Journal of Internet Banking and Commerce

An open access Internet journal (http://www.arraydev.com/commerce/jibc/)

Journal of Internet Banking and Commerce, December 2009, vol. 14, no.3
(http://www.arraydev.com/commerce/jibc/)

New Methodology for Distributed Real-Time Implementation of Telecommunications Architecture Based on Linux

Adil LEBBAT
Equipe Architectures des Systèmes, ENSEM, Université Hassan II Ain Chock.
Postal Address: **BP. 8118, Oasis Casablanca, Morocco**
Email: ***adil.lebbat@gmail.com***

Siham BENHADOU
Equipe Architectures des Systèmes, ENSEM, Université Hassan II Ain Chock.
Postal Address: **BP. 8118, Oasis Casablanca, Morocco**
Email: ***benhadou.siham@gmail.com***

Hicham MEDROMI
Equipe Architectures des Systèmes, ENSEM, Université Hassan II Ain Chock.
Postal Address: **. 8118, Oasis Casablanca, Morocco**
Email: ***hmedromi@yahoo.fr***

Mustafa RADOUI
Equipe Architectures des Systèmes, ENSEM, Université Hassan II Ain Chock.
Postal Address: **. 8118, Oasis Casablanca, Morocco**
Email: ***m_radoui@yahoo.fr***

Abstract

Our goal is to define an architecture to design a real-time distributed platform of network connection with an installation wizard and configuration of an ADSL modem, router and other telecommunication products under a GNU/Linux distribution, and an utility for managing ADSL connections. This architecture is based on the multi-agents. The design of this architecture is based on the concept of AUML

Keywords: **Distributed Architecture; Real Time; Multi-agent System; UML Agent; Linux; Kernel; Modem; Router; DSL modem**

© Adil LEBBAT, Siham BENHADOU, Hicham MEDROMI and Mustafa RADOUI, 2009

INTRODUCTION

An agent is a physical or virtual entity able to act in an environment that can communicate directly with other agents, driven by a set of trends (autonomy), which has its own resources, able to perceive its environment, which has skills and offers services that can possibly happen again, whose behavior tends to meet its objectives, taking into account the resources and skills available, and depending on its perception, its representations and communications received [1].

An agent possesses some or all of these features.

A multi-agent system is a set of autonomous agents in interaction. It helps coordinate the behavior of this group of agents to collectively solve a problem in a complex and evolving environment.

A multi-agent system is a set of intelligent agents interacting.

Agent → Autonomy
Intelligent → Goal, tasks
Interacting → Consideration of other distributed systems consisting of a set of agents

A multi-agent system can be:

• Open: agents enter and exit freely.
For example: a cafe, an e-commerce application.

• Closed: the set of agents remains the same.
For example: a football game, a multi-expert system.

• Homogeneous: all agents are built on the same model.
For example: a meeting, a colony of ants.

• Heterogeneous: agents of different models of different granularities.
For example: an ecosystem

• Mixed (or not): 'human' agents are part of the system. It is open and heterogeneous.
For example: a working group mediated by assistant's agents.

The oriented-agent programming has been proposed by Yoav Shoham in 1993 as a new programming paradigm.

In this approach, agents are the central elements of language.

Oriented-agent programming assumes that we will develop programs in which multiple agents interact, which focuses on the social dimension of agents.

	OOP	***OAP***
Basic Unit	Object	Agent
Settings defining the status of the base unit	no constraints	beliefs, decisions, requirements, skills
Calculation Process	sending messages and methods for response	sending messages and methods for response
Message Types	no constraints	inform, request, offer, promise, accept, reject, ..
Constraints on methods	no constraints	consistency, truth, ...

Figure.1 Differences between Object and Agent

PROPOSED ARCHITECTURE

Our goal is to build a distributed architecture based on multi-agents, able to solve configuration problems of ADSL modems, routers and other telecommunications products under a GNU/Linux distribution.

