Autonomous Network Equipments

Dominique Gaïti[1], Guy Pujolle[2], Mikaël Salaun[3], and Hubert Zimmermann[4]

[1] University of Troyes, UTT, 10010 Troyes, France
Dominique.Gaiti@utt.fr
[2] LIP6, UPMC, 8 rue du Capitaine Scott 75015 Paris, France
Guy.Pujolle@lip6.fr
[3] France Telecom, Lannion, France
Mikael.salaun@francetelecom.fr
[4] Ginkgo-Networks, 8 rue du Capitaine Scott 75015 Paris, France
Hubert.zimmermann@ginkgo-networks.com

Abstract. IP networks are now well established. However, control, management and optimization schemes are provided in a static and basic way. Network control and management schemes using an autonomy based technology offer a new way to master quality of service, security and mobility management. This new paradigm allows a dynamic and intelligent control of the equipment in a local manner, a global network control in a cooperative manner, a more powerful network management, and a better guaranty of all vital functionalities like end to end quality of service and security. In this paper, we provide a way to implement such a paradigm through the use of the agent and multi agent concept. A testbed of an architecture based on autonomous network equipment has been developed. This autonomous architecture is able to optimize the quality of service through the networks.

1 Introduction

The popularity of the Internet has caused the traffic on the Internet to grow drastically every year for the last several years. It has also spurred the emergence of the quality of service (QoS) for Internet Protocol (IP) to support multimedia application like ToIP. To sustain growth, the IP world needs to provide new technologies for guaranteeing quality of service. Integrated services and differentiated services have been normalized to support multimedia applications. The routers in the IP networks play a critical role in providing these services. The demand of QOS on private enterprise networks has also been growing rapidly. These networks face significant bandwidth challenges as new application types, especially desktop applications. Moreover, voice, video, and data traffic need to be delivered on the network infrastructure. This growth in IP traffic is beginning to stress the traditional software and hardware-based design of current-day routers and as a result has created new challenges for router design.

To achieve high-throughput and quality of service, high-performance software and hardware together with large memories were required. Fortunately, many changes in technology (both networking and silicon) have changed the landscape for implementing

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: 10.1007/978-3-540-32993-0_29

I. Stavrakakis and M. Smirnov (Eds.): WAC 2005, LNCS 3854, pp. 177 – 185, 2006.

high-speed network equipment. However, scalability problems were discovered with InterServ technologies and statistical problems with DiffServ. Moreover, these technologies are rather complicated to size and we assist to important configuration problems that need specialized engineers.

This paper proposes a new paradigm for providing a smart networking technique allowing a real time network configuration. Indeed, we propose to introduce an autonomy based technology within network equipments to configure themselves depending on the observed state of the network.

The rest of the paper is organized as follows. First, we introduce the autonomous paradigm and the implication on network equipment. Then, we introduce a new autonomy based architecture to support the deployment of the intelligent network equipment. Finally, we describe the agent architecture and we conclude this work.

2 The Autonomous Environment

As user needs are becoming increasingly various, demanding and customized, IP networks and more generally telecommunication networks have to evolve in order to satisfy these requirements. That is, a network has to integrate more quality of service, mobility, dynamicity, service adaptation, etc. This evolution will make users satisfied, but it will surely create more complexity in the network generating difficulties in the control process.

Since there is no control mechanism which gives optimal performance whatever the network conditions are, we argue that an adaptive and dynamic selection of control mechanisms, taking into account the current traffic situation, is able to optimize the network resources uses and to come up to a more important number of user expectations associated with QoS [0]. To realize such functionalities, it is necessary to be able to configure automatically the network in real time. Therefore, all the network equipment must be able to react to any kind of change in the network. Different techniques could be applied but as the most difficult moment is congestion, the technique has to be autonomic and network equipments have to turn into intelligent network equipments.

