Towards a global autonomic management and integration of heterogeneous networks and multimedia services

Anasser Ag Rhissa and Adil Hassnaoui

GET INT, CNRS Samovar
Institut National des Télécoms, 9 Rue charles Fourier 91011 Evry (FRANCE)
anasser.ag-rhissa@int-evry.fr, adil.hassnaoui@int-evry.fr

Abstract. Traditionally, networks and systems are manually managed. It usually takes one or more human operators to manage all aspects of a dynamically evolving computing and communicating system. The operator is tightly integrated in this management process, and his tasks range from defining high-level policies to executing low-level system commands for immediate problem solving. Although this form of human-in-the-loop management was appropriate in the past, it has become increasingly unsuitable for modern networked computing systems and telecommunication. The potential advantage that autonomic computing brings is reducing the cost and complexity of managing Information and Communication Technology Infrastructure(ICT).
The objectives of this paper are to underline the characteristics of autonomic architectures and present an outline of our autonomic management architecture based on OGSA (Open Grid Services Architecture) and Peer-to-Peer model. The autonomic management architectures of CISCO and IBM are briefly described and compared with our autonomic management architecture.

Keywords : Autonomic Computing, Grid computing, OGSA, Self-management, Autonomic Management, Peer-to-Peer, Global QoS management.

1 Introduction

Autonomic computing is a new paradigm with a goal to give systems the ability to manage themselves and dynamically adapt to change in accordance with business policies and objectives. Self-managing systems can perform management activities based on situations they observe or sense in the ICT (Information and Communication Technology) environment.

Like their biological origins, autonomic systems will maintain and adjust their operations in the face of changing components, workloads, demands and hardware or software failures. The autonomic system might continually monitor its own use, check for component upgrades for example and reconfigure itself if necessary. When it detects errors, the system will revert to the older version

while its automatic problem-determination algorithms try to isolate the source of the error.

Nowadays in most of management systems the adaptation to change situations in accordance with business policies are not autonomic, and they don't generally manage themselves. In order to do so, autonomic architectures are needed.

This paper is organized as follow. After an introduction, the second section introduces the characteristics of autonomic architectures and the link with OGSA. The third section introduces our autonomic management architecture based on Peer-to-Peer model. A comparison between our autonomic architecture, OGSA and the autonomic architectures of IBM and CISCO is made in the fourth section. Conclusion and perspectives are given in the last section.

2 Control loop and characteristics of autonomic architectures.

An autonomic system is made of a connected set of autonomic elements that contain resources and deliver services to humans and other autonomic elements. Autonomic elements will manage their internal behaviors and their relationships with other autonomic elements in accordance with policies that humans or other elements have established [7].

Autonomic architecture consists of a set of systems that are self-configuring (with autonomic configuration and adjustment), self-healing (with autonomic detection, diagnosis and repair of local problems), self-protecting (with autonomic protection and anticipation of problems) and self-optimizing (with autonomic improvement of performance and efficiency) [6].

The role of autonomic element consists on providing its services and managing its own behavior. To do so autonomic element monitors behavior through sensors, analyzes those data, then planes what action(s) should be taken, and executes that (those) action(s) through effectors. That creates a control loop [7] which allows to manage the systems (see figure 1).

The biggest challenge in an autonomic architecture is to build closed control loops, the most important concept of self-management.

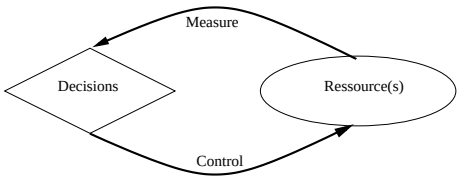


Fig. 1. control loop.

Autonomic computing proposes a solution for self-management system based on a service-oriented architectural approach such as Web services or OGSA infrastructure. OGSA combines web services and grid computing with open interfaces, it can be seen [5] as an extension and a refinement of the emerging Web Services architecture. By combining these two approaches (autonomic computing and OGSA), the autonomic computing profits from the advantages of OGSA such as computational capacity, virtualization, higher QoS, great availability and allows to integrate service mobility in management operations.

3 PARIS: our generic and autonomic management architecture

3.1 Overview of our architecture

At GET INT, the research works related to AGIRS [1] [2] [3] [4] has designed a generic architecture for the autonomic management of the heterogeneous networks and services, named PARIS (Platform for the autonomic Administration of netwoRks and Integration of multimedia Services). As depicted in figure 2, this architecture is divided into three generic classes.