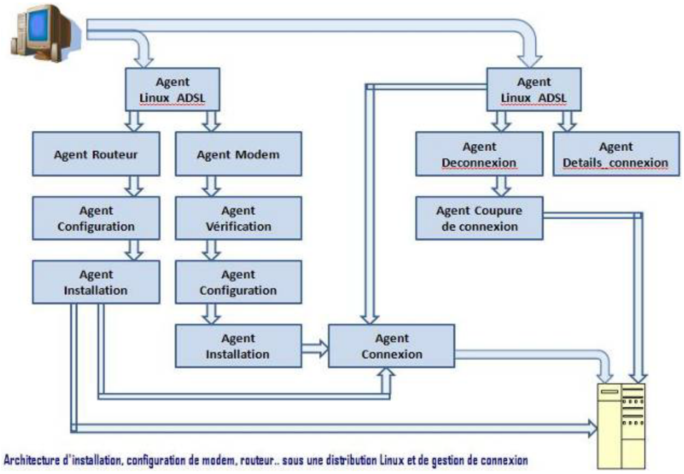


Figure.2 Proposed architecture

REALIZATION OF THE PLATFORM

✓ *Designing the Model*

In this section we propose a new design methodology based on the AUML language.

Agent UML is an extension of UML to take into account the agent notions. Agent UML inherits representations proposed by UML [3]. It thus contains ten types of diagrams symbolizing many different views to represent particular concepts of information system. They fall into two main groups:

Diagrams behavioral or dynamic charts:

1. Sequence diagrams
2. Collaboration diagrams
3. Activity diagrams
4. Statechart
5. Use case diagrams

Structural diagrams or static diagrams:

6. Class diagrams
7. Object diagrams
8. Packages
9. Component diagrams
10. Deployment diagrams

These diagrams are not necessarily all used at modeling.

The design of the proposed architecture is described through the two diagrams of use cases and classes of agents to illustrate the static aspect of the distributed platform developed.

✓ *Diagram of use case*

A representation in use case diagram is needed for the requirements specification. Several use cases were identified from the needs analysis:

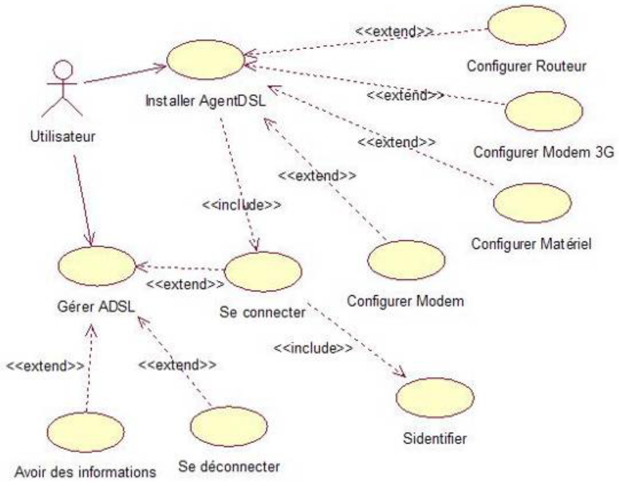


Figure.3 Use case diagram

✓ *Class agents diagram*

Initially, we look at the conceptual level that is high enough for the multi-agents system eliminating all surface information for understanding the structure of the system.

The class agents diagram in Figure 4 represents the conceptual level of the platform developed.

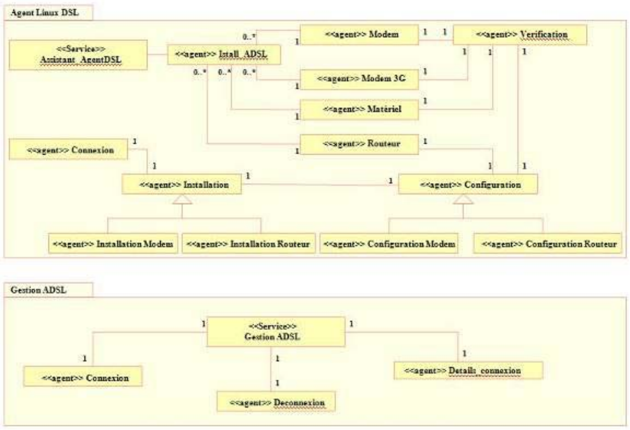


Figure.4 Class agents diagram (conceptual level)