Autonomic communication paradigm has been mainly defined through the ACF (Autonomous Communications Forum) [1] and particularly as follows: Autonomic communication is centered on selfware – an innovative approach to perform known and emerging tasks of network control plane, both end-to-end and middle box communication based. Selfware assures the capacity to evolve, however it requires generic network instrumentation. Figure 1 outlines a generic framework of a network element that is enhanced by a selfware mechanism to exchange generic policies with groups of other elements and, through embedding of "policies to functionality" rules that control the behavior of an element. Selfware principles and technologies borrow largely from well established research on distributed systems, fault tolerance among others, from emerging research on non-conventional networking (multihop ad hoc, sensor, peer-to-peer, group communication, etc.), and from similar initiatives, like Autonomic Computing of IBM, XG of DARPA, Harmonious Computing of Hitachi, Resonant Networking of NTT, etc.

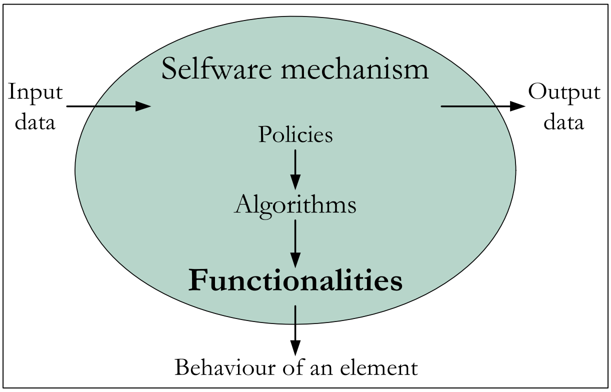

Fig. 1. Generic framework of a network element with a selfware mechanism

A visionary network would be able to (i) configure and re-configure itself, (ii) identify its operational state and take actions to drive itself to a desired stable state and finally (iii) organise the allocation and distribution of its resources. To build such a network, it is necessary to go beyond the improvement of techniques and algorithms by using a new concept, the knowledge plane. This concept was already proposed for managing the Internet. The knowledge plane is able to collect all information available in the network to provide the other elements of the network with services and advice and make the network perform what it is supposed to. There are many objectives to the configuration and reconfiguration of the network, from the optimisation of resources to the use of best available techniques in order to offer the most appropriate service, best adapted to the terminal capabilities.

The network architecture proposed in this paper aims at defining a functional architecture for the interconnection and interoperability of the different autonomous elements (i.e. network equipments as routers, firewall, middle box, etc.) interconnected to form a multiservice network. The architecture has to take into account different aspects for autonomy:

Self-configuration: the autonomic network elements must be able to configure themselves once into the network domain. Self-configuration includes such aspects as IP address, security, QoS among others. Self-configuration should also deal with the technology handover (e.g. going from Wi-Fi to UMTS) and with the parameterisation of each technology to obtain the optimal resource usage and interaction.

Self-management: the autonomic network must be able to self-manage in order to ensure a stable operational state. Whenever a new service must be deployed or a new terminal comes into the network, the self-management functions must drive the network to a stable operational state. This state would be calculated to be optimal with respect to the current operational conditions and the requirements of all available services within the available resources.

Self-diagnostics: the network as a whole must be able to identify its operational state and take action to drive itself to a desired stable state. The network must be able to identify the users accessing the service domain and recognise their profiles including the rights and associated parameters. Finally, an autonomous network consisting of heterogeneous home appliances designed for functions ranging from high complex decoding of video and audio signals to vacuum cleaning must be able to manage the interaction of their interoperation (e.g. interference from one appliance to the others) as well as precedence and priorities.

Self-protection: an autonomous network must be able to identify security threats to the content being carried or treated within the network, such as intrusions or denial of service attacks among others. An autonomous network must take appropriate action to protect itself against such threats and must ensure a transparent experience for the user.

Self-organisation: the autonomous network must be self-organised as regards resource allocation and distribution. Resources should be automatically allocated where necessary or appropriate for the current operational status and service configuration. In addition, taking into account the computational resources available in the network and the different computational grids that can be dynamically formed, the autonomic network must be able to self-organise in an optimal and secure way.

