



**HAL**  
open science

# VERS UNE PERSONNALISATION DES SERVICES RÉSEAU À LA DEMANDE

Gladys Diaz

► **To cite this version:**

Gladys Diaz. VERS UNE PERSONNALISATION DES SERVICES RÉSEAU À LA DEMANDE. Réseaux et télécommunications [cs.NI]. Université Pierre et Marie Curie - Paris 6, 2016. tel-02055306

**HAL Id: tel-02055306**

**<https://sorbonne-paris-nord.hal.science/tel-02055306>**

Submitted on 3 Mar 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Public Domain



**VERS UNE PERSONNALISATION DES SERVICES RÉSEAU À LA  
DEMANDE**

par  
**Gladys Diaz**

MÉMOIRE  
**d'Habilitation à Diriger des Recherches**

UNIVERSITÉ PIERRE ET MARIE CURIE

Soutenu le 28 Juin 2016

MEMBRES DU JURY

---

Francine KRIEF	Université de Bordeaux, Professeur	Rapporteur
Harry PERROS	NC State University, Professeur	Rapporteur
Michelle SIBILLA	IRIT, Professeur	Rapporteur
Christophe CÉRIN	Université Paris 13, Professeur	Examineur
Yacine GHAMRI-DOUDANE	Université de La Rochelle, Professeur	Examineur
Guy PUJOLLE	UPMC, Professeur	Examineur
Noémie SIMONI	Télécom ParisTech, Professeur	Examineur

---







# Résumé

Mes domaines de recherche couvrent le déploiement dynamique de services réseau, en intégrant deux aspects : les systèmes d'information et la QoS pour la gestion de réseaux et les architectures orientées services (SOA). J'ai pu traiter ces différentes thématiques à travers l'application de :

- l'User-Centric et les réseaux superposés dans le cadre du projet UBIS (*User Centric – uBiquité et Intégration de Services*),
- la virtualisation des réseaux, le SDN (*Software Defined Networking*), et NFV (*Network Functions Virtualization*) dans le cadre du projet OpenCloudware.

Dans cette HDR nous mettons en évidence les travaux concernant le déploiement des services réseau à la demande dans un contexte de Cloud Networking. Nous situons notre recherche dans un contexte évolutif des usages et des technologies. Ces deux points de vues introduisent des contraintes et des besoins à considérer lors la conception des nouvelles offres et des nouvelles architectures de services réseau. Dans ce contexte, *on demand* fait référence au modèle de consommation considéré autant à l'usage par le client final, que par le provisionning de services/ressources au niveau applicatif, réseau et infrastructure du fournisseur/opérateur.

Inspirés du modèle Cloud et son traitement des services applicatifs, nous introduisons notre démarche de dynamique de l'offre réseau dans le Cloud Networking. Le besoin d'agiliser la conception du service réseau pour fournir des services on demand, et d'introduire de la dynamique lors du déploiement et de la gestion des services réseau nous mènent à repenser les réseaux et leur conception en introduisant des mécanismes plus autonomes et intelligents. Pour ce faire nous introduisons la modélisation des aspects autant fonctionnels que non fonctionnels (la QoS) des composants. L'introduction de la QoS constitue notre fils conducteur pour considérer une personnalisation du service réseau et pour aller vers une convergence entre le contrôle et la gestion.

Nous présentons notre analyse et notre point de vue pour l'application des approches SOA, SDN et NFV permettant la définition du *réseau comme un service*. Nous intégrons une démarche QoS-aware pour définir le déploiement dynamique des réseaux virtuels personnalisés. Nous définissons à cet effet une nouvelle phase du cycle de vie des services réseau en milieu virtualisé : le *Network Virtual Deployment*. Nous complétons notre proposition avec la description de l'offre du réseau virtuel.

**Mots-clés :** Service réseau on demand, NaaS, Cloud Networking, Déploiement de réseaux virtuels, Description de réseaux virtuels.



# Abstract

My research interests cover the dynamic deployment of network services, by integrating two aspects: information systems and QoS for networks' management, and the Service-Oriented Architectures (SOA). I have treated these topics through the application of:

- User-Centric and networks overlay paradigms, which I applied in the UBIS project (User Centric - ubiquity and Integrated Services)
- Networks Virtualization, SDN (Software Defined Networking) and NFV (Network Functions Virtualization), which I investigated in the OpenCloudware project.

In this HDR, we present our work on the deployment of on-demand network services in the context of Cloud Networking. Our research addresses the issues of dynamic usage changes within the context of permanent technological evolution. Both usage and technological perspectives are associated to specific constraints and requirements, which need to be considered when designing new network services and architectures. Here, "On demand", refers to the consumption model as seen by the end customer, but also by the application and network levels for services/resources provisioning within the infrastructure of provider/operator.

Following the Cloud approach for application services, we introduce our proposal for dynamic providing of Cloud Networking. The need of an agile design of network services to provide *on demand* services, and the dynamicity when deploying and managing network services, lead us to rethink networking and its design with more autonomous and intelligent mechanisms. For this purpose, we introduce our modeling of functional and non-functional (QoS) aspects of the architecture components. QoS is our enabler to provide a *network customization service* and to move towards convergence between the control and management.

We present our perspective and our analysis on the application of SOA, SDN and NFV paradigms for the definition of the *network as a service*. We also propose to integrate a QoS-aware approach to define the dynamic deployment of customized virtual networks. For this purpose, we define a new phase in the lifecycle of network services within virtualized environments: the *Network Virtual Deployment*. We complete our proposal with the description of the virtual network offer.

**Key-words** : Network Service on demand, NaaS, Cloud Networking, Virtual Network Deployment, Virtual Network Description.





*"No sé como empiezan las cosas, y no tengo interés en verlas terminadas" CDP*

*A mon père Carlos, qui est dans ma vie source de motivation et inspiration*

*A ma mère Gladys, qui m'a toujours soutenue dans tous mes projets*

*A ma grande mère Elba, qui m'a appris à être fidèle à mes principes*

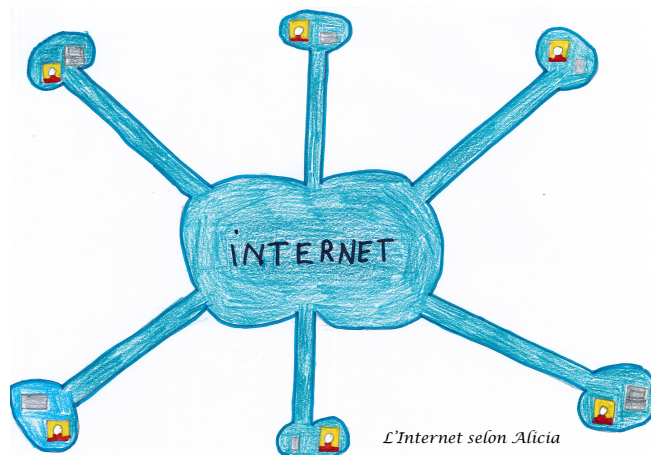
*A mes frères, Charles, Jorge, William et David, rencontrés grâce Internet*

*A ma petite sœur Elizabeth, qui est toujours avec moi dans mon cœur*

*A ma famille au Venezuela*

*A l'amour de ma vie, pour qui le temps reste encore un mystère*

*A mes trois enfants, Gabriel, Alicia et Raphaël, les uniques nœuds fixes dans le réseau personnel de ma vie*





# Remerciements

*Cette HDR constitue le fruit des précieuses collaborations avec un riche entourage auquel j'adresse mes remerciements.*

Je remercie très chaleureusement Noémie Simoni, pour le temps partagé et pour sa disponibilité, pour les nombreuses réunions qui nous ont permis de compléter nos points de vue, pour son aide dans la préparation de ce manuscrit, et pour m'avoir intégré à ces thématiques de recherche. Il est extrêmement rare de trouver, au milieu scientifique, un guide qui vous accompagne avec ses idées et qui vous permet de faire avancer les vôtres à travers une vraie collaboration. Grâce Noémie pour ta confiance et ton amitié.

Je remercie tous les membres du jury pour m'avoir fait l'honneur de participer à cette HDR en examinant mes travaux :

- M Guy Pujolle, mon parrain à UPMC, pour son appui et ses bonnes dispositions, pour la confiance accordée à travers son soutien, et pour ses conseils avisés. Merci également pour son inestimable aide en l'organisation de cette HDR.
- Mes rapporteurs, Mme Michelle Sibilla, Mme Francine Krief et Mr Harry Perros, pour leurs précieuses recommandations et pour m'avoir fait profiter de leurs expertises et points de vue sur mes travaux de recherche.
- Mr Yacine Ghamri-Doudane et Mr Christophe Cérin, les examinateurs, qui ont accepté de participer à mon jury et examiner mes travaux.

Un grand merci à toutes les personnes avec qui j'ai pu collaborer tout au long de ces années, mes collègues de bureau au L2TI, en particulier Khaled, Nadjib et Marwen, mes étudiants de master et de doctorat, mes partenaires de projets UBIS et OpenCloudware, l'équipe AIRS et mes collaborateurs à Télécom ParisTech, notamment Inès Ayadi.

Merci à ma mère et mes amis, qui ont toujours été présents partageant la vie au quotidien.

Enfin, merci à tous ceux qui directe ou indirectement m'ont permis d'avancer et d'aller vers le début d'un nouveau cycle de ma carrière.



# Table des Matières

Résumé	v
Abstract	vii
Remerciements	xi
Liste des Figures	xv
Liste des Tableaux	xvii
Acronymes	xix
<b>1 Introduction</b>	<b>1</b>
1.1 Notre Objectif	2
1.2 Notre Motivation	2
1.3 Les paradigmes	3
1.3.1 La Virtualisation	3
1.3.2 La Programmabilité	4
1.4 Convergence et QoS	5
1.4.1 Convergence du plan de contrôle et du plan de gestion	6
1.4.2 Un modèle QoS	7
1.5 Structuration du document	8
<b>2 L'évolution des réseaux en milieu virtualisé</b>	<b>11</b>
2.1 La virtualisation des réseaux	12
2.2 L'évolution du Cloud Networking	13
2.2.1 Virtualisation des équipements Réseaux	14
2.2.2 Services réseau basés QoS	15
2.2.3 Cloudification du réseau	18
2.2.4 Softwarisation du réseau	19
2.3 Conclusion	19
<b>3 Une architecture pour la personnalisation des services réseau</b>	<b>21</b>
3.1 NaaS (virtualisé (NFV) et programmable (SDN))	21
3.2 Description de l'architecture NaaS	22
3.3 Les nouveaux éléments constitutifs	24

3.3.1	Des composants des services réseau auto-contrôlables . . . . .	24
3.3.2	La Session réseau . . . . .	26
<b>4</b>	<b>Le cycle de vie du NaaS</b>	<b>29</b>
<b>5</b>	<b>Vers un déploiement continu de services réseau à la demande : la dimension fonctionnelle</b>	<b>33</b>
5.1	<i>Network Virtual Deployment</i> . . . . .	34
5.1.1	Phase de <i>Building VN+VNOS</i> . . . . .	35
5.1.2	Phase de <i>VN Functions Distribution</i> . . . . .	35
5.2	Une approche <i>QoS aware</i> pour le déploiement des réseaux virtuels à la demande	35
5.3	Le processus de déploiement de réseaux virtuels . . . . .	37
<b>6</b>	<b>Déploiement des réseaux virtuels : la dimension informationnelle</b>	<b>39</b>
6.1	Les travaux de standardisation . . . . .	40
6.2	OVF : un langage orienté déploiement . . . . .	40
6.3	Une description des réseaux virtuels pour un déploiement dynamique . . . . .	43
<b>7</b>	<b>Use case : vers une personnalisation autonome continue</b>	<b>49</b>
<b>8</b>	<b>Expérimentation</b>	<b>53</b>
8.1	Choix des composants . . . . .	53
8.2	But de l'expérimentation . . . . .	54
8.3	Expérimentations réalisées . . . . .	55
8.3.1	Use case 1 : Mesure du temps de réponse . . . . .	55
8.3.2	Use case 2 : Mesure de la gigue . . . . .	56
8.4	Analyse des résultats . . . . .	57
<b>9</b>	<b>Conclusion</b>	<b>59</b>
<b>10</b>	<b>Perspectives de recherche</b>	<b>61</b>
10.1	Axe 1 : Gestion et orchestration pour un déploiement on demande . . . . .	62
10.2	Axe 2 : Content Centric . . . . .	63
10.3	Axe 3 : Domaine Véhiculaire . . . . .	64
	<b>Liste des publications</b>	<b>65</b>
	<b>Bibliographie</b>	<b>67</b>

# Liste des Figures

1.1	L'évolution de la virtualisation réseau . . . . .	4
1.2	Convergence entre la gestion et le contrôle . . . . .	6
1.3	Modèle informationnel de QoS . . . . .	7
2.1	Architecture du fournisseur de Cloud Networking . . . . .	16
2.2	Classification de VN-CoS . . . . .	17
3.1	Architecture NaaS virtualisée et programmable . . . . .	23
3.2	Structure d'un composant SCC (Self-Controlled Component) . . . . .	25
3.3	La session réseau et les services : Network et Media Delivery . . . . .	26
3.4	La session réseau et les niveaux de visibilité en milieu virtualisé . . . . .	27
4.1	Le cycle de vie des services réseau en milieu virtualisé . . . . .	30
4.2	La phase de design . . . . .	30
4.3	La phase de déploiement dans : (a) un environnement IT, (b) proposée dans un environnement de virtualisation réseau . . . . .	31
4.4	La phase d'exploitation . . . . .	31
5.1	La phase du <i>Network Virtual Deployment</i> et les différents plans des services du NaaS . . . . .	34
5.2	Une approche <i>QoS aware</i> pour le NaaS . . . . .	36
5.3	Les contraintes de QoS et de placement au niveau de nœuds . . . . .	36
5.4	Les contraintes de QoS et de placement au niveau des liens . . . . .	37
5.5	Le processus de déploiement virtuel des réseaux virtuels. . . . .	38
6.1	Exemple d'un package OVF . . . . .	41
6.2	Positionnement de la <NetworkSection> dans une spécification OVF . . . . .	41
6.3	Définition dans OVF de la section <NetworkPortProfile> . . . . .	42
6.4	Définition dans OVF d'un <EthernetPortItem> . . . . .	42
6.5	Ensemble des nouvelles sections proposées . . . . .	44
6.6	Structure de l'extension OVF pour la description du réseau virtuel . . . . .	44
6.7	Extension OVF pour décrire le VirtualNetworkSection . . . . .	45
6.8	Extension OVF pour décrire le VirtualNodeSection . . . . .	45



6.9	Extension OVF pour décrire le VirtualLinkSection . . . . .	46
6.10	OVF extension - VirtualQoSNetSection . . . . .	46
6.11	Description du réseau virtuel : exemple de Fichier en OVF++ - Virtual Network Section . . . . .	47
7.1	Use case Cloud Gaming - NaaS Cloudifié, Virtualisé et programmable pour un service personnalisé . . . . .	50
8.1	Environnement d'expérimentation sur GRID 5000. . . . .	55
8.2	(a) Scénario 1 : configuration sans nœuds partagés (b) Scénario 2 : configuration avec des nœuds partagés . . . . .	57
10.1	Contexte de la recherche Les perspectives . . . . .	61
10.2	Les modèles de QoS et de profils appliqués au contexte du déploiement des réseaux virtuels . . . . .	63

# Liste des Tableaux

1.1	Les Profils et les valeurs mesurables pour le modèle de QoS . . . . .	8
6.1	Extension OVF proposée . . . . .	43
8.1	Caractéristiques des nœuds GRID 5000 . . . . .	53
8.2	Expériences RTT – Use Case 1 . . . . .	56
8.3	Configuration du Trafic - Use Case 2 . . . . .	56
8.4	(a) Flux RTP . . . . .	57
8.5	(b) Flux UDP . . . . .	57



# Acronymes

<b>ACTN</b>	<b>Abstraction and Control of Transport Networks</b>
<b>ADL</b>	<b>Architecture Definition Language</b>
<b>API</b>	<b>Application Programming Interface</b>
<b>ATM</b>	<b>Asynchronous Transfer Mode</b>
<b>CCDN</b>	<b>Cloud Content Delivery Networks</b>
<b>CDN</b>	<b>Content Delivery Networks</b>
<b>CDMI</b>	<b>Cloud Data Management Interface</b>
<b>CIMI</b>	<b>Cloud Infrastructure Management Interface</b>
<b>CNS</b>	<b>Cloud Networking Services</b>
<b>DNS</b>	<b>Domain Name System</b>
<b>DMTF</b>	<b>Distributed Management Task Force</b>
<b>EMML</b>	<b>Enterprise Mashup Markup Language</b>
<b>E2E</b>	<b>End To End</b>
<b>FIB</b>	<b>Forwarding Information Base</b>
<b>GCM</b>	<b>Grid Component Model</b>
<b>IaaS</b>	<b>Infrastructure as a Service</b>
<b>IMS</b>	<b>IP Multimedia Subsystem</b>
<b>IoT</b>	<b>Internet of Things</b>
<b>KPI</b>	<b>Key Performance Indicators</b>
<b>LSP</b>	<b>Label Switching virtual Path</b>
<b>MPLS</b>	<b>Multi Protocol Label Switching</b>
<b>M2M</b>	<b>Machine To Machine</b>
<b>NaaS</b>	<b>Network as a Service</b>
<b>NFV</b>	<b>Network Functions Virtualization</b>
<b>NFVIaaS</b>	<b>Network Functions Virtualization Infrastructure as a Service</b>
<b>NIaaS</b>	<b>Network Infrastructure as a Service</b>
<b>NLN</b>	<b>Node Link Network model</b>
<b>NOS</b>	<b>Network Operating System</b>
<b>NPaaS</b>	<b>Network Platform as a Service</b>
<b>NSaaS</b>	<b>Network Software as a Service</b>
<b>OCCI</b>	<b>Open Cloud Computing Interface</b>

