repeated regularly:

1. Internal error: it refers to communication internal errors in the NMS.
2. NE communication: these errors are related to the absence or rejection connection between the NEM and the NE.
3. Entity does not exist: it is a data misalignment problem, the NMS does not have the same entity that the NEM.
4. Entity already exists: it represents a data misalignment problem, this error occurs when NMS tries to create an existing entity in the NEM.
5. Unknown owner identifier: it represents a naming problem, this failure occurs when the NMS does not have the data to operate on the NEM.
6. Busy entity: this error occurs when NMS tries to occupy some entity which exists in the NEM; it represents a data misalignment problem.
7. Processing Failure: these errors are internal of NE (e.g. Management Information Base (MIB) in wrong state).
8. Functionality not supported: this error occurs when NMS requests a not supported operation to NEMs.
9. NEM communication: It is a communications problem between the NMS and the NEMs.

Fig. 2 shows the average frequency of each error pattern in the daily operation over the network. Nine patterns are the most frequent (90%). The pattern “Unknown owner identifier” is in first place.

When the operator finds any of these errors, he can adopt different solutions to solve it:

1. Retry: failure is expected to be sporadic and the user repeats the action.
2. Do it locally: the user implements the action directly in the NEM which produces a misalignment between the NMS and the NEM.

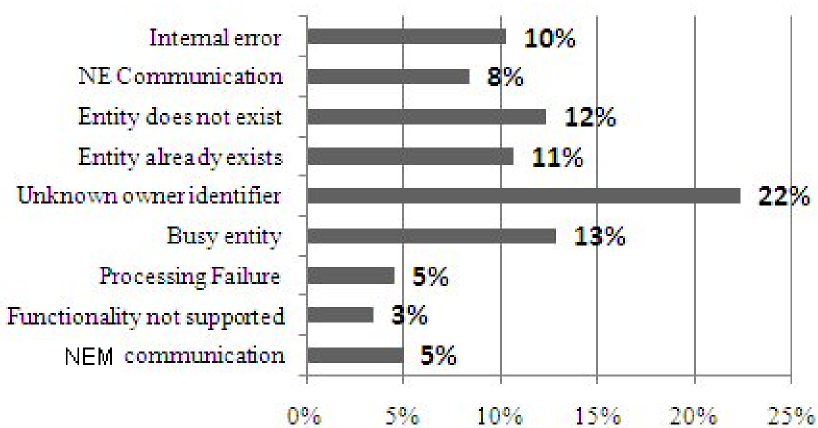

Fig. 2. Percentage of each class error over the total failures. In a month, the specific error average has been calculated in daily operation and its percentage has been considered over the total failures.

3. Align and retry: the operator can solve the problems by mean of specific actions in the NMS in order to align data (for instance, update the owner identifiers and then repeat the action).
4. Notices: the user send a notice to technicians to solve the problem.

On a month operators have been observed while working and they often solve errors in the NEMs directly. This practice produces data misalignments which will cause new failures in the future. The Fig. 3 shows the misalignment caused by this way of resolving errors. If operators do not solve errors in the NEMs then they must execute several procedures in the NMS and spend a vast amount of time to solve failures. It is shown in Fig. 4.

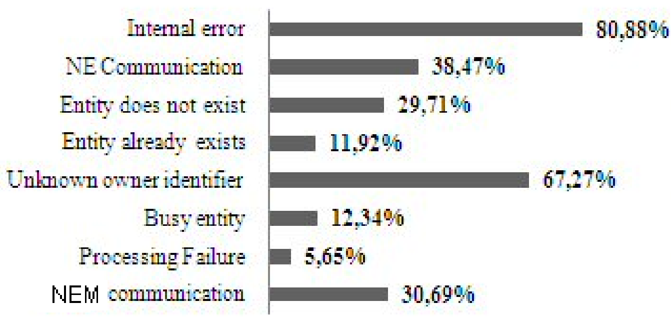

Fig. 3. Percentage of misalignment produced by resolution of each class error over the total failures. The average of misalignment by the resolution of each error has been calculated in the daily operation and its percentage has been considered over the total failures in a month.

3.2 Correct Network Errors Management

The correct actions to solve the errors are:

- Internal error: the operation should be tried again after a while.
- NE communication: all operations related to the NE with communications problems should be stopped until the situation is recovered. If the problem persists then a report with information about the failure has to be sent to technicians.
  - NEM communication: the actions to be done are similar to the ones in NE communication error case, but the failure affects to all the operations on the NE connected with the NEM.
  - Processing Failure: a report with information about the failure has to be sent to the technicians.
  - Functionality not supported: the operator must be warned about it.
  - Owner identifier unknown: in the NMS, all operations on the same NE should wait for the adaptation of owner identifiers.
  - Entity does not exist: it is necessary to compare the data on the NEM and on the NMS which allows the operator to solve the misalignment.