3 The 4-Plane Architecture

The 4-plane architecture approach [1] our proposal is relying on is described in Figure 2. Our proposal does not aim at proposing new algorithms or new schemes in the control plane but rather at selecting the best algorithms and the best values of the parameters at any time to reach the objective (network security, QoS, mobility management, resource optimization, etc.).

This approach will allow reconfiguring the different network elements (routers, switches, mobile elements, firewall, set-up-box, and middle-box) in quasi-real time. The goal of this approach is to secure the network, optimize the performance and control the mobility within the network. This driving process runs in real time and reconfiguration can occur several times per second if necessary. This compares to the configuration schemes used today where networks are configured only at set-up time, the configuration being decided on an average behavior of the network.

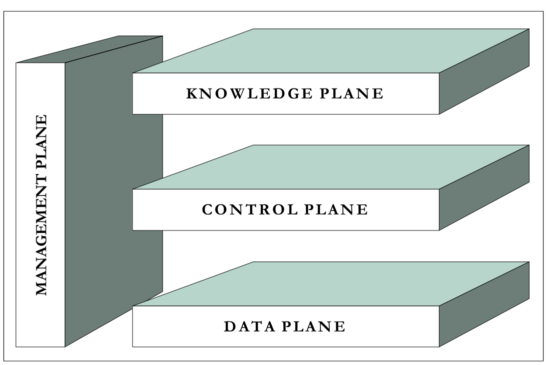

Fig. 2. The 4-plane architecture

The idea lying behind the knowledge plane is to locate our agents and the knowledge they need to act and to help the reactive agents to make the best decisions, take the most appropriate actions in view of attaining the goals set forth. Different types of implementation of the knowledge plane can be provided:

- The knowledge plane composed of meta agents with cognitive intelligence,
- The knowledge plane composed of a Policy Decision Point (PDP) or a set of Local PDPs [2],
- The knowledge plane composed of a supervisor,

A mixed of these different schemes can also offer a solution.

4 Adaptive and Autonomous System

Concerning the implementation of the autonomous system described in the previous section, a multi-agent approach could be a solution. In fact, agents own some features like autonomy, proactivity, cooperation, etc. predisposing them to operate actively in a dynamic environment like IP networks. Agents, by consulting their local knowledge and by taking into consideration the limited available information they possess about their neighbors, select the most relevant management mechanisms to the current situation.

A multi-agent system is composed of a set of agents which solve problems that are beyond their individual capabilities [3]. Multi-agent systems have proven their reliability when being used in numerous areas like: (1) the road traffic control ([4], [5]); (2) biologic phenomena simulation like the study of eco-systems [6] or the study of ant-colonies [7], for example; (3) social phenomena simulation like the study of consumer behaviors in a competitive market [8]; (4) industrial applications like the control of electrical power distribution systems, the negotiation of brands, etc. By its nature, multi-agent approach is well suited to control distributed systems. IP networks are good examples of such distributed systems. This explains partly the considerable contribution of agent technology when introduced in this area. The aim was mainly to solve a particular problem or a set of problems in networks like: the discovery of topology in a dynamic network by mobile agents ([9], [10]), the optimization of routing process in a constellation of satellites [11], the fault location by ant agents [12], and even the maximization of channel assignment in a cellular network [13].

Our approach consists in integrating agents to build an autonomous environment. These agents optimize the network QoS parameters (delay, jitter, loss percentage of a class of traffic, etc.), by adapting the activated control mechanisms in order to better fit the traffic nature and volume, and the user profiles. The agents share a global goal of the network through the knowledge plane. Agents may be reactive, cognitive or hybrid [3], [6], [14]. Reactive agents are suitable for situations where we need less treatment and faster actions. Cognitive agents, on the other side, allow making decisions and planning based on deliberations taking into account the knowledge of the agent about itself and the others. A hybrid agent is composed of several concurrent layers. In INTERRAP [15], for example, three layers are present: a reactive layer, a local planning layer, and a cooperative layer.