<b>OSI</b>	<b>Open System Interconnection</b>
<b>OTT</b>	<b>Over The Top</b>
<b>OVF</b>	<b>Open Virtualization Format</b>
<b>OWL</b>	<b>Ontology Web Language</b>
<b>PaaS</b>	<b>Platform as a Service</b>
<b>QoS</b>	<b>Quality of Service</b>
<b>RIB</b>	<b>Rules Information Base</b>
<b>RTP</b>	<b>Real Time Protocol</b>
<b>RTT</b>	<b>Round Trip Time</b>
<b>SaaS</b>	<b>Software as a Service</b>
<b>SCA</b>	<b>Service Component Architecture</b>
<b>SCC</b>	<b>Self Controlled Component</b>
<b>SDN</b>	<b>Software Defined Networking</b>
<b>SLA</b>	<b>Service Level Agreement</b>
<b>SOA</b>	<b>Service Oriented Architecture</b>
<b>TOSCA</b>	<b>Topology and Orchestration Specification for Cloud Applications</b>
<b>UDP</b>	<b>User Datagram Protocol</b>
<b>USDL</b>	<b>Unified Service Description Language</b>
<b>VLAN</b>	<b>Virtual Local Area Networks</b>
<b>VN</b>	<b>Virtual Network</b>
<b>VN-CoS</b>	<b>Virtual Network Classes of Services</b>
<b>VNF</b>	<b>Virtualized Network Function</b>
<b>VNFaaS</b>	<b>Virtual Network Function as a Service</b>
<b>VNOS</b>	<b>Virtual Network Operating System</b>
<b>VNPaaS</b>	<b>Virtual Network Platform as a Service</b>
<b>VC</b>	<b>Virtual Circuit</b>
<b>VPC</b>	<b>Virtual Private Cloud</b>
<b>VPCN</b>	<b>Virtual Private Connectivity Network</b>
<b>VPN</b>	<b>Virtual Private Network</b>
<b>VM</b>	<b>Virtual Machine</b>





# 1

## Introduction

Notre société numérique est aujourd'hui un réservoir des usages et des technologies qui évoluent très rapidement. Les services et les réseaux de communication prennent une place importante dans notre quotidien. L'environnement numérique devient un élément complexe, du d'une part à l'hétérogénéité de l'environnement réseaux (technologies réseaux différentes et utilisables 3G, 4G, ADSL, etc.), et d'autre part, aux nouvelles contraintes liés aux usages (la mobilité, l'ubiquité et l'accessibilité des services). Les opérateurs et fournisseurs doivent donc prendre en compte de plus en plus sérieusement les offres de service dans leurs stratégies, pour satisfaire les demandes selon les nouveaux modes de consommation et usages. Ces défis font référence à la garantie de la QoS, l'introduction rapide des nouveaux services et la maîtrise de nouvelles technologies, tout en réduisant les coûts opérationnels des solutions. Pouvoir s'adapter aux nouveaux usages et déployer rapidement et efficacement de nouveaux services réseaux constitue un défi majeur pour assurer une mise en place flexible des services à la demande.

Mais aujourd'hui, pour répondre aux demandes du marché nous constatons que le paysage technologique répond soit aux exigences de l'informatique (*Application Centric*), soit aux exigences réseaux (*Network Centric*), alors que l'utilisateur accède à l'ensemble des ressources, c'est à dire aux ressources applicatives, à celles du réseaux et/ou de stockage. L'*User Centric* a donc besoin d'une vue globale agile et dynamique pour des services *on demand* et une QoS de bout en bout.

Service *on demand* constitue le paradigme qui traduit la façon dont les utilisateurs désirent accéder aujourd'hui à leurs applications. L'utilisateur veut être au centre de l'architecture, il veut des accès personnalisés et simplifiés. Les modèles de consommation informatique ont ainsi évolués mettant en place une approche d'usage à la demande, c'est le cas du Cloud Computing. Mais quand on regarde les services du Cloud, ils sont livrés à distance le plus



souvent à travers Internet. Du point de vue du provisionning des services, les services Cloud impliquent donc non seulement des fonctions informatiques fournies par l'infrastructure mais aussi des fonctions de communications offertes par le réseau.

Bien que depuis une décennie, les technologies s'adaptent à cet *User Centric* elles ne supportent pas encore la nécessaire innovation continue !

Dans ce contexte évolutif des usages et technologies réseaux, notre objectif (1.1) passe par la réponse aux questions suivantes : Quelles fonctionnalités devraient permettre et favoriser la personnalisation des services réseaux? Motivée par l'évolution de l'IT, avec les modèles du Cloud (1.2), qui a mis à profit les propriétés de la Virtualisation, de la programmabilité, et du «As a Service», comment intégrer ces nouveaux paradigmes dans les réseaux (1.3) ? Y a-t-il d'autres aspects à considérer pour permettre la flexibilité de l'offre réseau dans des environnements dynamiques et compétitifs (1.4) ? En répondant à ces premières questions dans cette introduction, nous posons les fondamentaux qui nous permettent d'envisager une vue globale convergente afin d'avoir l'hyper connectivité désirée.

## 1.1 Notre Objectif

Dans le cycle de vie des réseaux, entre conception et exploitation nous avons le déploiement sur lequel repose la construction et l'évolution des infrastructures. Les cycles de déploiement se raccourcissent car il faut intégrer rapidement les changements technologiques et être en mesure d'en évaluer les conséquences et impacts sur son environnement. C'est dire que pour des services réseaux *on demand* le déploiement doit être intelligent et dynamique. Un déploiement intelligent implique du côté fournisseur, la capacité d'appliquer des stratégies différentes (énergie (green), QoS, sûreté de fonctionnement) lors du déploiement et de l'exploitation pour pouvoir satisfaire des demandes évolutives liées aux différents usages, et du côté utilisateur, la faculté de faire évoluer ces demandes en fonction de différents facteurs comme la préférence, la localisation, ou l'agenda de l'utilisateur.

Notre objectif est donc de faire évoluer le déploiement des réseaux vers un processus qui devient continu, répondant à la volée aux demandes des utilisateurs.

## 1.2 Notre Motivation

Les réseaux actuels ne répondent plus aux contraintes imposées par les nouveaux usages et les applications innovantes tirées par l'essor du Cloud Computing, le développement du M2M et plus globalement par l'Internet des objets (IoT).

Les architectures réseau deviennent de plus en plus complexes, difficiles à gérer et à maintenir et les réseaux doivent fournir des fonctionnalités de plus en plus riches. Les réseaux doivent donc évoluer afin de répondre aux nouveaux besoins, tant en termes de mise à l'échelle, de rapidité de déploiement de nouveaux services réseau, que de gestion plus facile et plus flexible de ces services réseau et des réseaux eux-mêmes.

Du côté des services applicatifs, le Cloud Computing est sans aucun doute, le modèle qui additionne la plupart des critères mentionnés (dynamicité, flexibilité, programmabilité,

virtualisation, personnalisation), et qui constitue à mon sens, la base pour la mise en place des solutions agiles dans l'Internet du futur.

À l'instar de IT, il est devenu évident pour les opérateurs de réseau et les fournisseurs de réseau, que la virtualisation, le Cloud et la programmabilité avec plus particulièrement les nouveaux paradigmes en émergence tels que le SDN (*Software Defined Networking*) [1] et le NFV (*Network Functions Virtualization*) [2] sont prometteurs pour concevoir des architectures réseau simplifiées et agiles à gérer.

## 1.3 Les paradigmes

La virtualisation et la programmabilité sont aujourd'hui deux fondamentaux guidant l'évolution de tout ce contexte. Ces deux tendances constituent les bases de construction des services réseau de l'Internet du Futur. Les services réseau et les équipements deviennent virtuels et personnalisables. Les équipements deviennent programmables, et ceci s'applique autant aux dispositifs physiques que virtuels. Ces deux approches combinées permettent d'offrir, aux différents acteurs, des environnements flexibles et dynamiques. En addition, le paradigme SOA (*Service Oriented Architecture*) guide l'évolution de la convergence actuelle entre les réseaux et les services pour des solutions de bout en bout.

### 1.3.1 La Virtualisation

La notion de virtualisation propose une abstraction logique par rapport au matériel. Le concept de virtualisation est connu depuis les années 1960 en informatique. Il se réfère à un ensemble de techniques d'abstraction des ressources de calcul logiques depuis des ressources physiques. Les avantages de la virtualisation sont diverses : le partage des ressources, la réduction du nombre d'équipements, la haute disponibilité (provisionnement à la demande), la scalabilité, l'isolation, la réduction de la complexité, la flexibilité dans l'utilisation des ressources, etc.

Au niveau de systèmes, la technique des machines virtuelles est attractive parce qu'elle permet de fournir une émulation exacte du matériel sous-jacent sur différents systèmes d'exploitation. Par exemple, faire exécuter plusieurs machines virtuelles sur un seul équipement physique (serveur, routeur, etc.) permet d'avoir des instances de plusieurs systèmes d'exploitation sur un seul ordinateur. Le concept de virtualisation en informatique a également été appliqué dans d'autres branches telles que la virtualisation de la mémoire (*memory virtualization*), la virtualisation du stockage (*storage virtualization*) et la virtualisation de bureau (*desktop virtualization*). La virtualisation est de plus en plus utilisée par les fournisseurs d'applications. Elle permet aussi aux clients d'isoler leurs propres systèmes des risques d'instabilité des applications lorsqu'ils abritent des services réseaux ou d'autres applications. Elle évite aux entreprises de multiplier le nombre des serveurs inutilement, et d'économiser la consommation d'énergie.

Au niveau réseau, les termes *virtualisation* et *virtuel* ont été associés d'abord à plusieurs technologies. Parmi ces dernières, on peut citer : les circuits virtuels d'ATM (*Asynchronous Transfer Mode virtual circuits*), les chemins d'accès virtuels de MPLS (*LSP : Label Switching virtual path*), les réseaux privés virtuels (VPN), les VLAN (*Virtual Local Area Networks*) et les réseaux

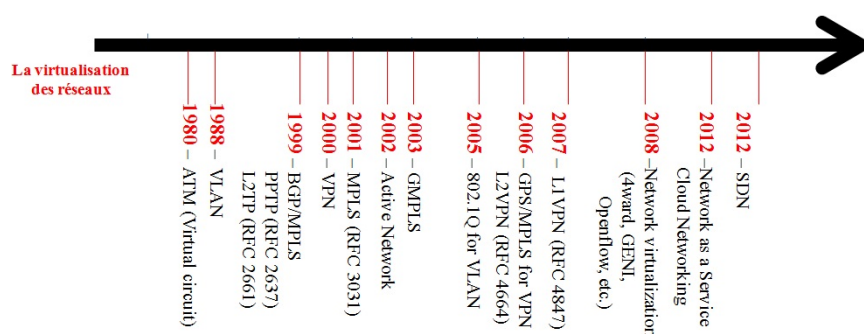


Figure 1.1 – L'évolution de la virtualisation réseau

de recouvrement virtuels (*Virtual overlay networks*) [3]. Plus récemment avec la notion de SDN la virtualisation peut être appliquée au plan de contrôle d'un réseau [1]. Nous voyons ainsi apparaître des nouvelles notions tels que le VNOS (*Virtual Network Operating System*) [4] [5]. A l'instar de la virtualisation des OS des machines virtuelles, le VNOS permet de gérer les ressources et fonctionnement d'un réseau virtuel. La figure 1.1 présente un résumé sur la virtualisation des réseaux, lié aux technologies et aux approches d'implémentation utilisées.

Dans ce même sens, la virtualisation est évoquée avec la notion de NFV qui a pour objet de mettre en service des fonctions réseau en tant que programmes logiciels sur des machines virtuelles [2]. Pour le NFV les appliances sont les réseaux virtuels. Le spectre des fonctions de réseau est large et inclut aujourd'hui des éléments de réseau des couches applicatives du modèle OSI (de 4 à 7) comme les pare-feux, les systèmes de prévention de sécurité, les répartiteurs de charge de travail, les solutions de CDN (*Content Delivery Network*), et des éléments d'accès réseau. L'enjeu visé par le NFV est l'exécution des fonctions réseau sur des environnements fournis par des fournisseurs différents, tout en garantissant une bonne portabilité et interopérabilité. Le réseau devient ainsi une plate-forme qui peut être ouverte à des développeurs tiers. Cette approche permet la programmation des applications spécifiquement pour les réseaux des opérateurs avec une promesse de meilleures performances. Pourtant, la virtualisation des fonctions réseaux proposée aujourd'hui se focalise sur la mise en service pour des équipements réseaux existants. La prise en charge de la composition des services réseaux à la demande reste encore un verrou à lever.

En résumé, la virtualisation en vigueur aujourd'hui se présente comme une solution qui permet de lever les barrières liées aux aspects matériels, de partager et d'optimiser l'utilisation des ressources dans un réseau [3]. Les applications et services n'étant plus liés à l'infrastructure physique, les avantages sont clairs, les réseaux gagnent en dynamique et flexibilité de leur offre, la maintenance est simplifiée.

### 1.3.2 La Programmabilité

La programmabilité met l'accent sur une logique de pilotage par le logiciel. Ainsi à travers des interfaces ouvertes, ce pilotage logique permet, à travers des programmes logiciels, de contrôler et de gérer les ressources à distance, de faire fonctionner et gérer des réseaux, etc. Le principal avantage de ce paradigme est de faire baisser les coûts d'exploitation et de dépenses

en équipements, grâce à l'automatisation de processus, du contrôle et de la gestion. La notion de programmabilité donne plus de souplesse à la fonction de configuration qui en générale est plutôt statique. Elle vise l'ouverture des équipements à travers l'utilisation des interfaces, de programmation ou d'accès, permettant à des tiers de concevoir, réaliser et déployer des nouveaux services. On peut aussi envisager l'activation et désactivation souple des fonctions dans les équipements d'un réseau ou l'installation rapide des nouvelles fonctionnalités.

La programmabilité introduit l'automatisation des traitements au niveau de la gestion, du contrôle et de l'usage des services. Cette automatisation permet :

- pour le traitement de la gestion : de gagner en efficacité à travers des solutions d'orchestration menant vers une capacité multi-décisionnelles,
- pour le traitement du contrôle : de mettre en place une activation, extensibilité et incrémentation rapides de la plate-forme et des solutions personnalisées (à travers l'interface Nord),
- pour le traitement lié à l'usage : d'introduire une rapidité de réaction vis-à-vis aux incidents opérationnels, et de mettre en place des déploiements flexibles avec une logique personnalisée selon l'usage.

Appliqué au niveau réseau, la programmabilité vise à fournir une dynamique en réponse au déploiement des nouveaux services et aux processus de contrôle et de gestion du réseau. Les technologies tels que le SDN et les interfaces standard du type OpenFlow [6] constituent aujourd'hui les solutions de ce paradigme pour les réseaux.

Le SDN propose une interface centrale pour piloter dynamiquement le réseau, tout en isolant les contraintes d'interfaces et autres spécificités associées au matériel. Le plan de contrôle étant séparé du plan de données, les administrateurs réseau peuvent contrôler et intervenir à distance et en temps réel sur le réseau, indépendamment de la présence physique des matériels. Cette capacité programmatique permet de faire évoluer les architectures réseau de manière à les rendre plus simples et plus dynamiques. Il s'agit de mettre l'intelligence à un niveau plus haut du réseau, au niveau de la mise à disposition de services, l'environnement réseau devient plus flexible et agile, avec des ressources pouvant être utilisées à la demande. Les différentes mises à jour seront simplifiées et plus rapides car effectuées par logiciel et des nombreuses manipulations sur le réseau peuvent être automatisées grâce à l'abstraction des couches réseau et l'accessibilité aux ressources via les API.

Ainsi les réseaux pilotés par logiciel et la virtualisation des fonctions réseau sont des technologies qui peuvent être utilisées de façon intégrée pour concevoir les nouvelles infrastructures programmables. On pourrait ainsi envisager au futur de modifier les interfaces pour avoir accès à de nouvelles fonctionnalités du plan de données, de celui du contrôle et de la gestion pour améliorer les capacités et l'efficacité des services réseaux. La standardisation des interfaces pour cet objectif, reste à ce jour un sujet ouvert.

## 1.4 Convergence et QoS

Avec la virtualisation et la programmabilité, propriétés donc, de tout environnement virtualisé, il nous faut reconsidérer les trois plans qui régissent notre domaine réseau : le plan des

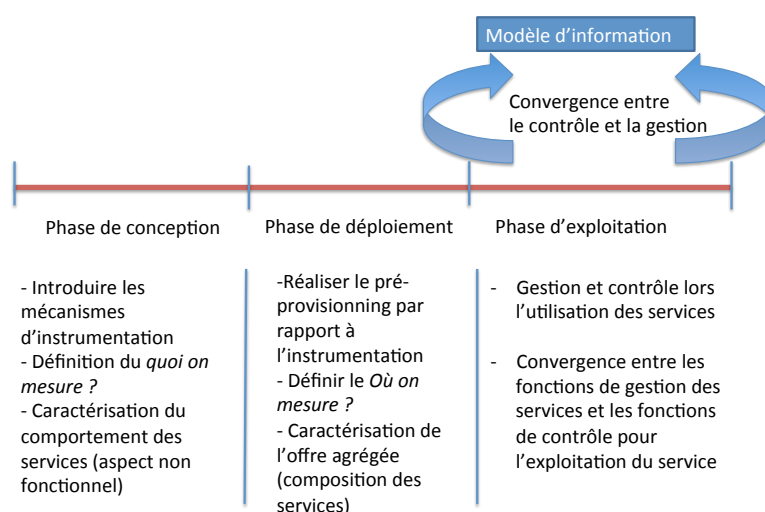


Figure 1.2 – Convergence entre la gestion et le contrôle

données avec SDN est piloté par le plan de contrôle qui doit, avec le plan de gestion, évoluer vers des mécanismes plus autonomes et intelligents.