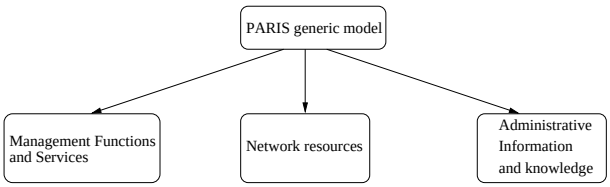


Fig. 2. Overview of PARIS generic model

The main components of PARIS are:

- *Administrative information :* This information shows the management resources use, like the services profile, the managers availability and the state of the managed resources. Therefore, this class helps the administrators to manage complex networks by providing strategic information to the organization and by defining management policies, in order to provide them a dynamic network management.
- *Management functions and services :* This class gathers all the necessary resources only for management.
- *Network resources :* This class represents the resources which are managed by the services of management system.

The components of PARIS are organized in three-layers. The bottom level represents physical devices such as switches, routers and hosts, as well as logical services such as VLANs, IP networks, file servers, and web services.

The medium level represents the autonomic management level which gathers all the necessary resources for autonomic management services. The top level is dedicated to SLS (Service Level Specification) and administrative information.

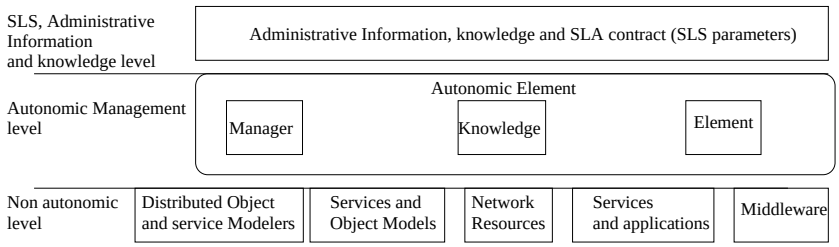


Fig. 3. Overview of layering view of PARIS architecture

In the bottom level (see figure 3), composed with non autonomic resources, services, applications and systems, we use OGSA for virtualisation, self-healing, computational and middleware capabilities. So, for the autonomic level (level 2) all the resources of layer 3 are considered as services and are transparents.

In order to deliver an integrated service to customers the different interconnected service providers must cooperate through their management domains using their business policies and objectives.

3.2 Global QoS policy based network and services autonomic management

Our QoS criterias are flexibility, scalability, safety, delay, jitter, mainly availability and survivability. According to the comparison (table 1) between P2P and hierarchical architectures, we have choosen an hybrid architecture for our autonomic management architecture.

Criterias / Architecture	P2P	Hierarchical
Response time	Slow/Medium	Fast
Survivability, Availability, Reliability	High	Low/medium
Scalability, Flexibility, Safety	High	low
Load for policy exchange	High	low
Manager between domains	No	Yes
Organization	Dynamic	Static

Table 1. Comparison between P2P and Hierarchical architectures.

The global QoS Policy management is based on a peer-to-peer approach (Peer-to-Peer QoS cooperation) between different operators' policy domains and a hierarchical approach in an operator's policy domain. An end-to-end QoS negotiation will take place to achieve the global business and policy goals.

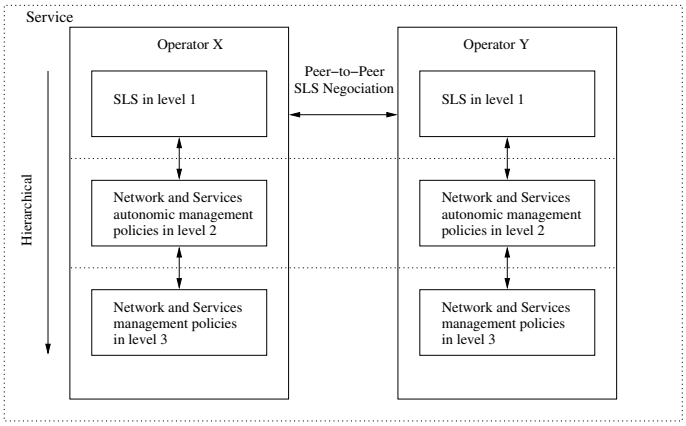
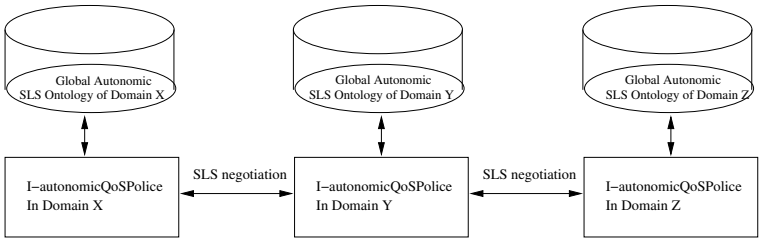


Fig. 4. Global QoS policy based services and network management.