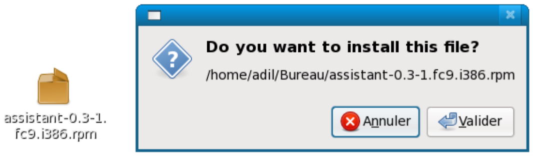
REALIZATION OF THE MODEL

We used the concepts explained in the previous sections to develop a demonstration application which implements this model.

We chose for our distributed development platform, open source solutions including:

- Fedora (GNU/Linux) as an Operating System
- GTK+ graphical library
- C and Bash Shell.
- RPM (Red Hat Package Manager) to package the application with its different components (programs, firmware,..) for an easy setup on the end-user OS.
- Supported products :
    - ADSL Modem: Sagem Fast 800 (chipset 1 to 4)
    - Router with a wired connection

In Figure 5 we can see the setup file (RPM package) and a running installation on Fedora (GNU/Linux).



Do you want to install this file?

/home/adil/Bureau/assistant-0.3-1.fc9.i386.rpm

Figure.5 Installing the application packaged in rpm

The installation is therefore without any user intervention, only a confirmation by pressing a button.

Then we will start our program from the menu that Fedora has been added automatically during the installation:

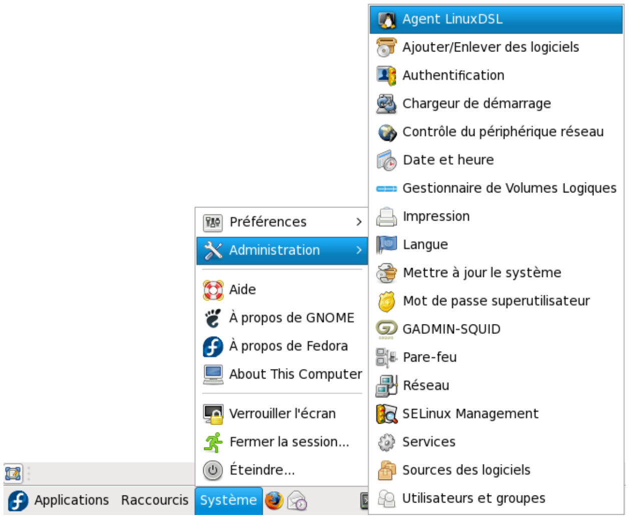
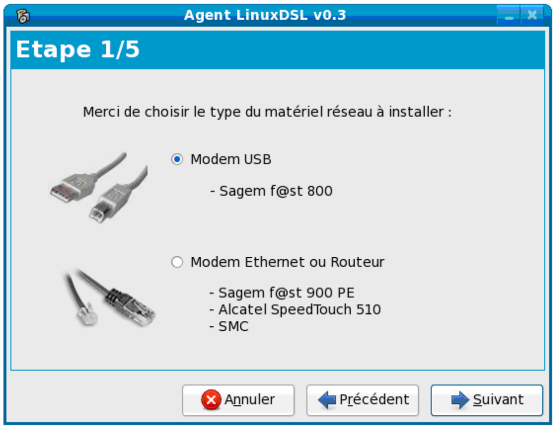


Figure.6 Launching the application through the menu

After the first screen, the application proposes to choose the type of device to install.