The approach we propose is different [16], [17], [18], [19]. In fact, every node has one cognitive agent that supervises, monitors, and manages a set of reactive agents.

Each reactive agent has a specific functioning realizing a given task (queue control, scheduling, dropping, metering, etc.) and aiming to optimize some QoS parameters. The cognitive agent (we call it Master Agent) is responsible for the control mechanisms selection of the different reactive agents, regarding the current situation and the occurring events. By using such an architecture, we aim to take advantage of both the reactive and cognitive approaches and avoid shortcomings of the hybrid approach (coordination between the different layers, for instance).

To get the agent-based autonomous approach, we propose to select the appropriate control mechanisms among:

- adaptive: the agent adapts its actions according to the incoming events and to its vision of the current system state. The approach we propose is adaptive as the agent adapts the current control mechanisms and the actions undertaken when a certain event occurs. The actions the control mechanism executes may become no longer valid and must therefore be replaced by other actions. These new actions are, indeed, more suitable to the current observed state [20];
- distributed: each agent is responsible for a local control. There is no centralization of the information collected by the different agents, and the decisions the agent performs are in no way based on global parameters. This feature is very important as it avoids having bottlenecks around a central control entity;
- local: the agent executes actions on the elements of the node it belongs to. These actions depend on local parameters. However, the agent can use information sent by its neighbors to adapt the activated control mechanisms;
- scalable: our approach is scalable because it is based on a multi-agent system which scales well with the growing size of the controlled network. In order to adaptively control a new node, one has to integrate an agent (or a group of agents) in this node to perform the control.

Our model relies on two levels:

At level 0, we find the different control mechanisms of the node, which are currently activated. Each control mechanism is characterized by its own parameters, conditions and actions, which can be monitored and modified by the Master Agent. Some of the proposed management mechanisms are inspired from known algorithms but have been agentified in order to optimize the performance and to improve cooperation between agents.

Different agents belong to this level (Scheduler Agent, Queue Control Agent, Admission Controller Agent, Routing Agent, Dropping Agent, Metering Agent, Classifying Agent, etc.). Each of these agents is responsible for a specific task within the node. So each agent responds to a limited set of events and performs actions ignoring the treatments handled by other agents lying on the same node or on the neighborhood. This allows the agents of this level to remain simple and fast. More complex treatments are indeed left to the Master Agent.

At level 1, is lying a Master Agent responsible for monitoring, managing, and controlling the entities of level 0 in addition to the different interactions with the other nodes like cooperation, negotiation, messages processing, etc. This agent owns a model of its local environment (its neighbors) that helps him to take its own decisions. The Master Agent chooses the actions to undertake by consulting the current state of

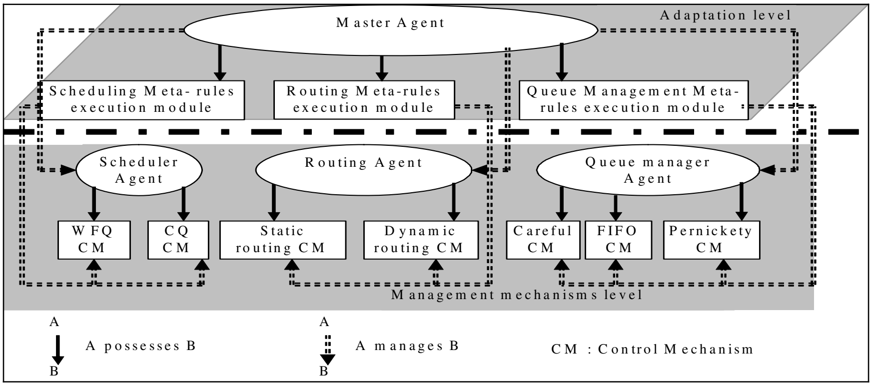

Fig. 3. Two levels of decision within the node

the system (neighbors nodes state, percentage of local loss, percentage of its queue load, etc.) and the meta-rules at its disposal in order to have only the most relevant control mechanisms activated with the appropriate parameters. The node, thanks to the two decision levels, responds to internal events (loss percentage for a class of traffic, load percentage of a queue, etc.) and to external ones (message sent by a neighbor node, reception of a new packet, etc.).