The figure 4 shows an overview of the global QoS policy based services and network management. It represents the peer-to-peer and hierarchical approach management. The following figures will describe these approaches in more details. In an operator's domain, management functions are organized in three levels. The top level contains global management policy and SLS parameters to negotiate with other operators. Once the two operators agreed, The SLA is transmitted to the second level (autonomic level) for enforcement then to the third level (non-autonomic level).



I–autonomicQoSPolice : Inter–domains autonomic QoS policy agent/manager.

Fig. 5. Peer-to-Peer QoS policy cooperation between different operators Domains.

The figure 5 highlights the QoS policy cooperation between the *Inter-domains-autonomicQoSPolicy Agents/Managers* of each operator domain : Each *I-autonomicQoSPolice* in one domain negotiates SLS parameters with other peer domains according to the global QoS Objectives.

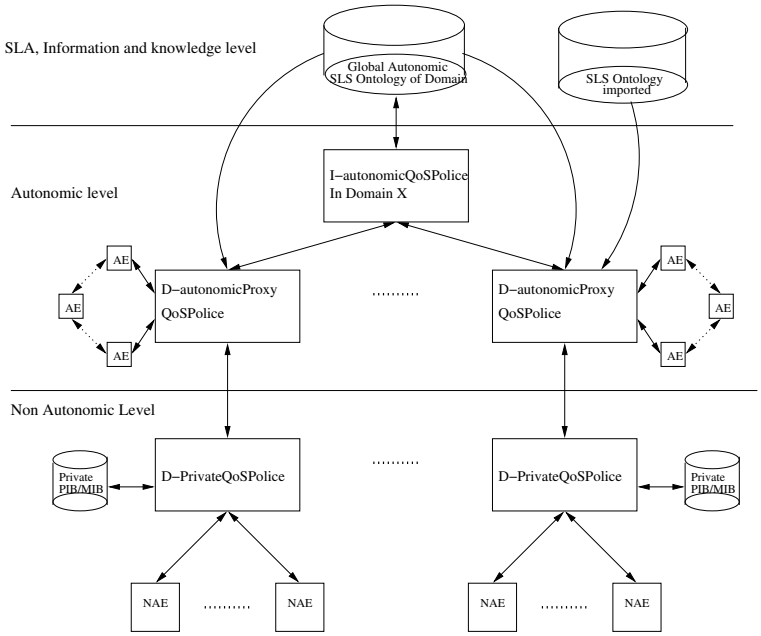


Fig. 6. Hierarchical QoS policy cooperation in an operator Domain.

In the hierarchical approach (see figure 6) we distinguish clearly the three levels of an operator's domain. The *I-autonomicQoSPolice* interacts with the SLS ontology in the Administrative information and SLS level to get the global QoS policy and manage the *Domains-AutonomicProxyQoSPolicy Agent/Manager*. Each *D-autonomicProxyQoSPolice* can recover its own QoS policy from the SLS ontology of the operator's domain and transmit it to Autonomic Element (AE) which manage themselves. By the same way the *D-autonomicProxyQoSpolice* allows to manage the Non-Autonomic Element (NAE) by using the *Domain-PrivateQoSPolicy Agent/Manager* of each sub-domain (i.e SNMP Domain, TMN Domain...) which recover the policy management information from its private PIB/MIB(Policy Information Base/Management Information Base).

This way, the hierarchical approach allows an effective QoS policy cooperation in the domain and limits the fault management propagation and topology changes.

3.3 Semantic negotiation using SLS ontology

In this scenario, we will consider a virtual web hosting, on our P2P architecture, in which clients negotiate their services parameters using a web services ontology (specification of the conceptualization as a hierarchy of concepts).

The web services ontology used contains a generic part about web services standard characteristics and a specific (local) one. For example, in the generic part, a web service belongs to a community service which is provided by a (or a set of) service provider(s). The QoS (Quality of Service) provided to the clients could be one or many of the following SLS parameters: availability, security, reliability,..., survivability. In the local part of this ontology, we have web hosting specific parameters such as operating system, transfer rate and storage disk space. A file SLAC(Service level Agrement Configuration) is used by the clients to negotiate their SLA contract using policies.

4 comparison between autonomic management architectures

In this section, we compare (see table 2) the autonomic computing Initiative (ACI) of IBM, the Adaptive Service Framework (ASF) of CISCO and the Open Grid Service Architecture (OGSA) with our architecture PARIS.

4.1 Comparison between ASF, ACI & OGSA.

The autonomic computing Initiative (ACI) of IBM is based on the control loop and the four area of self-management. Cisco and IBM, made the decision to collaborate on an Adaptive Services Framework (ASF) [8] based on the Adaptive Network Care (ANC) of CISCO and the Autonomic Computing Initiative(ACI) of IBM [7].

ASF is a set of proposed interfaces and formats that allow customers to interact with service providers.