Merci de choisir le type du matériel réseau à installer :

● Modem USB

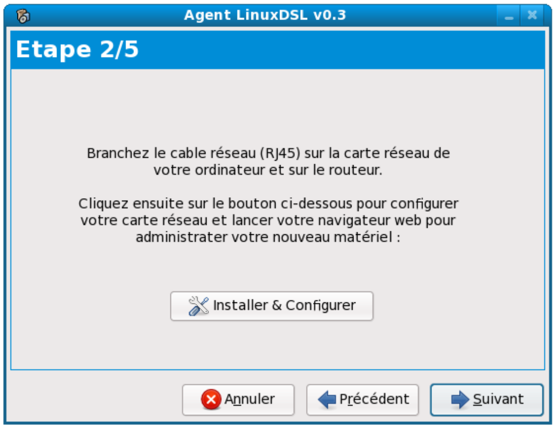
- Sagem f@st 800

○ Modem Ethernet ou Routeur

- Sagem f@st 900 PE
- Alcatel SpeedTouch 510
- SMC

Figure.7 Choosing the type of device to install

If you chose to install a router, the application proposes the configuration of the network card via DHCP since most routers have a DHCP server by default, and then launch the browser to administer the router via its web integrated application.

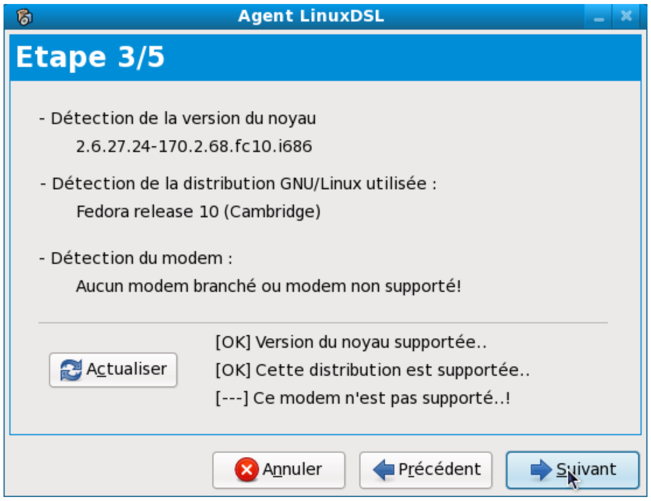


Branchez le cable réseau (RJ45) sur la carte réseau de votre ordinateur et sur le routeur.

Cliquez ensuite sur le bouton ci-dessous pour configurer votre carte réseau et lancer votre navigateur web pour administrater votre nouveau matériel :

Figure.8 Routers configuration

In the case of installation of an ADSL modem, the application detects and blocks the installation if the kernel version, GNU/Linux distribution or modems are not supported.



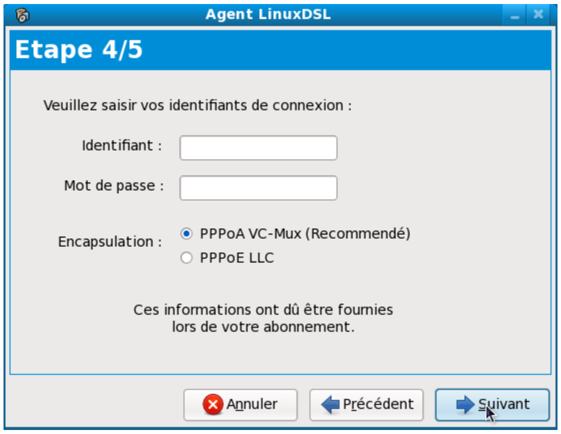
Etape 3/5

- Détection de la version du noyau
  2.6.27.24-170.2.68.fc10.i686
- Détection de la distribution GNU/Linux utilisée :
  Fedora release 10 (Cambridge)
- Détection du modem :
  Aucun modem branché ou modem non supporté!

[OK] Version du noyau supportée..
[OK] Cette distribution est supportée..
[---] Ce modem n'est pas supporté..!