The Master Agent owns a set of meta-rules allowing it to decide on actions to perform relating to the different node tasks like queue management, scheduling, etc. (see Figure 3). These meta-rules permit the selection of the appropriate control mechanisms to activate the best actions to execute. They respond to a set of events and trigger actions affecting the control mechanisms supervised by that Master Agent. Their role is to control a set of mechanisms in order to provide the best functioning of the node and to avoid incoherent decisions within the same node. These meta-rules give the node the means to guarantee that the set of actions executed, at every moment by its agents, are coherent in addition to be the most relevant to the current situation.

The actions of the routers have local consequences in that they modify some aspects of the operations of the router (its control mechanisms) and some parameters of the control mechanisms (queue load, loss percentage, etc.). However, they may influence the decisions of other nodes. In fact, by sending messages bringing new information on the state of the sender node, a Master Agent meta-rule on the receiver node may fire. This can involve a change within the receiver node (the inhibition of an activated control mechanism, or the activation of another one, etc.). This change may have repercussions on other nodes, and so forth until the entire network becomes affected.

This dynamic process aims to adapt the network to new conditions and to take advantage of the agent abilities to alleviate the global system. We argue that these agents will achieve an optimal adaptive control process because of the following two points:

(1) each agent holds different processes (control mechanisms and adaptive selection of these mechanisms) allowing to take the most relevant decision at every moment;

(2) the agents are implicitly cooperative in the sense that they own meta-rules that take into account the state of the neighbors in the process of control mechanisms selection. In fact, when having to decide on control mechanisms to adopt, the node takes into consideration the information received or guessed from other nodes.

5 Development

Ginkgo-Networks company is developing such an architecture integrating intelligent software agents. The technology developed by Ginkgo-Networks is unique because it is linked to a double skill almost non-existent today coming from the field of Artificial Intelligence and networks (the use of intelligent agents for the control and the management of the network). These concepts allow (1) a dynamic and intelligent control of the equipment in a local manner, (2) a global network control in a cooperative manner, (3) a more autonomous network management, and (4) a better warranty of the quality of service in an end to end manner. Thus, Ginkgo-Networks Company provides a solution where no equivalent solution on the market allows for the optimal functioning of the network.

Results of the first testbebs are very convincing. However, the gain depends on the integration of the agents inside the equipment or outside the equipment. We showed in our testbed that if the agents are inside the equipment, the optimal performances are obtained when configuring between every second and hundreds of millisecond. On the contrary, when the agents are implemented outside the equipment (in our testbed outside Cisco routers) the optimum is obtained when reconfiguration take place between one minute and several minutes. The gain in performance with Linux routers and inside agents could be between 10 and 50 %. All these results should appear in a future paper.

6 Conclusion

This paper introduced new implicit communication architecture to better support QoS and new functionalities using the autonomic communication paradigm. A knowledge plane that allows the agents to share a global goal of the overall network introduces this paradigm. Intelligent network equipments are self-configurable using an agent-based control scheme. This architecture and the associated protocols consider not only the policies provided by the business plan but also the constraints of the lower layers of the network. A 4-plane architecture was proposed in the autonomic communication community which help us to provide the selection of control mechanisms to optimize the configuration of the routers and of the protocols. This architecture interacts with the network equipment and protocols in order to configure the network with the selected protocols and parameters. An analysis of our architecture shows that a real time configuration of routers is available and brings an important improvement of the performance. Our proposal has been tested in a simulation environment and gave very good results in terms of delay, and lost packet reduction. Then the agent infrastructure has been implemented in a real environment composed of 9 different routers.