The SSP (Support Service Provider) acts as a proxy (mediation gateway) to achieve the actions of the autonomic manager for integrating multiple vendors services.

ASF (CISCO/IBM) and ACI (IBM) architectures are based on service oriented architecture and they use similar standards to develop Web services. However the ASF framework proposes five levels of security (Authentication, Authorization, Encryption, Data Privacy, Signature) contrary to ACI and OGSA (which represents several gaps of security).

In table 2, it appears that our autonomic management architecture PARIS is more suitable to take into account business needs of ICT and telecom managers, in term of global governance of their information systems, to support semantic and autonomic negotiation of configuration and services parameters and to permit self-organization in an operator's peer domain by using shared administrative information, ontology and self-governing capabilities of autonomic elements.

Criterias / Architecture	OGSA(GGF)	ACI(IBM)	ASF(CISCO)	PARIS (INT/AGIRS)
Self-configuring	-	+	+	+
Self-healing	+	+	+	+
Self-protecting	-	+/-	+	+/-
Self-optimizing	-	+	+	+
Services oriented Architecture and virtualization	+	+	+	+
Taking into account business needs of ICT and telecom managers: Global governance	-	-	-	+
Taking into account mobility and nomadisme	+/-	+/-	+/-	+/-
Complete self-organization, dynamic and end-to-end Qos management	-	+/-	+/-	+/-
Interface with non autonomic environment and complete integration	+/-	-	+	+/-
Semantic and automatic negotiation of configuration and services parameters	-	-	-	+

Table 2. Comparison between OGSA and autonomic management architectures.

4.2 Advantages of our autonomic management architecture

The global P2P management architecture in our administrative information and SLS layer supports *concurrent* multi-manager control of network elements. The regrouping of manager-element roles improves *safety* by eliminating the state synchronization problem between managers and elements. The replacement of management agents by Autonomic Management Elements improves reliability through reductions in the size and complexity of implementing managed network services. The P2P management architecture also provides scalable monitoring and control of network elements. Management functions can be safely distributed across multiple managers due to the protection of transactional concurrency control. The unification of the manager and element roles in a peering relation enables the delegation of management functions, effectively distributing management load and supports *self-healing* in the face of local network failures.

This new peer-to-peer architecture benefits from the advantage of both approaches, autonomic computing and peer-to-peer, in order to allow an autonomic and dynamic management and to provide to the user a service with a satisfactory quality of service (availability).

5 Conclusion and Prospective work

Current network management functions will not be able to support the growing of networked devices and complex dependencies created by new web-based services architectures. The proposed peer-to-peer autonomic management architecture offers several advantages over the traditional manager-agent (client-server) architecture by creating a flexible, scalable, reliable and survivable environment supporting safe multi-manager access. The unification of the traditional roles of manager and element allows management functions to be distributed in different elements supporting autonomic behavior.

In the future we plane to highly distribute a P2P repository of our architecture to support scalable operations as well as recovery after failures. Future research will determine the granularity of distribution (service, node, Autonomic Element...), will extend the security and mobility management aspects and will details the complete integration of non-autonomic devices, such as hubs, switches, etc. The other points of our research will be to define exactly how autonomic elements interact between themselves to allow a cooperation and learning in autonomic environment and how to make possible a complete self-organization in autonomic environments of extended operators, virtual organizations and enterprises.

References

1. A. Ag Rhissa, AGIRS project, Web technologies and information systems, Scientific meeting of GET at ENST Paris, october 14, 2004.
2. A. Ag-Rhissa, A. Hassnaoui, Global self-management of network and telecommunication information systems and services. IEEE/SITIS'05, November 27th - December 1st, 2005.
3. F. Benayoune et L. Lancieri, Models of Co-operations in Peer-to-Peer Networks-A Survey , 3rd European Conference on Universal Multiservice Networks, Vol. 3262: 327-336, 2004.
4. F. Benayoune, Adaptive management of content services for new generations of mobile networks, ongoing work of PhD thesis, Directors of thesis: A. Ag Rhissa et P. Vincent, INT/AGIRS and France Télécom R&D Caen, 2004.
5. I. Foster and C. Kesselman and J. Nick and al., The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration, Globus Project, 2002.
6. K. Herrmann, G. Muhl and K. Geihs, Self Management: The Solution to Complexity or Just Another Problem?, in the IEEE DISTRIBUTED SYSTEMS ONLINE, Vol. 6, No. 1, 2005.
7. J. Kephart and D.M. Chess., The vision of autonomic computing, in the IEEE Computer Journal, Vol. 36: 41-50, 2003.
8. T. Studwell and K. Sankar, Adaptive Services Framework, Wd-asf-1.00, 2003.