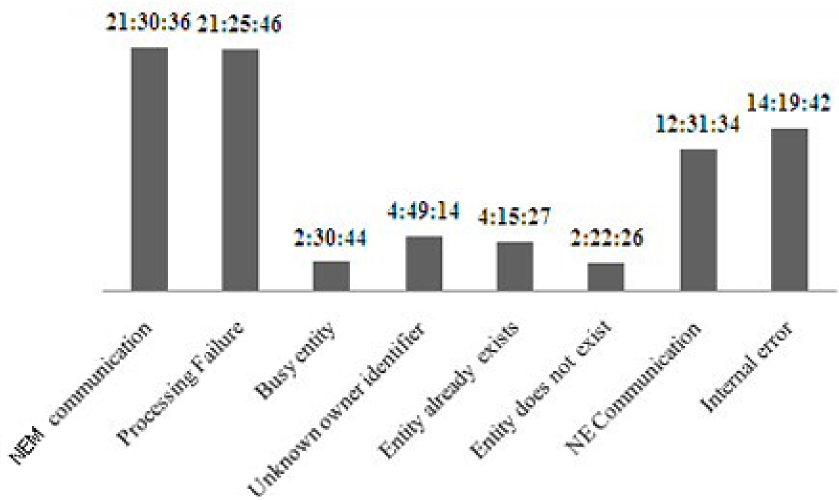

Fig. 4. Average time for solving each failure. The time is expresed as hh:mm:ss.

- Entity already exists: it is necessary to audit the situation to solve the problem.
- Busy entity: it is necessary to remove any residual entity in the network.

4 Solutions

The proposed solution in this paper, allows supervising and executing actions on the NMS automatically as soon as a failure occurs. The architecture of the solution is shown in Fig 5. It uses ORACLE mechanisms and C Language Integrated Production System (CLIPS).

CLIPS is an open source expert system tool developed by NASA-Johnson Space Centre. It is fast, efficient, free and updated and supported by the original author, Gary Riley [8]. CLIPS has been designed to facilitate the development of software to model human knowledge or expertise through rules, facts, functions and even object-oriented programming. Besides, CLIPS is designed for full integration with other languages [11].

The components in the architecture are: expert agents (Event Manager and Action Manager) and a Knowledge Base which contains different rules.

Expert Agent: *Event Manager*

In NMS, a trigger is fired by an error and it writes an event in an ORACLE queue, such information is collected by mean of SQL*Net [12] and processed by *Event Manager.*

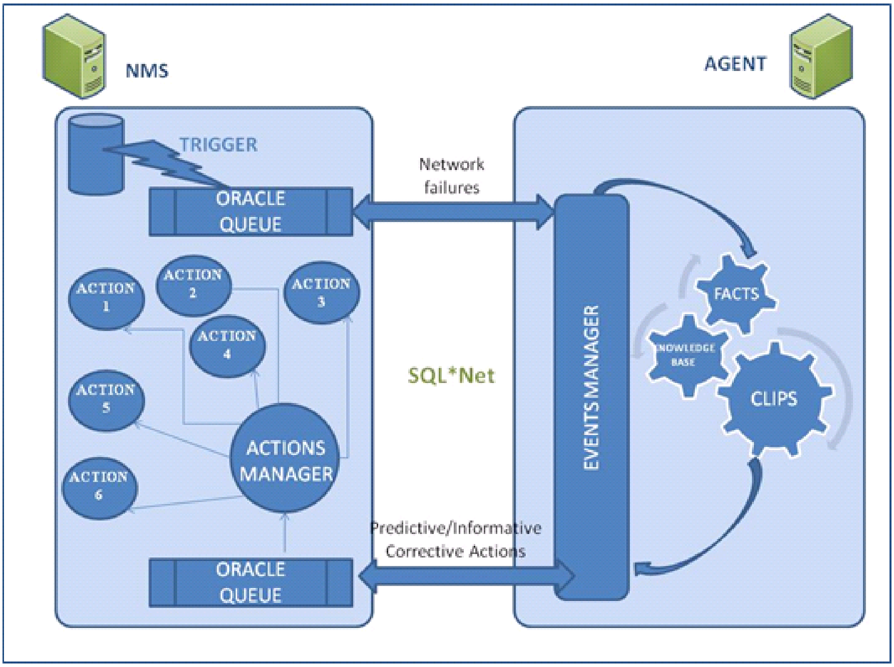

Fig. 5. Arquitecture

This event has the following format:

```
  <date>|<id action>|<id operation>|<failure
description>|<equipment>|<NEM>
```

Instance:

```
<17/06/2010 13:46:53|283214|153292|Failure: Q3:
OmisCnf::errProcessingFailure|B.ET0050TX01:TTSF16S
3:1662|INICIAL.GER:NM_NR7.0:105>
```

Event Manager does the following tasks:

- Read and send an event to an inference engine, which is written in CLIPS, where it is processed as a fact. It is not necessary to modify event information.