References

1. Castro M., Merghem L., Gaiti D., Mhamed A. – The Basis for an Adaptive IP QoS Management", Special Issue: Internet Technology IV, IEICE Transactions on Communications, vol.E87-B n°3, pp. 564-572, March 2004
2. Proceedings of the first WAC (Workshop on Autonomic Communications), Berlin, October 2004.

3. Verma D. C. – Simplifying Network administration using policy-based management, IEEE Network 16(2), 2002.
4. Ferber J. – Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence, Addison Wesley Longman, 1999.
5. Bazzan A.L.C., Wahle J. and Klügl F. – Agents in Traffic Modelling - From Reactive to Social Behaviour, KI’99, LNAI 1701, pp 303-307, Bonn, Germany, September 1999.
6. Moukas A., Chandrinos K. and Maes P. – Trafficopter A Distributed Collection System for Traffic Information, CIA'98, Paris, France, LNAI 1435 pp 34-43, July 1998.
7. Doran J. – Agent-Based Modelling of EcoSystems for Sustainable Resource Management, 3rd EASSS’01, Prague, Czech Republic, LNAI 2086, pp 383-403, July 2001.
8. Drogoul A., Corbara B. ad Fresneau D. – “MANTA: New experimental results on the emergence of (artificial) ant societies" in Artificial Societies: the computer simulation of social life, Nigel Gilbert & R. Conte (Eds), UCL Press, London, 1995.
9. Bensaid L., Drogoul A., and Bouron T. – Agent-Based Interaction Analysis of Consumer Behavior, AAMAS’2002, Bologna, Italy, July 2002.
10. Minar N., Kramer K.H. and Maes P. – Cooperating Mobile Agents for Dynamic Network Routing. in “Software Agents for Future Communication Systems”, Chapter 12, *Springer Verlag*, pp 287-304, 1999.
11. Roychoudhuri R., et al. – Topology discovery in ad hoc Wireless Networks Using Mobile Agents. MATA'2000, Paris, France, LNAI 1931, pp 1-15. September 2000.
12. Sigel E., et al. – Application of Ant Colony Optimization to Adaptive Routing in LEO Telecommunications Satellite Network, Annals of Telecommunications, vol.57, no.5-6, pp 520-539, May-June 2002.
13. White T. et al. – Distributed Fault Location in Networks using Learning Mobile Agents, PRIMA'99, Kyoto, Japan. LNAI 1733, pp 182-196. December 1999.
14. Bodanese E.L. and Cuthbert L.G. – A Multi-Agent Channel Allocation Scheme for Cellular Mobile Networks, ICMAS’2000, USA, IEEE Computer Society press, pp 63-70, July 2000.
15. Wooldridge M. – Intelligent Agents. In « Multiagent Systems : a Modern Approach to Distributed Artificial Intelligence », Weiss G. Press, pp 27-77, 1999.
16. Müller J.P and Pischel M. – Modelling Reactive Behaviour in Vertically Layered Agent Architecture. ECAI’94, Amsterdam, Netherlands, John Wiley & Sons, pp 709-713, 1994.
17. Merghem L., Gaïti D. and Pujolle G. – On Using Agents in End to End Adaptive Monitoring, E2EMon Workshop, in conjunction with MMNS’2003, Belfast, Northern Ireland, LNCS 2839, pp 422-435, September 2003.
18. Gaïti D., and Pujolle G. – Performance management issues in ATM networks: traffic and congestion control, IEEE/ACM Transactions on Networking, 4(2), 1996.
19. Gaïti D. and Merghem L. – Network modeling and simulation: a behavioral approach, Smartnet conference, Kluwer Academic Publishers, pp. 19-36, Finland, April 2002.
20. Merghem L. and Gaïti D. – Behavioural Multi-agent simulation of an Active Telecommunication Network, STAIRS 2002, France. IOS Press, pp 217-226, July 2002.
21. Pujolle G., Chaouchi H., Gaïti D. – Beyond TCP/IP : A Context Aware Architecture, Kluwer Publisher, Net-Con 2004, Palma, Spain, 2004.