Pour ce faire, deux premières notions nous semblent incontournables [7] [8] : celle de la convergence du plan de contrôle et du plan de gestion pour avoir des actions coordonnées et complémentaires (1.4.1) et celle de la QoS (1.4.2) pour avoir un *fil conducteur* depuis la conception jusqu'à l'exploitation.

### 1.4.1 Convergence du plan de contrôle et du plan de gestion

Les décisions prises par le contrôle et la gestion dépendent de la pertinence des données informationnelles disponibles. Cet aspect mène les opérateurs/fournisseurs à introduire des mécanismes d'instrumentation pour mesurer et caractériser le comportement des différents services. Cette instrumentation est nécessaire pour maîtriser les offres de services dynamiques et flexibles qui devront caractériser les environnements virtualisés du futur.

La gestion représente un élément clé intervenant dans les différentes phases de mise en place d'une solution architecturale et de son exploitation. La gestion s'applique donc aux services à différents niveaux (équipement, réseau, application).

L'information constitue dans ce contexte la base essentielle pour la prise de décisions. La modélisation de cette information est nécessaire pour permettre l'automatisation de processus de déploiement, et pour le contrôle et la gestion des solutions en réseaux, en phase d'exploitation. Le modèle doit ainsi représenter l'information permettant d'effectuer et automatiser les différentes opérations programmées par le concepteur de l'architecture. Pour ce faire il faut pouvoir indiquer dès la conception les évaluateurs de performance à utiliser.

La convergence des processus de contrôle et de gestion permettrait ainsi d'assurer l'agilité et la flexibilité nécessaires aux architectures des services. Cette convergence se fait à travers l'utilisation d'un modèle d'information (voir la figure 1.2), permettant d'introduire le niveau d'abstraction nécessaire à travers une représentation cohérente concernant les aspects à superviser et permettant de prendre les bonnes décisions au bon moment.

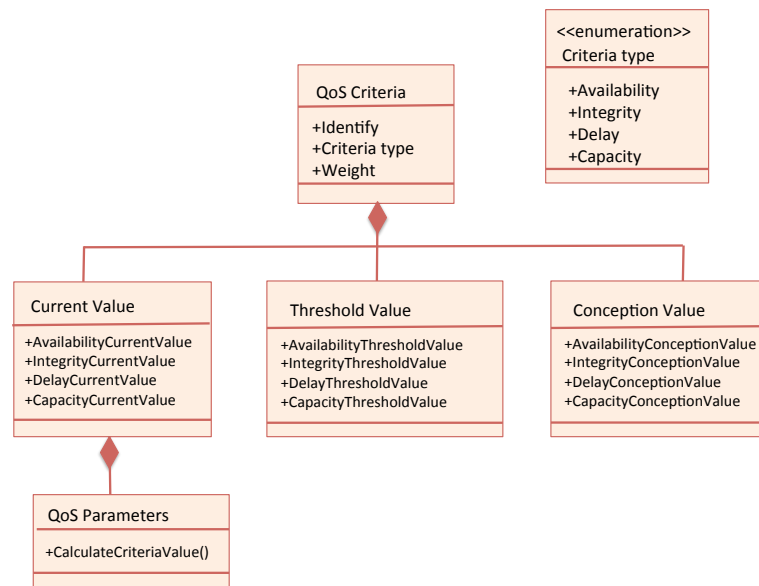


Figure 1.3 – Modèle informationnel de QoS

### 1.4.2 Un modèle QoS

Les propositions présentées dans ce document sont basées sur l'application d'un modèle générique de QoS [9] qui nous permet d'avoir une démarche QoS-aware.

L'objectif principal de notre modèle QoS générique est de spécifier les exigences non-fonctionnelles associées aux exigences fonctionnelles des services pour tenir compte non seulement des aspects de traitement, mais aussi de ceux liés au comportement. A cet effet, un modèle d'information est proposé pour décrire la QoS (aspects non-fonctionnels) de chaque ressource. Ce modèle est générique et unifié. Il est appliqué à toutes les couches d'une architecture (équipement, réseau, services). Les ressources dans chaque couche sont conçues comme un ensemble d'éléments de service. Le modèle d'information de QoS se décline en critères de qualité de service évalués à travers des paramètres mesurables.

La figure 1.3 montre le modèle informationnel de QoS. Le premier niveau représente les quatre critères de qualité de service que nous avons retenus pour chaque élément. Le second montre les valeurs selon la phase du cycle de vie. Enfin, le troisième niveau représente les paramètres nécessaires pour effectuer des mesures.

Les critères retenus sont :

- Disponibilité: représente le nombre moyen de demandes acceptées par le composant de service.
- Capacité: représente la capacité de traitement moyenne du composant de service pendant une unité de temps.
- Délai: est le temps moyen de traitement du composant de service.
- Fiabilité: est le nombre de demandes qui sont desservis par le composant de service.

Les valeurs considérées selon les phases du cycle de vie sont : [9] [10] (voir tableau 1.1).

- Les valeurs de conception sont déterminées lors de la phase de la conception des composants de service. Ils définissent les valeurs maximales du composant de service.
- Les valeurs courantes sont celles évaluées lors de la phase d'exécution.
- Les valeurs de seuil indiquent les valeurs limites à ne pas dépasser par un composant de service afin d'assurer un comportement normal.

Le modèle d'information de QoS permettra à l'architecte d'établir la structure qui répond le mieux à la demande. Il pourra ainsi considérer aussi bien les exigences fonctionnelles que non-fonctionnelles (QoS). Les informations de QoS sont incluses dans les profils des ressources [9].

Table 1.1 – Les Profils et les valeurs mesurables pour le modèle de QoS

	Desing phase	Deployment phase	Operational phase
Profile	Resource Profile	Usage resources profile	Real time profile
QoS Values	Desing QoS values	QoS Thresholds values	QoS current values

## 1.5 Structuration du document

Nous centrons cette HDR sur l'analyse des évolutions actuelles et à venir, pour faire face à tous ces changements liés aux usages et à leurs répercussions au niveau du réseau.

Nous pensons que les solutions futures seront basées sur la conception des nouvelles architectures réseaux orientée services, intégrant des mécanismes génériques de contrôle et de gestion, et automatisant à travers des modèles d'information le traitement dynamique du processus de déploiement et exploitation. La QoS devrait être intégré comme élément de base permettant d'automatiser le contrôle et la gestion, de caractériser et personnaliser les services réseau à déployer.

Suite à nos motivations, ce document présente nos propositions sur la mise en place des services réseau personnalisables et à la demande. Il présente les approches, intégrant un déploiement dynamique des services réseau, étudiées et proposées durant ces sept dernières années (2008-2015). Ces travaux ont été effectués dans le cadre de deux projets en particulier : UBIS (2008-2011) et OpenCloudware (2012-2015). Les approches étudiées lors la participation à ces projets se concentrent sur l'intégration de plusieurs points de vue : architectural, fonctionnel, organisationnel et informationnel. Nous considérons ces différents points de vues pour fournir une solution globale dans le cadre du déploiement dynamique des réseaux à la demande.

Après cette introduction, nous présentons dans le chapitre II notre analyse sur l'impact des paradigmes de virtualisation, programmabilité et Cloud dans le contexte du Cloud Networking. Nous proposons une classification représentant l'évolution du Cloud Networking qui nous mènera vers un NaaS virtualisé et personnalisable.

Le chapitre III décrit l'architecture NaaS proposée. Nous intégrons dans cette description le point de vue fonctionnelle de l'architecture (avec ces composants fonctionnels, et non fonctionnels) et les aspects organisationnels.

Le chapitre IV décrit le cycle de vie du NaaS et introduit le déploiement virtuel des services réseau. Pour ce faire nous analysons la problématique actuelle du placement, ce qui nous mène à introduire la définition d'un déploiement continu.

Le chapitre V décrit notre approche intégrant un déploiement virtuel et dynamique des services réseau. Nous présentons nos propositions concernant le déploiement personnalisé des réseaux virtuels à la demande. Nous décrivons la phase de déploiement virtuel ainsi que les deux nouveaux processus (*VN+VNOS Building* et *VN Functions Distribution*) introduits par cette phase.

Le chapitre VI décrit notre point de vue informationnel associé à notre démarche de déploiement dynamique des services réseau. Nous décrivons l'information qui accompagne notre approche de déploiement à travers une proposition de description pour le réseau virtuel à déployer en utilisant le langage OVF++.

Le chapitre VII présente notre point de vue sur la personnalisation des services. Nous présentons un use case dans la cadre du Cloud gaming.

Le chapitre VIII montre les premiers résultats des expérimentations réalisées pour la mise en place de l'architecture NaaS.

Nous finissons notre document en présentant les conclusions dans le chapitre IX, et nos perspectives de recherche dans le chapitre X.





# 2

## L'évolution des réseaux en milieu virtualisé

Les services et applications actuels sont conçus et déployés pour la plupart des cas dans des environnements virtuels dont le Cloud Computing constitue le représentant major. Le réseau joue un rôle fondamental dans le Cloud Computing aussi bien pour l'accès aux services de stockage du côté des Data Center, que pour l'accès aux services applicatifs du côté des serveurs. On peut dire que le modèle du Cloud impact tout l'écosystème et en particulier le modèle réseau. Ces changements sont perceptibles à plusieurs niveaux:

- Au niveau de l'infrastructure : car tout devient virtualisé et l'infrastructure devient programmable. Les serveurs et les applications deviennent mobiles.
- Au niveau des applications : car le Cloud permet d'intégrer diverses applications qui ont des besoins très différents par rapport au réseau.
- Au niveau de l'accès : car aujourd'hui avec la mobilité, l'accès aux espaces virtualisés et services dans le Cloud se réalisent aussi à travers des appareils mobiles. L'accès doit être garanti quelque soit la localisation et l'équipement utilisé.
- Au niveau du trafic : qui intègre les modèles principalement serveur-à-serveur et avec un emplacement indépendant des extrémités des deux côtés d'un service ou d'une transaction.

Nous pouvons constater également que le Cloud introduit les propriétés d'adaptabilité et de dynamique des solutions. L'adaptabilité implique le fait de pouvoir s'adapter aux changements de rendu du service en adaptant les ressources impliquées. La dynamique se réfère au fait de pouvoir instancier à la volée des services dans le Cloud. Au niveau réseau la dynamique impliquerait le fait de pouvoir déployer et adapter les services réseaux en accord avec les besoins exprimés au niveau applicatif (les flux de données).

Une telle vision exige de l'infrastructure du réseau sous-jacente d'être ouverte et d'exposer ses services vers les applications de couches supérieures. Cette vision vers le haut du système doit être complétée à l'instar du Cloud par une vision vers le bas, basée sur l'abstraction des ressources physiques du réseau. Comme nous l'avons déjà évoqué cette abstraction s'appuie sur la virtualisation. Appliqué au niveau réseau, la virtualisation permet le découplage des services de l'infrastructure réseau et la standardisation des équipements réseaux.

Plusieurs changements du réseau sont ainsi nécessaires pour tenir compte de ce nouvel paradigme. Les ressources réseau doivent pouvoir être attribuées de manière différenciée pour pouvoir répondre aux changements des demandes de services, de la QoS associée, et de la localisation des services applicatifs du Cloud. Mais garantir ces propriétés reste encore aujourd'hui un challenge à lever pour la mise en place des nouvelles solutions réseaux.

Nous allons étudier dans les sections suivantes l'évolution des technologies liées à la virtualisation réseaux (2.1) et aussi comment les notions de programmabilité et virtualisation ont impacté l'évolution du Cloud Networking (2.2) et ses différentes implémentations menant vers la définition d'un NaaS virtualisé et programmable que nous proposons dans le Chapitre 3.

## 2.1 La virtualisation des réseaux

L'Internet offre depuis 30 ans un support réseau pour exécuter une multitude des applications et de technologies, mais en raison de sa nature multifournisseur il est difficile aujourd'hui de trouver des consensus vis-à-vis l'évolution de son architecture. Les modifications de l'architecture existante sont limitées à de simples mises à jour incrémentielles. Un changement évolutif de son architecture actuelle devient presque impossible.

La virtualisation du réseau se présente aujourd'hui comme le paradigme permettant d'introduire les changements de l'architecture réseau nécessaires à l'évolution vers le Internet du futur. Un *réseau virtuel* a été défini dans la littérature comme une collection des nœuds et des liens virtuels placés sur un sous-ensemble des ressources réseaux physiques sous-jacentes [3]. Cette vision sur la virtualisation permet de découpler les rôles entre les fournisseurs de service, qui déploient leurs réseaux virtuels pour mettre en œuvre des nouvelles offres et le fournisseur de l'infrastructure, qui est le gestionnaire de l'infrastructure physique. En introduisant une pluralité des réseaux hétérogènes avec des architectures qui cohabitent sur un substrat physique partagé, la virtualisation favorise l'introduction des innovations et des applications diversifiées, permettant ainsi une évolution continue du réseau et son architecture.

Au niveau de la recherche différentes études ont été effectuées pour évaluer et analyser les bénéfices de la virtualisation au niveau réseaux. Les analyses effectuées permettent d'établir une série de caractéristiques afin d'évaluer le niveau de virtualisation, la technologie et l'architecture de support, le domaine d'application supporté par le réseau virtuel et son degré d'auto-gestion [3].

À l'instar de l'IT, les opérateurs se sont lancés dans la virtualisation des équipements réseaux, dans ce contexte des projets de recherche tels que X-Bone [11], VIOLIN [12], PlanetLab [13], se sont focalisés sur la connectivité des nœuds virtualisés ou sur le déploiement des liens virtuels entre des nœud physiques. Des autres projets plus récents, tels que GENI

[14], 4WARD [15], NouVeau [16] se focalisent sur la mise en oeuvre des autres mécanismes (que ceux de la virtualisation des nœuds et liens) pour atteindre la virtualisation des autres aspects réseaux, notamment du plan de gestion.

Les différents projets mentionnés mettent en évidence l'utilisation des technologies de virtualisation existantes : tunneling pour des VPNs (VIOLIN, GENI [17]) et des Overlays (X-Bone, Agave, 4Ward, PlanetLab) pour définir des liens virtuels, mais aussi des aspects de programmabilité (NetScript [18], Genesis [19]) inspirés des réseaux actifs et programmables et permettant de mettre en place de la flexibilité au niveau des nœuds physiques pour héberger plusieurs nœuds virtuels. Ces technologies sont utilisées à des niveaux OSI (physique, liaison, réseau ou application) différents selon les objectifs des projets.

Des concepts additionnels tels que : la Qualité de Service, et ses technologies d'implémentation associées à Internet, ont été aussi considérés par exemple dans le projet AGAVE [20] avec l'utilisation des techniques IP (DiffServ et InteServ). Des autres concepts tels que le Cloud et les SOA ont aussi été intégrés à la définition des projets en virtualisation réseau, par exemple dans 4WARD. [21] présente une étude exhaustive sur les derniers développements en matière de virtualisation de réseau orientée services pour soutenir le Cloud computing, en particulier du point de vue de la convergence entre le réseau et le cloud à travers le NaaS.

Cette évolution de la virtualisation se continue aujourd'hui à travers de la notion de SDN et la programmation du plan de contrôle. Pour aller plus loin, nous trouvons dans [22], [5] l'analogie entre l'OS d'une machine et le NOS (*Network Operating System*) du réseau. Dans l'équipement virtualisé, on s'abstrait de l'OS machine, chaque VM (*Virtual Machine*) peut s'installer avec son OS, l'hyperviseur gérant le partage des ressources. Similairement, le NOS gère l'accès à l'infrastructure réseaux pour délivrer le média entre deux extrémités dans une session. Il interroge les routeurs, les commutateurs, les câbles, pour établir la communication de bout en bout. La délivrance de média est effectuée à travers les services réseau qui définissent les règles d'accès, l'adressage, le multiplexage, le contrôle de flux, le contrôle d'erreur et le routage.

À l'instar de la virtualisation des OS, le NOS peut être aussi virtualisé dans le réseau. De la même façon que pour créer plusieurs VMs isolées et constituées de différents types d'applications, nous pouvons avoir plusieurs slices réseaux isolés, indépendants et personnalisés associés à un plan de contrôle propre. Pour poursuivre l'analogie il faudrait définir un hyperviseur réseau.

Afin de provisionner les différents demandes des ressources pour de fournisseurs de services différents, les fournisseurs d'infrastructure (physiques et virtuelles) doivent être en mesure de déterminer l'état des éléments de réseau correspondant, ce qui veut dire qu'un bon monitoring fera aussi partie d'une solution efficace.

## 2.2 L'évolution du Cloud Networking

Nous sommes intéressés par les leviers qui ont fait évoluer le Cloud Networking pour répondre aux besoins d'efficacité des ressources, de flexibilité des solutions, d'agilité dans la création de service, d'adaptation et de dynamique du réseautage. Nous allons étudier les travaux qui ont permis de faire évoluer le «cloud Networking» et les classer en fonction des besoins

couverts et des technologies utilisées. L'analyse de ce nouveau paysage nous permettra de proposer une architecture cible.

Une première solution a concerné les liens réseau pour mettre en place la connectique et les accès pour les Data Centers en utilisant la virtualisation des liens réseau. Puis il y a eu, celle des nœuds réseaux. Ainsi les routeurs virtuels embarquent plusieurs piles de protocoles afin d'avoir des sous réseaux adaptés. C'est pourquoi nous avons appelé cette phase du Cloud Networking *Virtualisation des équipements réseaux* (2.2.1).

Considérant que le Cloud Networking est là pour répondre aux besoins des applications Cloud, des offres différenciées, basées sur la QoS ont été investies par les académiques, nous avons appelé cette phase *Services réseau basés QoS* (2.2.2).