Instance:

```
      (defrule processing-failure; this rule establishes
if the type of failure is "Processing Failure"
      ?h1 <- (message (equipment ?equipment) (error
?error & 21068|21177|22052) (id_action ?id_action) (NEM
?NEM) (times 1))
```

```
        =>
        (printout t "CLIPS: Equipment " ?equipment "has
received a Processing Failure. Code: "?error crlf)
        (printout t "CLIPS: Send action type 3" crlf)
        (printout t (send_action (str-cat
"3||"?equipment"|"?NEM"|"?id_action)) crlf); To put
incidence
        (retract ?h1)):
```

- The inference engine decides the necessary actions to be sent to the *Event Manager* which sends information to an ORACLE queue through SQL*Net, where it will be processed by the *Action Manager*. The actions have the following format:

```
  [<type of action>|<description>|<equipment>|<NEM>|<list
of actions>]
```

Instance:

```
  Action: Retry
   [1||B.ET0050TX01:TTSF16S:18||231385]
```

Expert Agent: *Action Manager*

The *Action Manager* processes the instructions generated by the inference engine and interacts with the NMS to perform the suitable actions. This expert agent does the following tasks:

- Read the actions from ORACLE queue.
- Execute the specific procedures in NMS.

Knowledge Base

It contains different kinds of rules:

- Correlation rules: to establish relations between the facts and construct new ones.
- Decision rules about the actions: to determine the specific criteria to shoot actions.
- Metacontrol: to establish priorities in the rules according to their relevance.

The described architecture allows decoupling of the integration layer (ORACLE queues) from the business logic layer (management of rules and execution of corrective actions). In this way, This is an independent solution from the NMS which supervises and makes decisions based on users' experience. It does not interfere in normal NMS workflow.

5 Solutions

An experimental prototype aimed to test and validate the improvement in network provisioning has been prepared.

The rules of business knowledge have been implemented in CLIPS and they include the corrective actions to be performed when the NMS finds a network error.

Once the business knowledge has been modelled, the trigger has been prepared, the ORACLE queues have been configured and all processes have been implemented, which can be seen in Fig. 5, a set of provision requests were prepared to send them to the NMS. Such requests were processed by the NMS which sent the associated operations to NEMs. A set of 100 requests were prepared which result in 700 operations on the network. Of these 100 requests, 70 finished successfully at first attempt. However, the other 30 had problems and some of its operations cannot be executed successfully on the NEM. These 30 requests were integrated by 210 operations on the network, of which 70 ended unsuccessfully. Table 1. shows the distribution of these 70 errors.

The 70 identified problems were sent to our architecture and the agent took the associated decisions. These decisions were:

- Internal error: the agent tried again the 7 occurrences of this error. These redone operations finished successfully every time. Those new attempts were made 10 minutes after the error occurred.
- NE Communication: there were 7 occurrences of this error, 3 of them happened in the same NE and the other 4 in another one. From the first error in the NEs, operations were stopped by the agent. The two NEs were isolated from the NMS, and communication was recovered 23 minutes later. After that, these 7 operations were sent again by the agent and they finished successfully.

Table 1. Error Distributions

Error	Times
Internal error	7
NE Communication	7
Entity does not exist	7
Entity already exists	7
Unknown owner identifier	18
Busy entity	7
Processing Failure	4
NEM Communication	5
Functionality not supported	2
Another errors	6

- Entity does not exist: there were 7 occurrences. In all cases the agent decided to execute the audit procedure automatically in order to compare the situations in the NEM and in the NMS. The agent showed the results to the user. The misalignments were solved.

- Entity already exists: the agent had the same behaviour as in the previous case.
- Unknown owner identifier: in the 18 occurrences, the agent decided to execute the procedure for updating the information. 4 NEs were affected. The procedures ran in parallel and finished in 7 minutes. Eventually, the agent decided to retry the operations which ended successfully.
- Busy entity: the results for this case were similar to the misalignment cases.
- Processing failure: in the 4 occurrences, the agent decided to send a notification to a technician in order to review the state of the equipment.
- EMS Communications: the 5 occurrences happened in the same NEM. When the first failure occurred the agent decided to stop the operations on the NEM and retried them a time later, once the communication was reestablished. The operation finished successfully.
- Functionality not supported: in the 2 occurrences, the NMS could not program a loop in a port because this operation was not supported on the interface with the NEM. Both problems were unrecoverable, so that the agent decided not to do anything.
- Other errors: these errors did not have a regular frequency. No specific actions were planned.