Figure.9 Checking kernel, OS and modem

If the modem is connected and supported, the following screen appears:



Etape 4/5

Veuillez saisir vos identifiants de connexion :

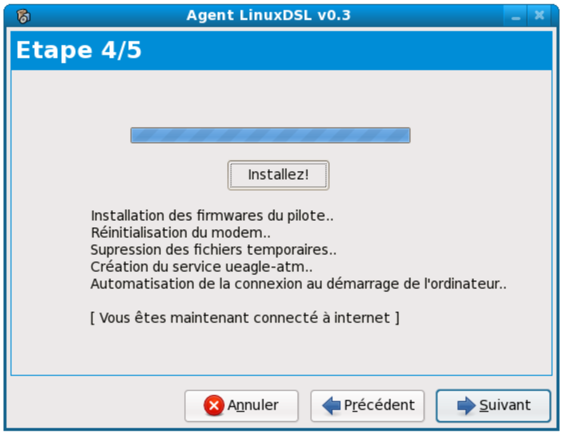
Identifiant :

Mot de passe :

Encapsulation : ● PPPoA VC-Mux (Recommendé)
○ PPPoE LLC

Ces informations ont dû être fournies
lors de votre abonnement.

Figure.10 Entering DSL Connection Informations



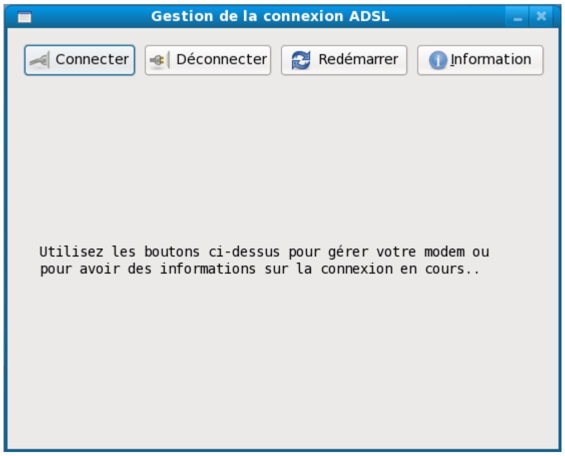
Etape 4/5

Installation des firmwares du pilote..
Réinitialisation du modem..
Supression des fichiers temporaires..
Création du service ueagle-atm..
Automatisation de la connexion au démarrage de l'ordinateur..

[Vous êtes maintenant connecté à internet]

Figure.11 Display performed tasks

The final screen is dedicated to the program launched just after the wizard to manage an ADSL connection:



```
Utilisez les boutons ci-dessus pour gérer votre modem ou
pour avoir des informations sur la connexion en cours..
```

Figure.12 ADSL Management

CONCLUSION

We have proposed a new distributed real-time architecture based on multi-agent systems that allow the installation and configuration of modem, routers and other telecommunications products on a GNU/Linux distribution and also the management of an ADSL connection. This new architecture designed to facilitate the installation tasks for an end user not familiar with shell commands.

REFERENCES

[1] J. Ferber. “Les systèmes multi-agents, vers une intelligence collective”. InterEditions, 1995.

[2] Weiss, G., (Ed.), 1999. Multiagent Systems. A Modern Approach to Distributed Artificial Intelligence, The MIT Press, Cambridge, Massachusetts.

[3] M. Huget, Une application d’AgentUMLau Supply Chain Management, JFIADSMA 2002

[4] A. Agarwal, K. B. Wang. “Supporting Quality of Service in IP multicast network”. Computer Communication, vol. 26, pp. 1533-1540, 2003.

[5] P. G. BACKES, K. S. TSO, G. K. THARP. “Mars Pathfinder Mission Internet-Based Operations Using WITS”. IEEE International Conference on Robotics and Automation ICRA, 1998.

[6] V. Braitenberg. “Vehicles: Experiments in synthetic psychology”. MIT press /Bradford Books, Cambridge MA, USA, 1984.

[7] R. Dargos, M. Collier. “An MPLS based architecture for differentiated Web service”, International Conference on Software Telecommunications and Computer Networks (IEEE - SoftCom2002), Split, Croatie, October 2002.

[8] D. McFarland, U. Bosser. “Intelligent behavior in animal and robots”. MIT press, Cambridge MA, USA, 1993.

[9] A. Sayouti, F. Qrichi Aniba, H. Medromi. “Remote Control Architecture over Internet Based on Multi agents Systems”. International Review on Computers and Software (I.RE.CO.S), November 2008.