Mais cette solution repose sur les infrastructures d'aujourd'hui (IP, MPLS, . . .), or le réseau doit subir une mutation plus importante pour avoir la dynamique tant attendue par notre nouvel écosystème.

Nous trouvons ensuite, ce que nous pouvons appeler la *Cloudification du réseau* (2.2.3). En fait, basé sur le principe des NFV, c'est à dire la virtualization des fonctions réseaux, on déploie des VMs (contenant des NFV) sur des équipements banalisés. On a ainsi un menu complet des fonctions réseaux composables à la demande.

Parallèlement le besoin d'agilité favorise les capacités de programmabilité que propose SDN. Très vite une complémentarité SDN/NFV s'impose (2.2.4). Nous l'avons dénommé *Softwarisation du réseau*.

Finalement, on s'aperçoit qu'il serait nécessaire de fédérer l'ensemble de ces avancées, car chacune d'elles remplit un rôle important pour converger et avoir une solution globale (Chapitre 3).

### 2.2.1 Virtualisation des équipements Réseaux

On a très tôt fait de la virtualisation des liens réseaux, à travers des protocoles pour partager les ressources. VLAN, VPN et les mécanismes de tunneling sont des techniques de virtualisation des liens réseau, usitées depuis longtemps pour relier des sites distants et sécuriser les données échangées.

Le Cloud Networking, s'intéressant au début à mettre en place et garantir une connectique et un accès vers des services et des ressources placées dans des Data Centers, commence par utiliser la virtualisation des ressources physiques (des noeuds et des liens) au niveau IaaS. En ce sens, cette notion est ainsi généralement associée à la fourniture d'un réseau virtuel par les propriétaires de l'infrastructure réseau. Ainsi, de plus en plus les fournisseurs de Cloud se concentrent sur la virtualisation de leurs réseaux pour avoir plus de flexibilité et d'évolutivité dans leurs Data Centers. Un exemple est le Virtual Private Cloud (VPC) fournit par Amazon pour sécuriser la connectivité en nuage de Amazon EC2 [23]. De même, Microsoft Windows Azure [24] propose de construire des réseaux privés virtuels dans le nuage en créant les connectivités entre leurs sites. Ces exemples constituent un premier pas pour ajouter de la flexibilité dans les offres des services en réseau, mais ils n'offrent pas encore un réseau cloud virtuel à la demande, ni non plus des garanties de performance de trafic.

Des architectures de réseaux virtuels ont été proposées par la communauté de recherche afin d'optimiser l'utilisation des Data Centers, CloudNaas, SecondNet, VL2, Diverter, VICTOR

and NetLord sont des exemples de ces architectures [25]. Mais ces architectures présentent des solutions avec des limitations, l'allocation de la bande passante n'est pas dynamique mais statique et prédictive, sinon inexistante; d'autres sont limitées par rapport à leur scalabilité, performance et déploiement, qui dépendent de la topologie physique de l'infrastructure réseau et des équipements physiques et services installés. Aucune de ces architectures ne prend en considération la garantie de la QoS en fonction des flux de données à traiter.

Les travaux précédemment cités présentent la mise en place du Cloud Networking, impliquant la virtualisation des réseaux pour la connectique entre des data centers et des applications comme le CDN.

Ces approches visent à fournir une flexibilité dans le support de l'évolution des réseaux de services, et dans le contexte de la mise en réseau pour le Cloud. Mais les réseaux virtuels restent pour le Cloud Networking une offre avec des configurations statiques et sans garanties de QoS en fonction des flux transportés. La virtualisation reste appliquée aux ressources au niveau de l'infrastructure, et sans intégrer la prise en compte de l'approche de niveau service.

### 2.2.2 Services réseau basés QoS

C'est dans le contexte de la recherche que nous trouvons au cours de la dernière décennie, de nombreux efforts réalisés au niveau de l'analyse des flux applicatifs afin de fournir le réseau le plus adéquat. Les solutions proposées se basent sur la mise en place des réseaux superposés [26] pour faire coexister plusieurs réseaux sur la même infrastructure de substrat existant. Par exemple, [27] introduit le concept de VPSN (*Virtual Private Service Network*) [10] et de VPCN (*Virtual Private Connectivity Network*) qui définissent des réseaux superposés composés d'un ensemble de composants autonomes, partageables et auto-gérables (nœuds et liens). Nous trouvons aussi les projets GENI, 4WARD, PlanetLab, VINI, Trellis, etc., qui utilisent les technologies de virtualisation pour créer/tester des réseaux virtuels sur plusieurs technologies de réseau (sans fil, capteurs, réseaux cellulaires, etc.) permettant ainsi d'évaluer plusieurs protocoles réseau dans des environnements réalistes [3].

Mais la plupart des approches mentionnées définissent un réseau virtuel à partir d'un point de vue de l'équipement, c'est à dire de l'interface sud de l'architecture réseau, car comme dans le cas classique, le déploiement se focalise sur la mise en place d'un chemin pour connecter des serveurs et des équipement réseaux, lesquels sont maintenant virtualisés. Est ce suffisant pour offrir la flexibilité, c'est à dire un réseautage personnalisé selon les applications ? En fait, ces approches visent à fournir une flexibilité au niveau du support et dans le contexte de la mise en réseau pour le Cloud. Cependant, la composition d'un réseau en tant que service selon la demande et les caractéristiques des flux de données reste un défi à relever.

Une première réponse a consisté à créer des réseaux virtuels logiques en accord avec les exigences des flux selon un modèle de QoS (section 1.4.2). Ce modèle appliqué au niveau réseau décrit les aspects non-fonctionnels des services réseaux tels que la transmission, le routage, la sécurité, etc. Les paramètres de services réseaux à mesurer sont en fonction des exigences de niveau d'application. Le délai est par exemple un paramètre plus critique dans l'application en temps réel que dans le service de transfert de fichiers. Dans le contexte de Cloud Networking, le modèle permet de caractériser le comportement des services de réseau, en définissant le critère de qualité de service associé avec le réseau virtuel à déployer. La section suivante

présente l'application du modèle de QoS pour la mise en place des réseaux virtuels personnalisables dans le contexte du Cloud Networking.

### *VN-CoS pour le Cloud Networking*

L'idée de cette solution est de permettre la mise en place des services réseaux Cloud avec une approche QoS-aware pour traiter les besoins des flux des applications Cloud (provenant du SaaS ou PaaS) [28], [4].

Dans cette proposition la principale contribution est la classification de services Cloud Networking (VN-CoS) pour représenter les différents types de flux. Cette classification permet de faire évoluer les offres du Cloud Networking en intégrant la QoS à l'offre du service du réseau virtuel.

Deux questions importantes, concernant les fournisseurs du Cloud Networking, ont été adressées par cette proposition : (i) déterminer la qualité de service à fournir par chaque service de réseautage pour le Cloud; (ii) fournir une différenciation des services pour assurer la prise en compte des exigences de performance des applications. Deux aspects ont été traités pour répondre à ces questions. D'une part, une modélisation des composants CNS (*Cloud Networking Services*) avec un mécanisme d'auto-contrôle de leur comportement au moment de l'exécution [28]. Et d'autre part, une nouvelle classification des CoS est faite, pour les flux applicatifs dans le Cloud, basée sur les sensibilités à la QoS [4]. Ainsi la personnalisation de la composition et la mise en réseau d'un Cloud virtuel sont adressées en choisissant les services de réseautage les plus appropriées et en déterminant la classe de service la plus adéquate. L'adaptation, dans le cas où les exigences des flux changent ou lorsqu'il y a dégradations du réseau, est ainsi possible.

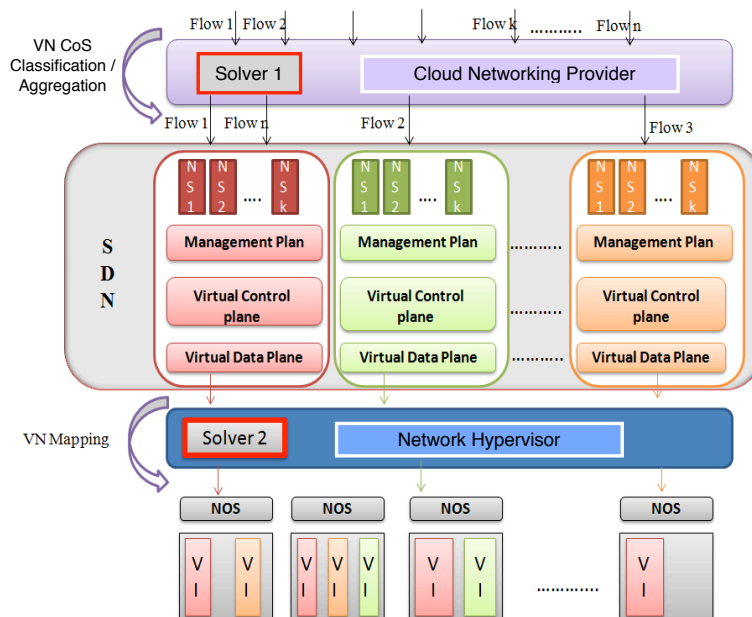


Figure 2.1 – Architecture du fournisseur de Cloud Networking

Pour intégrer ces deux aspects [22] propose une architecture permettant de sélectionner, composer et configurer les services de réseaux virtuels qui répondent aux exigences des flux, indépendamment des dispositifs réseaux et des technologies de réseau sous-jacent. Pour ce faire l'architecture présente deux solveurs, celui de l'interface sud et celui de l'interface nord ainsi que l'hyperviseur réseaux gérant les ressources réseaux de l'infrastructure. La figure 2.1 montre l'architecture proposée.

Le «solveur 1» (interface nord) traite le classement des flux par la détermination de ses sensibilités aux critères de qualité de service (disponibilité, délai, capacité et fiabilité). Selon le degré de sensibilité (faible, moyen, élevé) à chaque critère de QoS, des classes de services possibles sont définies. Le Cloud Networking provider associe ainsi le flux à une classe de service (VN-CoS). Ensuite, le Cloud Networking provider effectue une agrégation sur la sélection de CoS en utilisant des mécanismes d'ingénierie de trafic. La solution logique, le réseau virtuel, est construite (VN-CoS).

Pour décrire les classes de flux pour le Cloud Networking une classification de services «Virtual Network Classe of Service» (VN-CoS) a été proposée [4]. La figure 2.2 montre des exemples de VN-CoS.

Le «solveur 2» (interface sud) s'occupe du mapping entre la solution virtuelle et son placement dans l'infrastructure physique. Une fois que la description du plan de données pour chaque réseau virtuel est faite, une mise en correspondance entre les éléments virtualisés (nœud et liens) et les éléments de l'infrastructure de réseau sous-jacent est nécessaire. La requête de mapping du réseau virtuel comprend: le graphe de la logique du service réseau (ses nœuds et ses liens), sa topologie et les contraintes qui doivent être garanties au cours du mapping. Le solveur 2 permet d'intégrer les différentes contraintes de placements pour décrire le réseau virtuel qui doit être par la suite mappé.

	Flow sensitivity				Existent CoS					
	Availability	Delay	Capacity	Reliability	Y.1541	Diffserv PHB	MPLS (DS-TE)	UMTS	802.1d	WIMAX
VN-CoS1	x	x	x	x	Class 5	BE	0	Background	1	BE
VN-CoS2	x	xx	x	x	Class 1	AF3.3	3	Streaming	5	ErtPs
VN-CoS3	x	xxx	x	x	Class 0	EF	5	Conversational	7	ErtPs
VN-CoS4	x	x	xxx	x	Class 2	AF4.1	5	Streaming	5	RtPs
VN-CoS5	xxx	x	xx	xxx	Class 4	AF2.1	2	Interactive	2	NrtPs
VN-CoS6	x	xxx	xxx	x	Class	AF4.1	5	Streaming	4	ErtPs
...	...	...	...	...	...	...	...	...	...	...
VN-CoSk	xx	xx	xx	x	Class 1	AF4.2	4	Streaming	5	RtPs
...	...	...	...	...	...	...	...	...	...	...
VN-CoS81	xxx	xxx	xxx	xxx	Class 0	EF	5	Conversational	6	UGS

Figure 2.2 – Classification de VN-CoS

Contrairement à la différenciation des CoS classique qui fait des traitements similaires pour les flux appartenant au même niveau de priorité, les classes VN-CoS ont été conçues pour personnaliser les traitements de chaque flux en déterminant de manière appropriée ses sensibilités aux critères de QoS. La différenciation des services ainsi proposée aide le fournisseur du Cloud Networking à contrôler la performance du réseau et à déterminer la pile de protocole adaptée lors de la construction du réseau virtuel. En fait, une fois que la classe adéquate VN-CoS est sélectionnée, une correspondance une-à-une entre cette classe et les classes de trafic existants (Y.1541, Diffserv PHB, 802.1d, etc.) est alors effectuée.

Mais le paysage réseau commence à se modifier avec la virtualisation des fonctions réseau (NFV). Les paradigmes évoluent, à partir d'une classification prédéfinies des services, vers une composition des services virtualisés et à la demande.



### 2.2.3 Cloudification du réseau

L'objectif général des approches SDN et NFV dans le Cloud, est de fournir des fonctions, des réseaux et des infrastructures en tant que des services plutôt que comme caractéristiques des systèmes intégrés verticalement.

Comme mentionné dans l'introduction, à l'instar des informaticiens du cloud, la plupart des opérateurs utilisent la virtualisation des nœuds pour la conception et déploiement des solutions réseaux. La virtualisation permet donc de rester proche des applications sans avoir à s'inquiéter pour l'aspect physique d'implémentation et la nature des équipements réseaux. Les opérateurs peuvent ainsi offrir des réseaux plus adaptés et de façon dynamique sans avoir à toucher leur infrastructure physique. La similitude s'est concrétisée avec la formalisation des NFV [2], [29] qui vise à virtualiser les fonctions du réseau et supporter tous types de services de télécommunication exécutés aujourd'hui dans un Cloud IT standard. Dans le contexte de NFV, certaines fonctions telles que la fonction de routage [30], et le contrôleur de session ont été étudiés [31], DNS, IMS etc. d'autres fonctions réseau restent encore à virtualiser.

NFV considère la cloudification des réseaux des opérateurs à travers la virtualisation de ses fonctions réseau (VNF) en définissant un modèle appliqué sur trois niveaux de services :

- *Virtual Network Function as a Service (VNaaS)* : offrant un accès vers une fonction réseau virtualisée qui est instanciée sur un point de présence du réseau ou de l'infrastructure NFV (NFVI). Elle représente à l'instar du SaaS, des services clés en main, mais au niveau du service de Media Delivery.
- *Virtual Network Platform as a Service (VNaaS)* : offrant une plate-forme pour fournir les fonctions réseau virtualisées (NAT, firewall, etc.) en outils de programmation et permettant leur composition.
- *Network Function Virtualization Infrastructure as a Service (NFVIaaS)* : NFVI, comme son nom l'indique, consiste en l'infrastructure physique virtualisée. Ce modèle offre des environnements d'exécution pour les fonctions réseau virtualisées et ainsi des capacités et des fonctionnalités de calcul et de stockage au même titre que l'IaaS du cloud computing.

Avec la virtualisation des fonctions réseaux on devrait aussi gérer le déploiement des applicatifs réseaux (les appliances des réseaux virtuels) sur une infrastructure virtualisée. Ces fonctions réseau virtualisées seront installées sur des machines virtuelles sur une infrastructure virtuelle/virtualisée contenant des ressources dans du hardware banalisé. Mais, ne s'agit-il pas une fois de plus d'une virtualisation des équipements réseau, indépendamment de la fonction que ces équipements rendent ?

Cette «cloudification» du réseau autorisera aussi la composition de services NaaS avec les autres catégories de services du modèle Cloud telles que SaaS, PaaS ou IaaS. De façon équivalent au modèle NFV, en appliquant le modèle du Cloud au niveau réseau, la modélisation du NaaS se décompose ainsi en plusieurs couches : NSaaS (*Network Software as a Service*), NPaaS (*Network Platform as a Service*) et NIaaS (*Network Infrastructure as a Service*).

- Le NSaaS propose, à l'instar du SaaS, le service Network Delivery pour une demande d'application spécifique provenant du SaaS/PaaS informatique.

- Le NPaaS propose la composition des services réseaux (routage, adressage, NAT, firewall, etc.) et l'appel d'un applicatif dédié (par exemple des services IT, localisation, facturation, autorisation, etc.) à une fonction réseau nécessaire pour fournir le service de délivrance de média du NSaaS. Le NPaaS programme ainsi le plan d'usage (data plane) du réseau virtuel à déployer.
- Le NIaaS consiste dans la virtualisation de l'infrastructure réseau (IaaS) dont le NaaS programme le plan de contrôle et déploie le réseau virtuel fournissant le service de Media Delivery demandé.

En considérant ces différentes couches, le NaaS pourra proposer un menu de services réseaux à la demande et composables.

#### 2.2.4 Softwarisation du réseau

Pour la softwarisation du réseau, le SDN [1] est le candidat pour une architecture simplifiée et agile à gérer. Il favorise l'évolution dans le Cloud Networking, en permettant l'introduction de la programmabilité du plan des données et en le séparant du plan de contrôle. Cette approche permet au réseau de répondre rapidement aux changements de configurations pour traiter des flux de données différents grâce à une centralisation de la logique (de routage) réseau dans un contrôleur [32]. Dans cette recherche de simplification, le SDN permet également aux opérateurs de s'affranchir des effets de dépendance avec les fournisseurs d'équipements réseau qui développent leurs propres systèmes, souvent spécifiques et nécessitant des adaptations pour assurer l'interopérabilité / la compatibilité.

C'est cette approche que nous qualifions de *Softwarisation du réseau*.