The prototype solved 90% of errors automatically and its actions did not cause other failures. It contained 102 rules, 88 of them were correlation and decision rules and 14 were metacontrol rules. The operations finished successfully in a short time, as we see in Table 2. This table also shows the spent time by the operators at the present.

All errors in network provisioning can be included in the definitive solution by extending the knowledge base with new rules, in this way, all problems would be solved automatically.

6 Conclusions and Future Works

This paper described a technical method for automatic solving fulfillment problems on the transmission network. In particular, this procedure is applied to network provisioning. This solution is based on expert agents who detect failures and solve them by means of users' experience.

Table 2. Results

Error	Average Time with Agent (hh:mm:ss)	Previous Average Time (hh:mm:ss)
Internal error	00:10:23	14:19:42
NE Communication	00:23:12	12:31:34
Entity does not exist	00:40:34	2:22:26
Entity already exists	00:40:34	4:15:27
Unknown owner identifier	00:43:28	4:49:14
Busy entity	00:24:21	2:30:44
Processing Failure	15:45:28	21:25:46
NEM Communication	00:20:15	21:30:36

The solution uses different technologies to build an AmI environment which provides greater efficiency in the operations on the transmission network and reduces the OPEX because when a failure occurs, the operators do not have to spend so much time solving it.

This method can be applied to other areas:

- Adaptive graphical user interface: it is possible to detect user preferences for developing adaptive intelligent user interfaces [13] [14].
- To correct data in repositories: a right inventory SDH, Ethernet, and WDM components is complicated because many operations are necessary due to the multilayer hierarchy in these networks. Sometimes information is corrupt in data repositories. These situations could be solved by mean of expert agents based on users' experience.

Acknowledgments. We thank our department colleagues their support in the development, especially Alfonso Badillo Llada and Francisco Javier Ramos Gutiérrez.

References

1. Mouronte, M.L., Benito, R.M., Cárdenas, J.P.: Complexity in Spanish optical fiber and SDH transport networks. Computer Physics Communications 180(4), 523–526 (2009)
2. Santiago, A., Cárdenas, J.P., Mouronte, M.L., Feliu, V., Benito, R.M.: Modeling the topology of SDH networks. International Journal of Modern Physics C 19(12), 1809–1820 (2008)
3. Chan, V.: Optical network architecture from the point of view of the end user and cost. IEEE Journal on Selected Areas in Communications 24(12), 1–2 (2006)
4. Gasson, M., Warwick, K.: D12.2: Study on emerging AmI technologies, Future of Identity in the Information Society FIDIS Deliverables, European Union (2007)
5. Song, H., Xu, Y., Gui, X., Zhang, J., Gu, W.: Design and implementation of intelligent optical network management system. In: Proceedings of ICCT, pp. 625–628 (2003)
6. Mir, M., Banerjee, A., Mao, L., McGinnis, E., Sheikh, H., Galimberti, G., Jerphagnon, O., Ati, R., Liem, S., Gyurek, R., Tatipamula, M., Gerstel, O.: A GMPLS-controlled multi-vendor optical network demonstrating a variety of switching, control, and protection mechanisms. In: 31st European Conference. Optical Communication ECOC, vol. 2, pp. 215–216 (2005)
7. Verdi, F.L., Magalhães, M.F., Cardozo, E., Madeira, E.R., Welin, A.: A service oriented architecture-based approach for interdomain optical network services. Journal of Network and Systems Management 15(2) (2007)
8. Industrial Adoption of Agent-Based Technologies, `http://www.inf.udec.cl/~atkinson/cursos/electivos/mas/Papers/APLICA-industrial.pdf`
9. Ambient Intelligence and Home Networking for Wellness Management and Home Automation, `http://www.ercim.eu/publication/Ercim_News/enw47/korhonen.html`
10. `http://en.wikipedia.org/wiki/CLIPS`

11. CLIPS User's guide. Version 6.20, http://clipsrules.sourceforge.net/documentation/v624/ug.pdf
12. http://www.orafaq.com/wiki/SQLNet_FAQ
13. Mourlas, C., Germanakos, P.: Intelligent user interfaces: Adaptation and personalization systems and technologies. Information Science Reference (2009)
14. Nasoz, F., Lisetti, C.L.: Affective user modelling for adaptive intelligent user interfaces. In: HCI Intelligent Multimodal Interaction Environments, pp. 421–430 (2007)