Associé à la virtualisation réseau, le SDN permet de construire des réseaux virtuels en reposant sur l'abstraction des équipements/ éléments de réseau où le contrôleur SDN ayant une vue globale de l'infrastructure de réseau physique, spécifie différentes topologies logiques pour plusieurs utilisateurs (les différents tenants) en programmant ces équipements réseau. Il s'agit d'une virtualisation de réseau basée sur la définition des chemins réseau pour les flux d'un utilisateur (application/tenant). Cependant, bien que cela soit fait dynamiquement grâce à l'interface centrale du SDN, et le protocole OpenFlow, le «SDN-based network virtualization» reste un partage/ une mutualisation de l'infrastructure réseau au niveau des équipements réseau qui traitent des flux appartenant à différentes applications.

### 2.3 Conclusion

Comme nous l'avons présenté dans ce chapitre, l'évolution du Cloud Networking passe à travers des approches visant à mettre en place une programmabilité et une agilité au niveau de la plate-forme réseau en considérant un point de vue fonctionnelle. Les différentes propositions (virtualisation des équipements, cloudification ou softwarization du réseau) restent basées sur l'application des techniques permettant un partage de l'infrastructure sans considérer forcément un traitement différencié par rapport aux flux transportés par les réseaux virtuels.

En effet, indépendamment de la cloudification du réseau, nous devons considérer un modèle de service supplémentaire pour représenter le Cloud Networking. Le NaaS est donc un concept récent qui introduit la vision du «Réseau comme un Service», de la même façon que des services rendus aux autres niveaux comme le PaaS ou le IaaS. Mais la notion de NaaS ne se réduit pas aux services de connexion entre des noeuds du réseau physique. Le NaaS prend en considération l'ensemble de services réseau et est ainsi capable d'offrir des réseaux virtuels à la volée en accord avec les besoins du flux applicatif.

Nous pensons que la fonction de la virtualisation ne doit pas se limiter aux niveaux des équipements réseaux, que se soient des nœuds (ex. routeurs, commutateurs, points d'accès) ou des circuits (ex. VC, Vlan, VPN), mais doit englober aussi le plan de contrôle réseau associé à chaque flux. Ce dernier comprend le système d'exploitation réseau (le NOS) qui prend les décisions de gestion et de contrôle des flux. La virtualisation du NOS permet de créer des réseaux virtuels personnalisés offrant des services réseaux appropriés aux types et aux exigences des flux à transférer. Comment alors procéder pour virtualiser les fonctions du NOS? Comment les personnaliser en fonction des exigences des flux ?

La section suivante décrit l'architecture NaaS qui contient les premiers éléments de réponse à ces questions : Comment composer des services fournis par des fournisseurs différents ? Comment gérer les infrastructures et les ressources ? Comment introduire, contrôler la dynamique et la flexibilité dans les offres de service réseau ? Quels composants doivent être introduits pour gérer le fonctionnement des réseaux virtuels dans un environnement global et intégré.

# 3

## Une architecture pour la personnalisation des services réseau

Comme nous l'avons présenté dans les sections précédentes, le Cloud impose un nouveau modèle, le NaaS (3.1), qui représente le Cloud Networking dont l'architecture réseau prend en compte les nouvelles contraintes et met à profit les nouvelles technologies. Le NaaS a pour objectif (3.2) la définition des réseaux virtuels que nous voulons personnaliser à travers une composition des fonctions réseaux (NFV). Mais pour que ceci soit une réalité il nous faut d'abord définir les constituants des réseaux virtuels (3.3) afin d'introduire le maximum d'automatisme et de dynamique. Nous pourrions ainsi composer les services réseaux à la volée selon les besoins.

### 3.1 NaaS (virtualisé (NFV) et programmable (SDN))

Aux vues de l'analyse ci-dessus on peut dire que la virtualisation (i), la programmabilité (ii) et l'approche SOA (iii) sont les trois ingrédients majeurs du futur NaaS. Ils pourront apporter des avantages autant conceptuels qu'opérationnels, affectant à la fois les processus orientés ressources (comme l'approvisionnement, la gestion, et le monitoring du réseau), et les processus orientés services (tels que la vente, la configuration et l'activation des nouveaux services).

1. La virtualisation s'applique au niveau des équipements (infrastructure du NaaS), au niveau du réseau (réseaux virtuels découplés de l'infrastructure) et au niveau de composants (plate-forme virtuelle offrant des NFV).
2. La programmabilité s'applique au niveau des équipements qui deviennent personnalisables selon les besoins des flux à traiter (SDN), au niveau du plan de contrôle qui

expose les fonctionnalités du réseau sous-jacent à travers l'abstraction des ressources, permettant leur composition et leur réalisation.

3. L'approche SOA appliqué au réseau permet de considérer l'automatisation de l'approvisionnement du réseau et faciliter ainsi la maintenance et la création des nouveaux services à la demande. Encore faut il des services ayant toutes les propriétés préconisées par SOA en particulier des composants «stateless» et des liens lâches [33] permettant une composition personnalisée. Les auteurs en [21] présentent une étude complète sur la façon dont les principes de l'architecture orientée services (SOA) peuvent être appliqués à la virtualisation de réseau, permettant au paradigme NaaS, d'être appliqué dans les télécommunications et le futur Internet.

En intégrant toutes ces approches, notre proposition de NaaS virtualisé et programmable basé QoS doit :

- (A) être en mesure de fournir des services de réseau à la demande et d'offrir des services de réseau appropriés et adaptés pour faire face aux exigences des flux,
- (B) appliquer l'approche basée sur la QoS pour refléter l'accord entre la QoS SaaS / PaaS exigé en fonction de la qualité de service offerte par IaaS, et
- (C) configurer le service de délivrance des médias.

Le NaaS a ainsi différents objectifs :

- Fournir des APIs facilitant la sélection de services.
- Fournir les services réseau à la demande (*Network Service Delivery*) pour traiter les exigences des différents flux applicatifs que sélectionnera le demandeur (architecte et ou développeur).
- Contrôler le service de *Media Delivery* de chaque réseau virtuel et assurer un traitement continu et dynamique.
- Gérer les réseaux virtuels à déployer en fonction des possibilités de l'infrastructure.
- Placer les réseaux virtuels dans l'infrastructure physique.

## 3.2 Description de l'architecture NaaS

Notre proposition NaaS architecture fonctionnelle vise à fournir une réponse pour donner une solution de réseau virtuel personnalisé en tant que service. La figure 3.1 montre une vue globale de l'architecture d'un NaaS intégrant le SDN (séparation plan de contrôle, plan de données), les NFV (fonctions réseau dans des VMs) et la virtualisation réseau au niveau flux.

Notre architecture applique l'approche SDN en définissant un plan de contrôle virtualisé et personnalisé pour chaque réseau virtuel. Au même titre que l'on décrit les appliances de machines virtuelles, on va décrire les slices des réseaux virtuels avec leur contrôleur SDN, leur VEOS associé (*Virtual Network Operating System*). Chaque réseau virtuel proposé est défini par trois plans : gestion, contrôle et données. Le plan de gestion permet de gérer les ressources

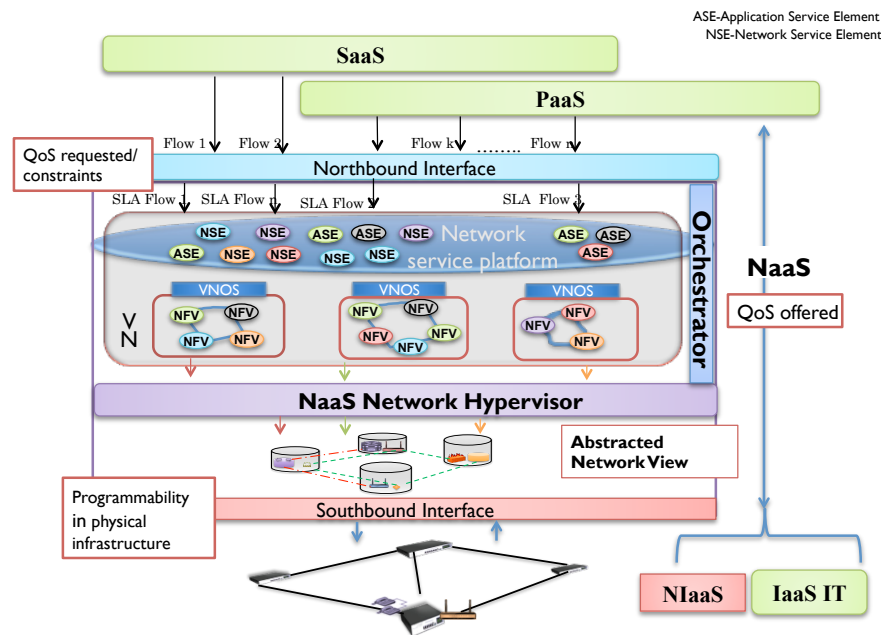


Figure 3.1 – Architecture NaaS virtualisée et programmable

du réseau, le plan de contrôle tient compte des contraintes et établit le routage (par réseau virtuel) et le plan de données contiendra les tables de forwarding. La virtualisation ne s’arrête pas ainsi aux équipements, mais elle englobe aussi le plan de contrôle réseau associé à chaque flux, construit avec des fonctionnalités de contrôle personnalisables afin d’offrir un service de Media Delivery en accord aux exigences de qualité de service des flux transférés. Ce dernier prend les décisions de gestion et de contrôle des flux, en regroupant les services réseau et de contrôle (NFVs) de chaque réseau virtuel.

Notre proposition de NaaS intègre l’approche NFV, en considérant les services réseaux comme des fonctions virtualisées et composables. Les NFVs sont proposés dans la plate-forme du NaaS (NPaaS) fournissant un catalogue de services réseau.

Ainsi, lors de la construction du réseau virtuel, la programmation du VNOS est effectuée et les NFV(s) sélectionnés permettent de définir les composants du réseau à être déployés (composition et sélection faite au niveau du NPaaS). Leur composition est vue à travers l’enchaînement de composants réseau. Chaque composant implémentant une NFV.

Nous complétons notre vision du NaaS en appliquant l’approche SOA pour permettre une composition personnalisée des services réseaux. Les composants sont modélisés comme étant des SCC (*Self-Controlled Component*) (voir la section 3.3.1) incorporant un auto-contrôle.

Le dimensionnement du réseau virtuel s’effectue en prenant en compte les contraintes de qualité de service associées et en garantissant ainsi la distribution des composants réseaux sur des nœuds virtuels sélectionnés en accord avec le SLA (*Service Level Agreement*).

Pour prendre en considération l’ensemble des approches mentionnées, l’architecture NaaS compte avec les éléments suivants :

- Deux interfaces. L’interface nord qui permet de capturer les demandes et exigences

des flux à traiter. L'interface sud qui permet au NaaS de communiquer avec le NIaaS pour lui permettre la programmation des équipements. NaaS détermine en fonction des besoins des flux quelles VNFs sont les plus appropriées.

- Une plate-forme de services qui seront décrits dans un catalogue. NaaS permet ainsi d'offrir des services réseaux composables dans sa couche NPaaS.
- Un contrôleur par réseau virtuel que nous appelons *Virtual Network Operating System* (VNOS). VNOS doit traiter et provisionner selon la spécification du réseau virtuel et les remontées de l'infrastructure physique.
- Un hyperviseur réseau qui au même titre que l'hyperviseur machine, jouera au niveau du réseau le rôle de gestionnaire du NaaS. A travers une couche d'abstraction de l'infrastructure physique sous-jacente, il contrôle et gère les éléments réseaux.
- L'orchestrateur NaaS qui automatise le déploiement continu des plans (contrôle, données et gestion) dans les éléments du réseau virtuel en fonction des informations recueillies sur l'état sur ces mêmes éléments pendant la phase d'exploitation.

Le NaaS configure ainsi le service de *Network Delivery* le plus approprié pour faire face à la qualité de service en relation au traitement du média demandé (abstraction faite à travers la couche NSaaS). La prestation de service offerte par le NaaS est en relation avec la sélection des services réseau (NFVs) choisis. NaaS offre ainsi un déploiement du réseau virtuel au niveau du NIaaS.

### 3.3 Les nouveaux éléments constitutifs

Pour satisfaire ces objectifs les noeuds (3.3.1) et liens (3.3.2) du NaaS doivent satisfaire un certain nombre de propriétés.

#### 3.3.1 Des composants des services réseau auto-contrôlables

Dans le projet «OpenCloudware» [34] nous avons appliqué les propriétés SOA et le modèle de QoS pour modéliser les composants de service du Cloud et du Cloud Networking. Ces composants sont décrits par leurs aspects fonctionnels et non fonctionnels (QoS) dénommés SCC (*Self-Controlled Component*) [35]. Cette modélisation permet aux fournisseurs NaaS de proposer des services réseaux en fonction de la QoS. En considérant les propriétés SOA [33] [36], les services SCC sont caractérisés pour être :

- Descriptifs : en utilisant des méta-données la description du service est faite en le découplant de son implémentation.
- Gérables : les services doivent être conscients de leurs états, et ils doivent pouvoir le partager pour faciliter la gestion.
- Enregistrables : les services doivent pouvoir être utilisés par des tiers.
- Invocables : à travers la définition des API ouvertes/ standards.

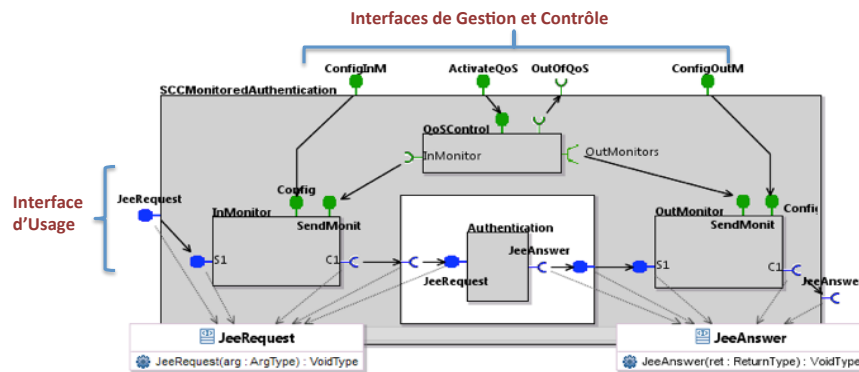


Figure 3.2 – Structure d'un composant SCC (Self-Controlled Component)

Nous appliquons ces propriétés pour modéliser les composants de services réseau. Les fonctions réseaux sont ainsi vues comme un ensemble de services qui peuvent être composés. Comme le montre la figure 3.2, un composant SCC définit :

- Un contenu fonctionnel (business).
- Deux natures d'interfaces externes (client et serveur) utilisés pour communiquer avec l'environnement.
- Trois types d'interfaces : l'interface fonctionnelle, l'interface de gestion, et l'interface de contrôle.
- Une membrane pour les aspects non-fonctionnels (QoS et gestion).

La notion de membrane provient du modèle GCM (*Grid Component Model*) [37] qui permet de définir un cadre de haut niveau pour le développement des capacités de gestion. Ainsi, les composants de surveillance sont hébergés dans la membrane de chaque composant de service. La logique de structuration de la membrane permet la réutilisabilité et la généralité de nos composants. On y trouve pour l'autocontrôle un triptyque «InMonitor, OutMonitor et un contrôleur de QoS». Ce dernier s'assure que le contrat de QoS est toujours respecté par le service.

Pour les interfaces fonctionnelles, elles sont au nombre de deux : «JeeRequest» et «JeeAnswer». Une interface serveur qui comprend les fonctions de traitement (opérations) qui peuvent être exécutées par le composant de service. Une interface cliente qui effectue l'invocation sur le prochain service de la chaîne, et la transmission de son résultat courant pour un traitement ultérieur ou pour l'exploitation du résultat final.

Pour les interfaces de gestion qui sont des interfaces serveurs non-fonctionnelles (ConfigInM, ConfigOutM, et ActivateQoS, dans la figure 3.2). Elles supportent les requêtes nécessaires pour gérer la configuration des composants non fonctionnels de la membrane.

Pour l'interface de contrôle c'est une interface client non-fonctionnelle (OutOfQoS dans la figure 3.2). Elles supportent les notifications envoyées par le contrôleur de QoS. Il émet *InContract* notifications tant que le comportement est conforme au contrat, sinon il déclenche *OutContract*. L'absence du signal *InContract* sera utilisée par le gestionnaire pour détecter les



défaillances graves de la composante SCC. Le contrôleur de QoS est intégré à chaque composant de service SCC. Il permet de signaler les violations de contrat (*Out-contrat*) quand elles se produisent, ainsi le gestionnaire prend les décisions opérationnelles appropriées.

Les composants de mesure (monitors) permettent le mesurage des données nécessaires au calcul des critères pour le contrôle du comportement (capacités non fonctionnelles conformes au modèle de QoS) du SCC.

Avec l'intégration de ces composants, nous obtiendrons un NaaS virtualisé et programmable basé QoS, mais au niveau du réseau ces approches ne sont pas encore appliquées dans leur totalité.

### 3.3.2 La Session réseau

Nous définissons la «session réseau» à travers des mécanismes d'orchestration qui combinent des efforts de composition horizontale et verticale. La session réseau regroupe tous les composants assurant le service réseau personnalisé d'un utilisateur, nous l'avons dénommée VPCN (*Virtual Private Connectivity Network*). Le NaaS supporte deux types de services (placés à des niveaux différents) : le *Network Service Delivery* et le *Media Service Delivery*. La figure 3.3 montre ces deux notions. Le *Network Service Delivery* sollicite le *Media Service Delivery* pour transporter le contenu des flux en adéquation aux contraintes QoS exprimées par l'appliquatif.

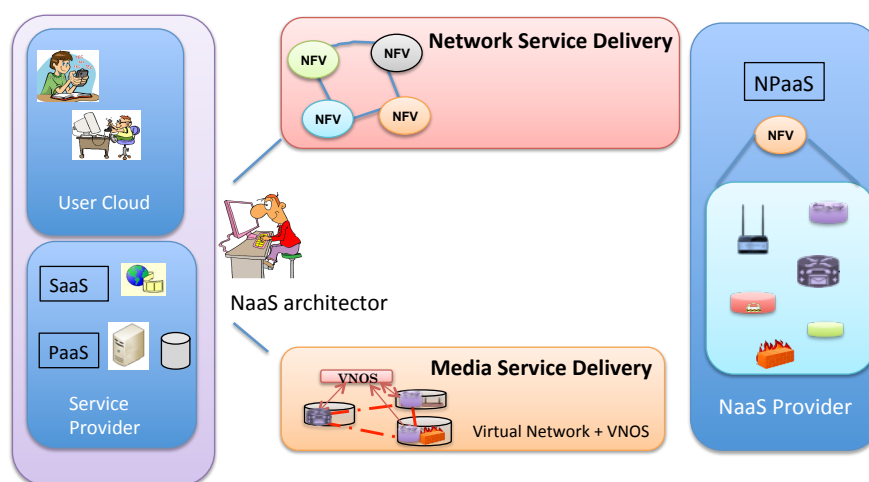


Figure 3.3 – La session réseau et les services : Network et Media Delivery

La logique du service réseau est décrite par le réseau virtuel correspond au *Network Service Delivery* (placé au niveau du NSaaS/NPaaS). Le plan de contrôle et le plan de données sont programmés en conséquence fournissant ensemble le *Media Service Delivery*. Le plan de gestion est aussi programmé à travers des composants de monitoring qui permettent de surveiller le comportement des compositions autant verticales que horizontales.

En d'autres termes, l'orchestrateur gère chaque session réseau qui se veut mobile pour suivre temporellement l'utilisateur et ainsi suivre la composition dynamique de son réseau virtuel en adéquation avec la logique de services applicatifs.

Les mécanismes d'orchestration intégrés dans l'architecture du NaaS permettent une composition de services mais aussi un contrôle et gestion permettant aux fournisseurs de services réseau une intégration autant verticale que horizontale des solutions. Mais étant donné qu'un réseau virtuel peut s'étaler sur plusieurs réseaux physiques sous-jacents, les demandes de placement doivent également être développés avec une information permettant une vue globale. Dans les environnements hétérogènes sous jacents, les demandes peuvent être contradictoires, les paradigmes de gestion sont nécessaires pour permettre aux fournisseurs d'infrastructure de manipuler globalement l'information. L'introduction d'une couche d'abstraction commune, avec un modèle d'information unique qui sera suivie par tous les logiciels de gestion, peut être une solution efficace [3] [38].

L'architecte du service global a besoin donc d'utiliser une information pertinente pour représenter le service de bout-en-bout. Pour ce faire, nous avons proposé de considérer plusieurs niveaux de visibilité (figure 3.4) pour bien distinguer les complémentarités des services rendus: équipement, réseau, service, utilisateur. Ces niveaux de visibilités représentent l'aspect vertical pour un service de bout-en-bout permettant d'introduire une abstraction des services rendus par niveau [39] [40]. Qu'en est il pour le NaaS dans un contexte virtualisé ?

En milieu virtualisé le niveau équipement contient les équipements physiques virtualisés (isolation et indépendance de l'OS). C'est sur cette infrastructure physique (NaaS) que seront placés les entités du réseau virtuel (NFV). Le niveau réseau aura deux vues : celle de l'interface nord qui représente le flux applicatif, et celle de l'interface sud qui représente la connectivité des composants à travers une vue abstraite de l'infrastructure physique. La figure 3.4 présente ces différents niveaux de visibilités en milieu virtualisé.

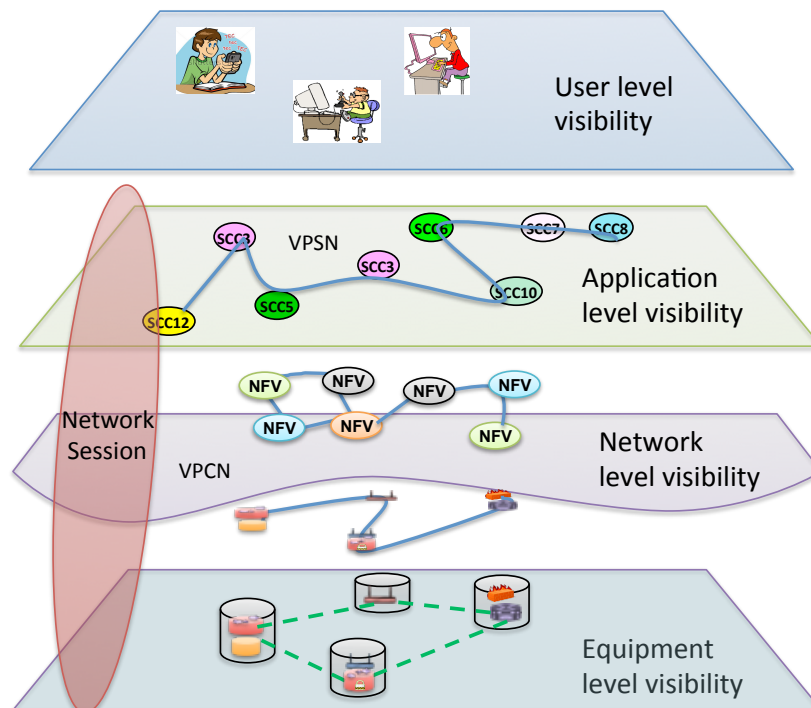


Figure 3.4 – La session réseau et les niveaux de visibilités en milieu virtualisé

Les éléments interconnectés à chaque niveau sont sélectionnés et pré-alloués en fonction de leur QoS offerte (représentée par leur valeur de conception dans notre modèle de QoS). La réservation réelle de ressources est faite lors de la phase d'exécution avec un provisionnement dynamique, basé cette fois sur la QoS courante. Pour le contrôle et la gestion nous utilisons les valeurs QoS seuils.

Une agrégation verticale vient compléter la description de bout-en-bout et nous permet de compléter la définition la session réseau. Cet aspect est représenté par l'interconnexion des services sélectionnés à différents niveaux. En utilisant notre modèle de QoS, la description verticale s'adapte d'une manière dynamique à tout changement dans les préférences de l'utilisateur ainsi que dans son contexte ambiant (mobilité, localisation, changement d'équipement, etc.) [41], [42]. Cette capacité de mesurer le comportement des entités à différents niveaux nous permet d'adapter dynamiquement le service rendu dans le niveau nécessaire. Par exemple si un changement se produit au niveau du service de réseau delivery (du, par exemple, aux changements de point d'accès au service réseaux à cause de la mobilité des utilisateurs) le service de média delivery est adapté en conséquence (par exemple à travers un redimensionnement du réseau virtuel déployé).

# 4

## Le cycle de vie du NaaS

Comme nous l'avons défini dans le chapitre 3, l'architecture NaaS propose le découplage entre les services réseau et l'infrastructure, fournissant ainsi un service de connectivité flexible, mais introduit aussi le défi de définir le service réseau à travers la composition dynamique des fonctions réseau virtualisées.

Le NaaS doit répondre à la personnalisation du service réseau, c-à-d, comment sélectionner les composants «réseau appropriés» en fonction de l'application ou OTT (*Over The Top*) demandés, et faire face au traitement d'un type de flux spécifique.

Tous ces éléments poussent à aller au-delà de l'intégration des nouvelles technologies, et au besoin des opérateurs à repenser la virtualisation de réseau, qui d'un point de vue de l'offre du réseau de l'opérateur en tant que service (*Network as a Service*) constitue un paradigme primordial. En effet, nous pensons que pour tirer profit des avancées tels que SDN et NFV, il faut non seulement maîtriser la virtualisation des équipements réseaux, mais il faut aussi bien appréhender la virtualisation réseau au niveau de la conception des réseaux virtuels et ainsi repenser la façon de les déployer pour qu'ils répondent aux demandes des utilisateurs. C'est ces deux niveaux de virtualisation, associés bien sûr à une forte capacité d'automatisation et à suffisamment d'informations (monitoring) sur le réseau, qui permettront le déploiement dynamique et efficace de réseaux virtuels pour les besoins de notre nouvel écosystème.

Toute cette analyse nous mène à proposer dans cette étude une nouvelle phase dans le cycle de vie appropriée au déploiement dans le NaaS, que nous avons dénommée «Network Virtual Deployment». Cette nouvelle phase met en exerce les mécanismes nécessaires pour un déploiement continu et à la demande des services réseau. La description complète de la proposition sera l'objet du chapitre 5.

### Les phases du cycle de vie des services réseau en milieu virtualisé

La figure 4.1 introduit notre proposition de changement dans le cycle de vie des services réseau en milieu virtualisé.

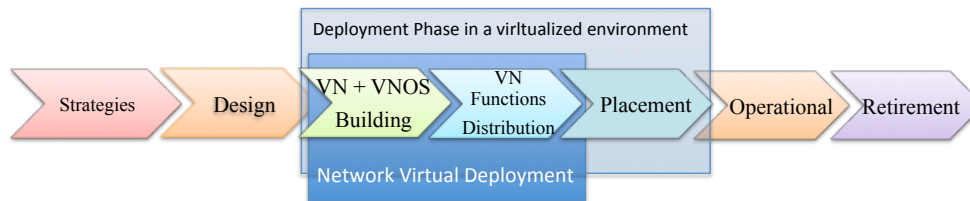


Figure 4.1 – Le cycle de vie des services réseau en milieu virtualisé

La Phase de Stratégie guide la mise en œuvre des plans d'action de l'entreprise en accord avec les objectifs de gestion et sa position sur le marché. Dans le cas des solutions actuelles de Cloud, les stratégies commerciales se doivent dynamiques et les entreprises doivent être en mesure d'adapter leurs offres en fonction des changements de marché. En particulier pour un fournisseur, le modèle business associé permet d'envisager des technologies différentes (disponibles et nouvelles) pour concevoir et déployer dynamiquement leurs réseaux virtuels, et rester ainsi flexible dans leur offres.

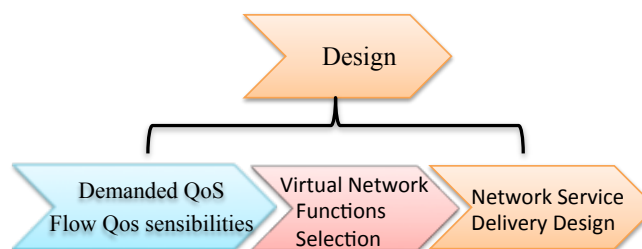


Figure 4.2 – La phase de design

La Phase de Design (figure 4.2) permet à l'architecte de concevoir le service selon les demandes des clients et le business model. Dans cette phase c'est d'abord le PaaS qui offre à travers son catalogue des services applicatifs, et qui se positionne par rapport aux possibilités de coopération avec des partenaires tiers. La tâche suivante de cette phase est de concevoir le réseau support des flux applicatifs en sélectionnant les fonctions de réseau virtuel appropriées (VNFs) en conformité avec les exigences des sensibilités des flux et la logique de service. On obtient un réseau de services dont les nœuds sont les composants réseau et les liens les chaînes entre ces composants. On définit la nature des échanges et la QoS entre les points d'accès des clients du réseau virtuel à déployer.

Dans un contexte IT reposant sur un réseau local (figure 4.3 (a)) nous trouvons ensuite la phase de déploiement qui est le processus de placement sur des équipements des composants d'une application. Le processus de déploiement consiste à définir les images disque qui vont être utilisés pour déployer les VM et qui revient à cette problématique de placement sur les équipements virtualisés.

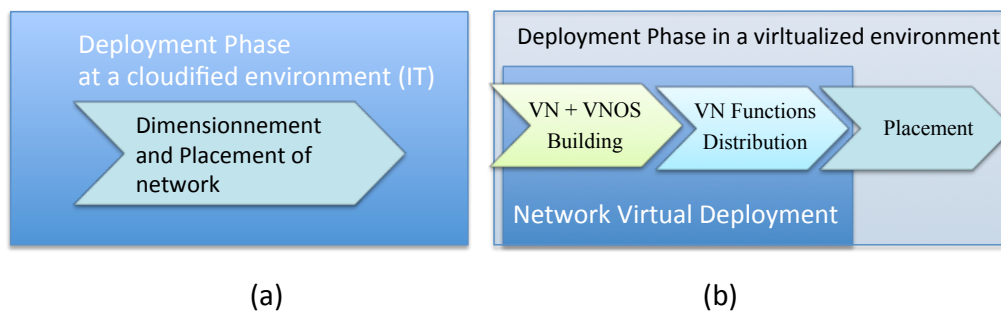


Figure 4.3 – La phase de déploiement dans : (a) un environnement IT, (b) proposée dans un environnement de virtualisation réseau

Dans un contexte de virtualisation réseau (figure 4.3 (b)), le NaaS, qui repose sur la séparation plan de contrôle/plan de données, l'abstraction des ressources physiques (SDN), et la virtualisation réseau à deux niveaux (chapitre 5), nous conduit à décomposer la phase de déploiement en deux phases, celle du *déploiement du réseau virtuel* et celle du *placement*. Sachant que la phase de placement permet de réaliser le mapping réseaux virtuels / abstraction infrastructure physique, nous allons nous concentrer sur le *déploiement du réseau virtuel*. La description du réseau virtuel à déployer doit être fournie par cette phase de déploiement virtuel.

La Phase d'exploitation (figure 4.4) correspond à la consommation réelle des services déployés. Dans cette phase, le fournisseur garantit la fourniture de ressources pour permettre les transactions liées à la consommation des services (la délivrance du service de bout-en-bout). Aussi les tâches de gestion doivent être mises en œuvre, permettant de vérifier en permanence le comportement des services déployés. NaaS doit garantir que le comportement du service de *Media Delivery* soit en accord avec l'accord SLA négocié lors de la phase de design (*Network Service Delivery*).

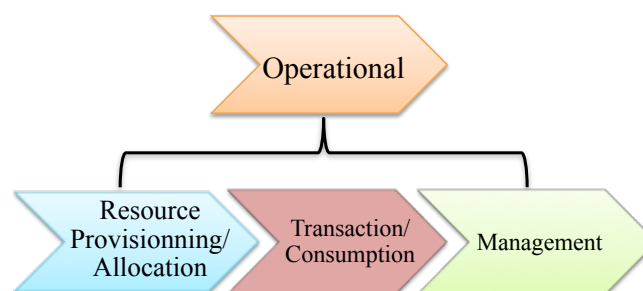


Figure 4.4 – La phase d'exploitation

La Phase de retirement permet de retirer le ou les services qui ne sont plus sollicités.

Nous allons nous concentrer par la suite sur la phase de déploiement de réseaux virtuels. Nous présentons dans le chapitre suivant la description de notre proposition.



# 5

## Vers un déploiement continu de services réseau à la demande : la dimension fonctionnelle

Le déploiement est une phase importante du cycle de vie des réseaux. La question que l'on s'est posée dans ce document est : n'est-il pas opportun de modifier / repenser le processus de déploiement de réseaux dans le nouvel écosystème impacté par l'adoption massive de la virtualisation? Pour ce faire nous avons d'abord relevé les changements introduits par le Cloud Computing et les bénéfices apportés (Chapitre 2). Ce qui nous a conduit à préciser la notion de virtualisation réseau que l'on doit différencier de la virtualisation des équipements de réseau, de la virtualisation des fonctions réseau ou encore de la virtualisation des circuits réseaux et à définir les caractéristiques attendues du NaaS (Chapitre 3). C'est ainsi que nous avons été amenés à distinguer deux problématiques pour la mise en œuvre des réseaux virtuels : celle du déploiement proprement dit au niveau des réseaux virtuels à un niveau du «Network Service Delivery» et celle du placement au niveau du réseau physique au niveau du «Media Service Delivery».

Le NaaS représentant le Cloud Networking (ou le service réseau pour n'importe quel OTT) doit fournir le service Réseau Delivery le plus adéquate aux flux applicatifs et superviser le Media Delivery. Pour ce faire, conformément aux concepts du SDN, le Média Delivery se fait dans le plan data qui est au niveau du N1aaS. Le service Network Delivery, lui, doit être conçu suite à la demande du SaaS ou du PaaS (qui eux assurent le Service Delivery E2E, au niveau applicatif) pour offrir le service réseau entre les points d'accès. C'est le réseau virtuel correspondant au Network Service Delivery qu'il faut pré-provisionner. C'est cette phase où



l'on doit prévoir et estimer les ressources selon les contraintes de la demande que nous appelons « déploiement de réseau virtuel » qui est en fait un déploiement virtuel puisqu'on fera le placement que dans un deuxième temps.

Nous décrivons dans ce chapitre notre proposition pour le déploiement de réseaux virtuels dans le cycle de vie des services réseau en milieu virtualisé. Cette proposition insiste sur le besoin d'adopter le meilleur contrôle de réseau et la meilleure gestion en se basant sur l'abstraction du Network Operating System. La phase de «Network Virtual Deployment» intègre une réponse à une demande de SLA dès la phase de déploiement. Dans un contexte de Network-as-a-Service, elle prend en compte les propriétés de flexibilité, d'adaptabilité et de dynamicité nécessaires et vitales à ce nouvel écosystème réseau.

## 5.1 Network Virtual Deployment

Le déploiement des réseaux virtuels par le NaaS se situe entre les demandes de l'API nord et le placement sur les ressources physiques à travers l'API sud. Nous définissons cette phase comme un déploiement à deux étapes : la phase de construction du VN et de son plan de contrôle associé (le VNOS), à partir de la vue fonctionnelle transmise par le design, et la phase de distribution de fonctions du réseau virtuel sur l'abstraction du réseau physique (NIaaS).

La figure 5.1 montre les différents plans associés aux différentes étapes impliquées dans le déploiement virtuel des réseaux virtuels. Dans le plan de service réseau (au plus haut niveau d'abstraction), nous trouvons la logique du service réseau global désiré (le VPCN).

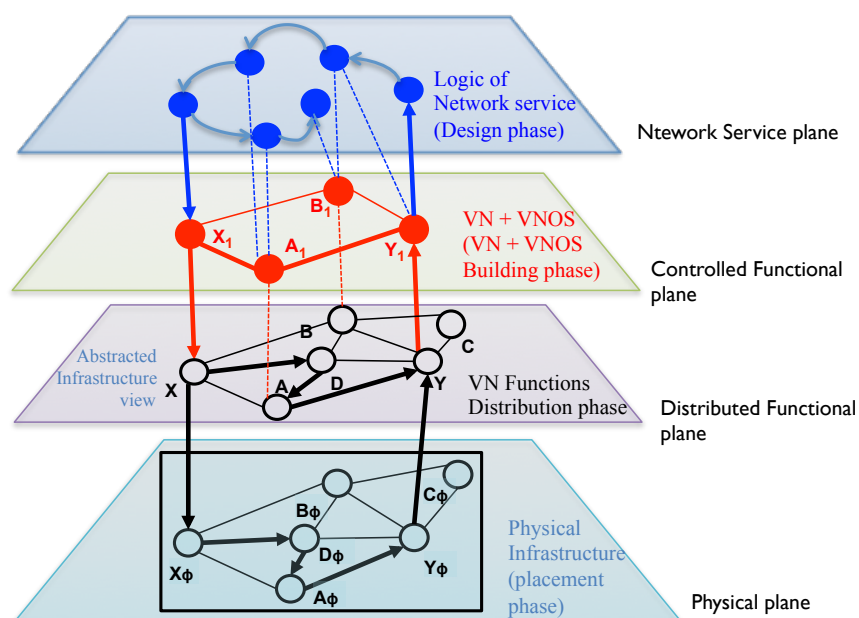


Figure 5.1 – La phase du *Network Virtual Deployment* et les différents plans des services du NaaS

### 5.1.1 Phase de *Building VN+VNOS*

Dans cette phase on s'intéresse à exprimer une traduction des besoins et exigences provenant de l'applicatif pour caractériser le comportement du réseau virtuel à déployer, c'est la composition de service répondant à la demande, à la logique du service réseau à offrir. Cette phase correspond à la caractérisation des éléments virtuels en fonction du flux de données à traiter, visant une traduction des besoins vers une virtualisation au niveau service réseau. Nous obtenons un réseau virtuel dont les composants réseau contiennent les NFVs nécessaires à la mise en œuvre et au transfert définis par le Network Service Delivery du Design.

Le plan de contrôle virtuel (le VNOS) est programmé dans cette étape. Il définit, entre autre, les contraintes des flux, les règles de routage (RIB-Rules Information Base) pour le service de Media Delivery. Ces RIB induiront les FIB (Forwarding Information Base) que l'on placera dans le data plan (le N1aaS). Le plan où se déroule cette phase (VN+VNOS) est appelé «Plan Fonctionnel Contrôlé».

### 5.1.2 Phase de *VN Functions Distribution*

Cette phase reprend le réseau virtuel et distribue les composants sur la vue représentant l'abstraction des ressources physiques (le Plan Fonctionnel Distribué). Nous obtenons un réseau avec tous les nœuds qui seront traversés (chemin), avec le bon protocole de routage (RIB), l'architecture et l'ingénierie spécifiques aux services supportés.

L'Hypervisor NaaS choisit les ressources virtuelles (nœuds et liens) pour l'ensemble des réseaux virtuels sur la vue abstraite de l'infrastructure réelle (basée sur les ressources négociées avec IaaS). NaaS incorpore aussi la description/traduction des exigences pour le placement des différents éléments du réseau virtuel permettant de satisfaire les demandes d'exigence des flux.

## 5.2 Une approche *QoS aware* pour le déploiement des réseaux virtuels à la demande

Pour avoir l'assurance que les exigences, demandées par l'application, soient effectivement répercutées lors du placement, nous proposons d'associer aux exigences fonctionnelles, les exigences non fonctionnelles, c'est à dire la QoS. La QoS permet de caractériser les besoins non-fonctionnels des flux applicatifs et ainsi définir les propriétés à assurer lors de leur traitement par le réseau à déployer. La phase de déploiement virtuelle est alors réalisée suivant l'approche QoS-aware, c'est à dire une déclinaison à tous les niveaux des exigences de QoS. L'application du modèle de QoS (figure 5.2) nous permettant de refléter un accord entre la QoS SaaS/PaaS et la QoS offerte par l'IaaS, autant pour définir le service de Network Delivery, que pour assurer le service de Media Delivery. Le NaaS tient compte de :

- La vue service : c'est le Network Service Delivery, qui caractérise l'offre du service de Media Delivery pour le traitement des besoins des flux.
- La vue placement en introduisant les contraintes de QoS lors du mapping vers l'infrastructure physique et qui conditionne le Media Service Delivery.

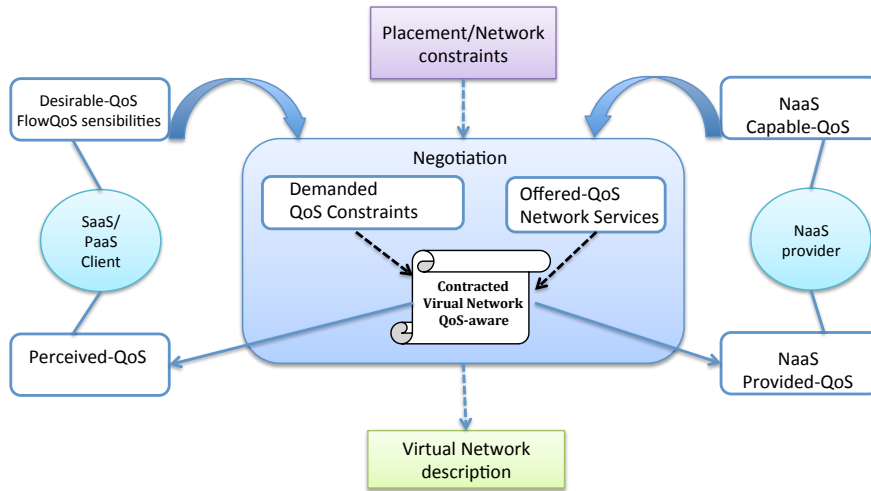


Figure 5.2 – Une approche QoS aware pour le NaaS

Les offres de services Naas dépendent de la capacité de ses ressources, négociée avec l’IaaS. Les ressources Naas sont représentées à travers la vue abstraite de l’infrastructure réseau. Notre approche QoS vient fournir un moyen de caractériser, contrôler et gérer le comportement des réseaux virtuels. Ainsi, le procédé proposé permet de considérer la virtualisation de réseau avec un deuxième niveau d’abstraction, qui permet de prendre en compte la nature des flux applicatifs. La démarche et les critères QoS proposées permettent d’exprimer les contraintes à respecter par les nœuds et de liens à chaque niveau d’abstraction, et pour tenir compte des exigences de qualité de service d’applications jusqu’au niveau physique.

Plus précisément, la phase de design se traduit par la définition du SLA avec ses aspects fonctionnels et non-fonctionnels, et la première phase du «Network Virtual Deployment» appelée «Building VN + VNOS» construit le réseau virtuel (VN) et le plan de contrôle (VNOS) avec les contraintes à garantir.

Contraintes de haut niveau	
<b>NodeAffinity</b> (v : set <VNE>)	Tous les nœuds virtuels qui appartiennent au groupe v doivent être mappés sur le même nœud physique
<b>NodeAnti-Affinity</b> (v : set <VNE>)	Tous les nœuds virtuels au sein du groupe v doivent être mappés sur des nœuds physiques distincts
<b>NodeIsolation</b> (v : set <VNE>)	Tous les nœuds virtuels VNEs au sein du groupe v ne doivent pas colocalisés avec d'autres VNEs existants
Contraintes de QoS	
<b>NodeTreatmentTime</b> (v1 : VNE, d : time)	Le temps de traitement de v1 doit être inférieur à d
<b>NodeAvailability</b> (v1 : VNE, $\alpha$ : rate)	Le taux de disponibilité de v1 doit être au moins égal à $\alpha$
<b>NodeReliability</b> (v1 : VNE, $\beta$ : rate)	Le taux de fiabilité de v1 doit être au moins égal à $\beta$
<b>NodeCapacity</b> (v1 : VNE, c : String, z : number)	La capacité de type c du nœud v1 doit être au moins égale à z

Figure 5.3 – Les contraintes de QoS et de placement au niveau de nœuds

Puis, en fonction de l’abstraction des ressources, la deuxième phase appelée «VN Functions Distribution» distribue les fonctionnalités réseau au sein des nœuds et de liens sélectionnés pour leurs «QoS offerte».

Contraintes de haut niveau	
<b>BanPath</b> (v1 : VNE, v2 : VNE, s : set<INNode>)	Le chemin entre v1 et v2 ne doit pas traverser les nœuds intermédiaires appartenant à s
<b>IsoPath</b> (v1 : VNE, v2 : VNE, nb : number)	Le nombre de nœuds intermédiaires entre v1 et v2 ne doit pas dépasser nb
Contraintes de QoS	
<b>MaxLinkDelay</b> (v1 : VNE, v2 : VNE, d : number)	Le délai d'acheminement entre un nœud virtuel v1 et un autre v2 ne doit pas dépasser d
<b>LinkDisponibility</b> (v1 : VNE, v2 : VNE, $\alpha$ : rate)	Le taux de disponibilité du lien reliant v1 et v2 doit être au moins égal à $\alpha$
<b>LinkReliability</b> (v1 : VNE, v2 : VNE, $\beta$ : rate)	Le taux de fiabilité du lien entre v1 et v2 doit être au moins égal à $\beta$
<b>LinkCapacity</b> (v1 : VNE, v2 : VNE, t : string, v : number)	La capacité de type t du lien (v1, v2) doit être supérieure ou égale à v
<b>LinkUtilisationRate</b> (v1 : VNE, v2 : VNE, E : rate)	Le taux d'utilisation d'un lien reliant v1 et v2 ne doit pas dépasser E
<b>LinkActivityRate</b> (v1 : VNE, v2 : VNE, $\theta$ : rate)	Le taux d'activité d'un lien (v1, v2) ne doit pas dépasser $\theta$

Figure 5.4 – Les contraintes de QoS et de placement au niveau des liens

Les figures 5.3 et 5.4 montrent un exemple de résultat de la modélisation des exigences réseau à considérer à travers la déclinaison des différentes valeurs de QoS à appliquer au niveau des nœuds et des liens du réseau virtuel puis à ceux du réseau physique lors du placement [22], [5].

### 5.3 Le processus de déploiement de réseaux virtuels

Le processus du déploiement de réseau virtuel se déroule comme suit (figure 5.5) :

- (1) Dans un premier temps PaaS/SaaS envoie au NaaS la description des besoins correspondant aux flux applicatifs, c'est le SLA.
- (2) Pendant la phase de Design une sélection est faite sur les services réseau virtualisés (VNFs) proposés par la plate-forme du NaaS et convenant au mieux à la logique de service applicatifs. Ces services sont exposés avec les aspects fonctionnels et non fonctionnels (QoS), ils sont donc choisis en fonction du SLA demandé.

Les informations concernant : (3) les VNFs sélectionnées, (4) les contraintes, ainsi que (5) la logique du service réseau sont envoyées à la phase de déploiement virtuel. Pendant la phase de building du VN, et la programmation du VNOS, les VNFs sélectionnées permettent de définir (6) les composants réseau à placer à la phase suivante. L'information associée à la description des flux permet aussi de déduire les contraintes QoS (7) à introduire pour garantir que le comportement global du réseau virtuel à déployer est prévu en accord aux besoins exprimés dans le SLA.

Pendant la phase de VN Functions Distribution, le dimensionnement (nombre de nœud du chemin) se fait en fonction des contraintes (8). Cette phase permet ainsi de tenir compte des contraintes pour le placement (9). Les nœuds virtuels choisis permettent ainsi de distribuer les composants réseaux virtuels définis dans la phase de building du VN. La description du réseau virtuel est ainsi complétée. La description du réseau virtuel est envoyée à la phase suivante, le placement. (10) Le processus de placement réalise le mapping vers l'infrastructure physique.

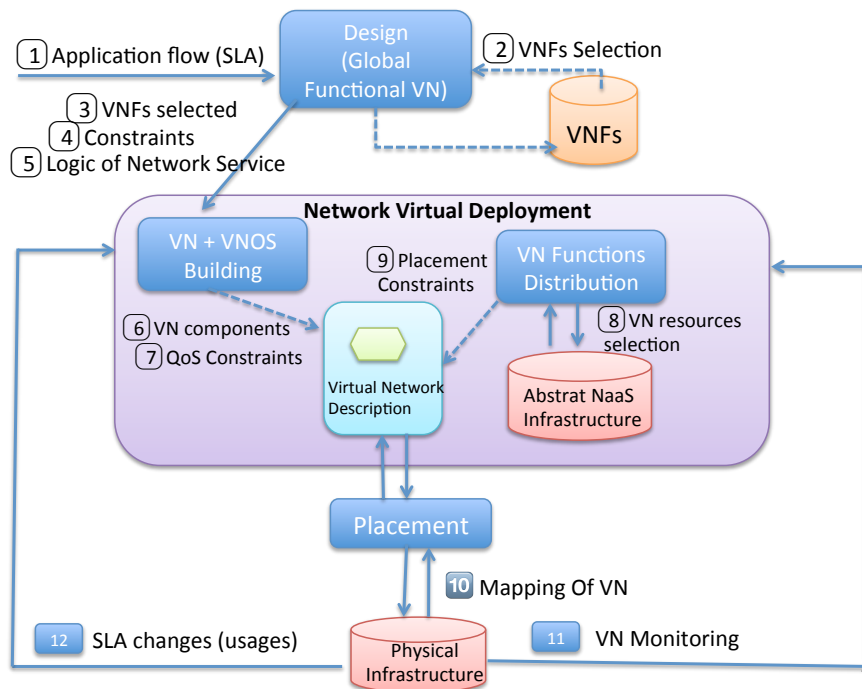


Figure 5.5 – Le processus de déploiement virtuel des réseaux virtuels.

(11) En fonction du monitoring qui renseigne sur les capacités courantes du réseau, le plan de contrôle révisé ses tables de routage et met à jour dynamiquement les tables de forwarding.

(12) En fonction des changements prévus dans le SLA, et pour prendre en compte : les préférences de l'utilisateur, la mobilité de ce dernier, la dégradation de la QoS de bout en bout, le VN+VNOS se reconfigure suite à une notification concernant un ou plusieurs de ces changements.

# 6

## Déploiement des réseaux virtuels : la dimension informationnelle

L'agilité du déploiement passe par la description, l'installation et la configuration de ses services. Un souci d'interopérabilité se pose lors de la construction de solutions dans des Cloud fédérés et distribués. Dans ces environnements il est nécessaire de mettre en place un modèle d'information et une description générique permettant la description des VMs, et les échanges liés au déploiement des solutions entre fournisseurs Cloud partenaires mais aussi pour permettre la gestion de l'environnement fédéré. Cette gestion comprend aujourd'hui des défis sur l'automatisation du déploiement et de la gestion des processus concernant les ressources virtuelles exploitées avec des approches SDN et NFV.

Pour permettre l'automatisation du déploiement et de processus de gestion, il est nécessaire de décrire l'ensemble des informations utilisées par ces processus. Décrire le service demandé, en prenant en considération des informations connexes autant aux services qu'à la description des ressources pour leur déploiement, devrait permettre de combler le fossé actuel existant entre l'information de la couche de services et celles des ressources physiques.

La description de la QoS offerte par le réseau et la façon, dont les éléments du réseau virtuel composent les services nécessaires, devraient également être décrites à travers le modèle d'information pour avoir une vision globale du réseau virtuel à déployer.

Nous proposons dans ce chapitre une première approche pour décrire notre modèle d'information, celle de la description du réseau virtuel à déployer à partir de notre expérience dans le projet OpenCloudware.

Nous présentons un aperçu sur les stratégies de standardisation (6.1), les avantages du langage OVF (6.2) et finalement notre proposition de description de réseaux virtuels (6.3).

## 6.1 Les travaux de standardisation

La standardisation pour permettre l'interopérabilité dans un contexte Cloud constitue une des principales problématiques actuelles. Les solutions proposées touchent différentes catégories des normes pour l'interopérabilité avec des objectifs différents, parmi d'autres nous pouvons mentionner [43], [44] :

- Modélisation de service: Open-SCA (Service composition et interaction), USDL / SysML / CloudML (services multi-vues), EMMML (mashups)
- Interfaces de service: OCCI (gestion de l'infrastructure), CIMI (gestion des infrastructures), EC2 (standard de-facto), TOSCA (portabilité), CDMI (gestion des données)
- Infrastructure et déploiement: OVF (machines virtuelles)
- Virtualisation des services et du plan de contrôle réseau : VNF, ACTN

Pour les aspects réseaux, le placement des NFVs et des VMs concerne aujourd'hui de plus en plus des environnements distribués, et avec la participation de plusieurs fournisseurs. Il devient donc nécessaire de représenter et partager une vue globale de l'information des composants virtuels et physiques pour contrôler/gérer le déploiement en continu. L'information concernant le déploiement commence à être traitée dans quelques communautés de recherche tels que l'IETF avec le modèle ACTN (*Abstraction and Control of Transport Networks*) [45], l'ETSI [29] avec son Framework pour NFV, et SDN et l'ONF [46] et l'ETSI [47] avec le placement des NFVs sur des équipements standards.

## 6.2 OVF : un langage orienté déploiement

La façon de décrire les services est donc un point clé à traiter dans le processus de déploiement. Trouver une façon standard de décrire les informations de déploiement est un besoin basique pour ce processus. OVF (*Open Virtualization Format*) a été proposé à cet effet par la DMTF (Distributed Management Task Force) [48].

OVF propose une spécification neutre vis-à-vis de l'hyperviseur ; qui se doit d'être performante, extensible aux futures évolutions des plate-formes et ouverte pour en assurer son indépendance. Elle définit les principes d'assemblage et de distribution des appliances virtuelles composées d'une ou plusieurs machines virtuelles.

OVF est une spécification définissant un paquetage portable et de format neutre pour les machines virtuelles. Le package OVF est composé de différents fichiers [49] : OVF descripteur, OVF manifeste, certificat OVF, les fichiers images de disque et les fichiers de ressources supplémentaires. Le OVF descripteur est défini par un format de méta-données XML associés aux machines virtuelles. OVF descripteur met l'accent sur les aspects de déploiement, et sur la description des ressources matérielles et logicielles dans les machines virtuelles. La figure 6.1 montre un exemple du contenu du package OVF.

Dans la proposition actuelle du descripteur d'OVF, une section est prévue pour décrire les réseaux logiques entre les machines virtuelles. Cette section est appelée : <NetworkSection> (voir figure 6.2).

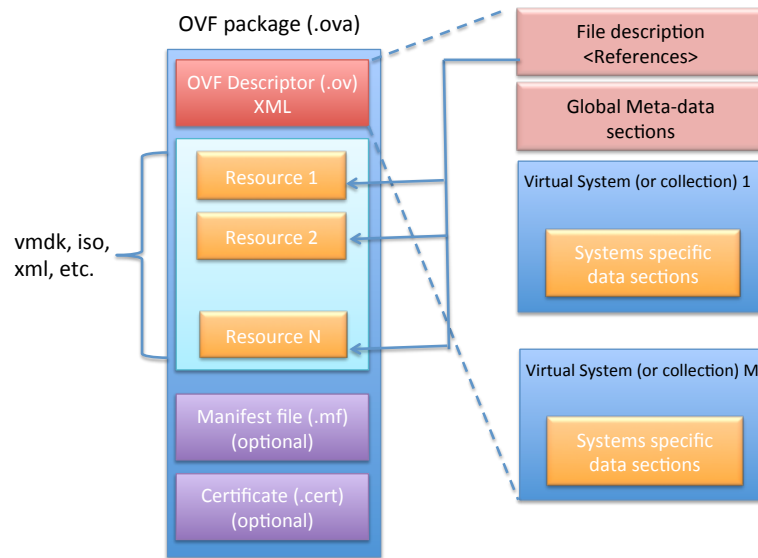


Figure 6.1 – Exemple d’un package OVF

```

<ovf:Envelope>
  [...]
  <ovf:NetworkSection>
    <Info>List of logical networks used in the package</Info>
    <Network ovf:name="VM Network">
      <Description>The network that the service will be available
on</Description>
    </ovf:NetworkSection>
  [...]
  <ovf:VirtualSystem>
    <ovf:ProductSection>
      [...]
    </ovf:ProductSection>
    <ovf:VirtualHardwareSection>
      [...]
    </ovf:VirtualHardwareSection>
  [...]
</ovf:VirtualSystem>
[...]
```

Figure 6.2 – Positionnement de la &lt;NetworkSection&gt; dans une spécification OVF

Différentes versions de OVF ont été proposées; la plus récente est la V2.0.1 (décembre 2013) [50], elle introduit le concept des profils de ports réseau en définissant la section <NetworkPortProfile>. La figure 6.3 montre un exemple de définition de la section <NetworkPortProfile> ([50]).

La section *NetworkPortProfile* est définie par une extension de schéma XML, qui utilise les éléments XML provenant des propriétés définies dans le modèle d’information CIM [51] (en particulier la classe *EthernetPortAllocationSettingData*). *NetworkPortProfile* vise à donner une représentation cohérente des attributs du réseau lors du déploiement et la gestion des composants des réseaux virtuels. Il est basé sur la définition d’un profil de port Ethernet (*EthernetPortItem*) qui est proposé comme étant un nouvel item dans la section <VirtualHardwareSection>. L’*EthernetPortItem* étend la capacité de gestion pour le réseau virtuel en ajoutant la représentation d’un port Ethernet, son contrôleur associé et les interfaces Ethernet. L’extension



```

<!-- Describes all networks used in the package -->
<NetworkSection>
  <Info>List of logical networks used in the package</Info>
  <Network ovf:name="VS Network">
    <Description>The network that the VSs connect to</Description>
    <NetworkPortProfile>
      <!-- Network port profile describing bandwidth reservation. Network port profile is identified by UUID. -->
      <Item>
        <epasd:AllocationUnits>bit / second * 10^9</epasd:AllocationUnits>
        <epasd:ElementName>Network Port Profile 1</epasd:ElementName>
        <epasd:InstanceID>1</epasd:InstanceID>
        <epasd:NetworkPortProfileID>aaaaaaa-bbbb-cccc-dddd-eeeeeeee</epasd:NetworkPortProfileID>
        <epasd:NetworkPortProfileIDType>3</epasd:NetworkPortProfileIDType>
        <epasd:Reservation>1</epasd:Reservation>
      </Item>
    </NetworkPortProfile>
  </Network>
</NetworkSection>

```

Figure 6.3 – Définition dans OVF de la section &lt;NetworkPortProfile&gt;

proposée met l'accent sur les informations représentant la réservation de la bande passante et les entités paires au niveau MAC/VLAN autorisés. La figure 6.4 montre un exemple d'*EthernetPortItem* ([50]).

```

<EthernetPortItem>
  <epasd:Address>00-16-8B-DB-00-5E</epasd:Address>
  <epasd:AllocationUnits>bit / second * 10^9 </epasd:AllocationUnits>
  <epasd:Connection>VS Network</epasd:Connection>
  <epasd:Description>Virtual NIC</epasd:Description>
  <epasd:ElementName>Ethernet Port</epasd:ElementName>
  <epasd:InstanceID>3</epasd:InstanceID>
  <epasd:NetworkPortProfileID>aaaaaaa-bbbb-cccc-dddd-eeeeeeee</epasd:NetworkPortProfileID>
  <epasd:NetworkPortProfileIDType>3</epasd:NetworkPortProfileIDType>
  <epasd:Reservation>1</epasd:Reservation> <epasd:ResourceType>10</epasd:ResourceType>
  <epasd:VirtualQuantityUnits>1</epasd:VirtualQuantityUnits>
</EthernetPortItem>

```

Figure 6.4 – Définition dans OVF d'un &lt;EthernetPortItem&gt;

Ainsi les propositions actuelles dans la version 2 d'OVF permettent la représentation des informations de réseau basées sur les exigences physiques tels que la bande passante et les ports utilisés au niveau 2. Mais il y a encore un manque en ce qui concerne la description complète du réseau virtuel (la description du service associée au traitement des flux par le VN) et les propriétés à décrire afin d'introduire la notion du «Réseau en tant que service».

Différents travaux de recherche ont étudié les possibilités d'extension de OVF. Les auteurs dans [49] et [52] proposent un langage de définition de service basée sur OVF pour faciliter le déploiement d'applications Internet dans les IaaS en incluant des règles d'élasticité. Les propositions sont basées sur la définition des nouvelles sections avec le langage OVF. Une des sections propose des balises KPI pour permettre des opérations de monitoring. L'autre section définit les règles d'élasticité qui, faisant usage de la KPI, indiquent comment le service doit être mis à l'échelle. Une autre proposition est la définition des macros dans la <ProductSection> dont les OVF peut décrire l'ensemble de l'installation IaaS requis pour un service.

Dans [53] les auteurs introduisent de nouvelles abstractions sous forme d'extensions de OVF pour définir le langage «Manifest Language» utilisé dans le cadre du projet RESERVOIR [54]. OVF est utilisé dans ce cas pour définir la syntaxe abstraite du langage décrivant les éléments de base et de leurs attributs. Une autre extension de OVF, appelée OVF ++ (*Open Virtualization Format ++*) a été proposée dans le cadre du projet 4CaaS [55].

Vu les possibilités d'extension de langage OVF, et la portabilité des packages OVF dans les systèmes virtuels OVF se présente aujourd'hui comme le candidat approprié pour décrire de nouvelles informations sur des réseaux virtuels dans des environnements virtualisés.

### 6.3 Une description des réseaux virtuels pour un déploiement dynamique

Nous avons utilisé OVF ++ dans le cadre du projet de OpenCloudware [34] [56] pour décrire l'architecture complète du système à déployer. La spécification de la section d'architecture est basée sur le langage ADL (*Architecture Definition Language*). OVF ++ intègre la description des composants de l'architecture en utilisant le modèle de composant Fractal. OVF ++ est utilisé pour décrire à la fois les ressources matérielles des machines virtuelles et les composants logiciels pour déployer l'architecture.

Le projet OpenCloudware a choisi de fournir une description dans OVF ++ pour le module de déploiement / placement. Une réalisation a été faite pour une architecture à trois niveaux appelée SPRINGOO qui définit trois serveurs (Apache, Jonas et celui de base de données).

Nous décrivons dans cette section, par analogie à l'applicatif notre approche pour décrire l'architecture du réseau virtuel. Notre proposition de description est basée sur l'utilisation d'OVF++. Notre proposition permet de compléter la vue actuelle d'un réseau virtuel, basée juste sur la virtualisation des équipements (switch, routeurs, commutateurs, etc) et de ressources (carte Ethernet, bande passante), avec une vue du service offert. Cette vue est caractérisée par les contraintes de QoS et de placement, ainsi que par la description des services réseaux liés aux NFVs contenus dans les nœuds du réseau virtuel.

Pour la phase de déploiement du VN, la description proposée permet au NaaS de décrire le réseau virtuel en conformité avec la qualité de service demandée. Pendant la phase d'exploitation, cette information est utilisée pour les processus de gestion. Le processus de monitoring peut utiliser cette information pour vérifier que le comportement actuel des éléments de réseau concernés est en conformité à la configuration de déploiement.

Nous décrivons dans cette section la Meta-information permettant de spécifier la nouvelle section appelée <VirtualNetworkSection> qui vient compléter la section actuelle proposée par la norme, le <NetworkSection>.

Table 6.1 – Extension OVF proposée

Extension proposed	Problem to deal	Extension type
VirtualNetworkSection	Description of QoS behavior of virtual networks	New Section at current <NetworkSection> Definition of new elements and attributs

Notre proposition décrit les éléments du réseau virtuel (le noeuds, les liens, et les contraintes réseaux) au niveau de la section proposée : le <VirtualNetworkSection> (voir tableau 6.1).

Nous appliquons notre démarche QoS-aware pour introduire des nouvelles informations de description du service réseau offert par la NaaS, dans le fichier OVF++. Pour incorporer une vue à différents niveau nous appliquons le modèle NLN (Node, Link, Network), auquel nous complétons par la déclinaison des contraintes de QoS en appliquant le modèle générique de QoS [9]. La description de réseau virtuel comprend ainsi trois parties: la section de liaison virtuelle, la section de noeud virtuel et la section des contraintes de QoS du réseau virtuel. Plusieurs « section de réseau virtuel » peut être décrites dans une Network Section d’OVF. La figure 6.5 montre l’ensemble des sections proposées. La figure 6.6 et 6.11 introduisent la structure proposée comme extension dans le fichier OVF++.

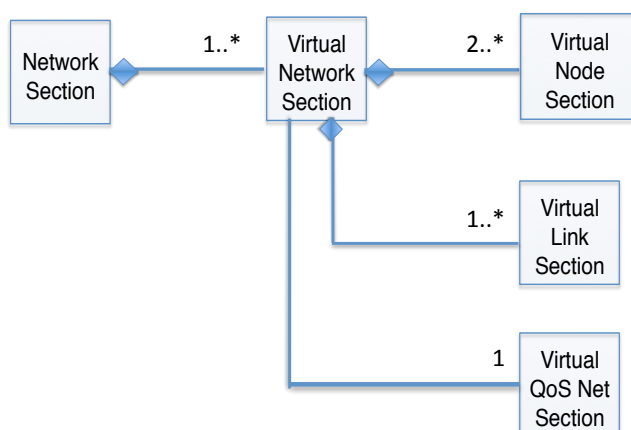


Figure 6.5 – Ensemble des nouvelles sections proposées

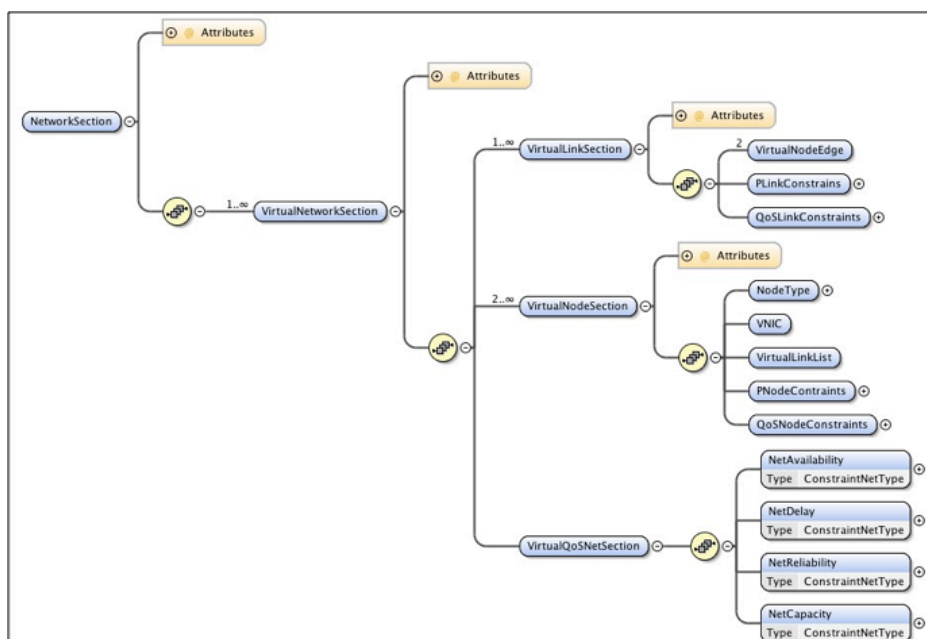


Figure 6.6 – Structure de l’extension OVF pour la description du réseau virtuel

Pour chaque section proposée, nous intégrons la description des contraintes de QoS et de

placement, suivant la définition donnée dans [22] (voir figures 5.3 et 5.4) et nous appliquons ainsi notre approche QoS-aware (voir section 5.2). La description finale du fichier OVF ++ est complétée par le NaaS en fonction du service offert et suivant notre processus de déploiement des réseaux virtuels, comme il a été décrit dans le chapitre 5.

Les figures 6.7 à 6.10 montrent l'ensemble des extensions proposées pour la description des réseaux virtuels.

VirtualNetworkSection	Type of extension	Description of proposal
VirtualNetName	Attribute	Logical name of the virtual network
VirtualNetDescription	Attribute	Textual description about the virtual network
VirtualLinkSection	Section	Describes the QoS constraints of the links in a Virtual Network. These constraints are view through the nodes making part of the link and described at VirtualNodeSection
VirtualNodeSection	Section	Describes the properties and QoS/Placement constraints of the nodes involve in virtual link
VirtualQoSNetSection	Section	Describes the behavior of Virtual Network by using the four QoS criteria

Figure 6.7 – Extension OVF pour décrire le VirtualNetworkSection

VirtualLinkSection	Type of extension	Description
Virtual LinkName	Attribute	Logical name of a virtual link.
VirtualNodeEdge	Section	Reference of two nodes edge of virtual link
PLinkConstraints	Section	Information about the placement constraints to the virtual link (the intermediate virtual nodes to be considered).
	TransitionNodeNumber	Subsection giving information about the number maximum of intermediate nodes to be considered in a virtual link.
	TransitionNodeList	Subsection giving the reference of intermediate nodes.
QoSLinkConstraints	Section	QoS constraints considered at link level.
	LinkAvailability	Indicates that the required availability rate of the virtual link must be at least equal to a given rate.
	MaxLinkDelay	Indicates the maximum delay to be considered in the transmission with this virtual link. This delay involves the sum of transmission delay and the sum of processing time considering in intermediate nodes.
	LinkReliability	Indicates that the required reliability rate of the virtual link must be at least equal to a given rate.
	LinkCapacity	Indicates the required capacity of virtual link must be at least equal to a given rate. This capacity involves both: the capacity of intermediate nodes, is they exist, and the capacity of link between two intermediate nodes.

Figure 6.8 – Extension OVF pour décrire le VirtualNodeSection

VirtualNode Section	Type of extension	Description
VirtualNodeName	Attribute	Logical name of a virtual node.
VirtualNodeFonctions	Attribute	References the VNF(s) to be programmed at the virtual node
NodeType	Section	Indicates the kind of node (Edge or Intermediate)
VNIC	Section	Indicates the reference of Virtual Network Interface associated with the virtual node
VirtualLinkList	Section	References the list of virtual links where the virtual node takes part
PNodeConstraints	Section	Placement constraints considered at node level.
	IsolationNode Constraint	Allows indicate a isolation constraint to be considered in the placement of certain virtual nodes involved in the link
	AfinityNode Constraint	Indicates the constraints of placement of virtual nodes inside one physical node. If affinity is activating, a list of affinity virtual nodes must be indicate.
	Anti-AfinityNode Constraint	Indicates that the virtual node does not must be placed with other virtual nodes. If anti-affinity is activating, a list of virtual anti-affinity nodes must be indicate.
QoSNode Constraints	Section	QoS constraints considered at node level.
	NodeAvailability	Indicates that the required availability rate of the virtual node must be at least equal to a given rate.
	NodeDelay	Indicates that the required time of treatment of the virtual node has to be lower than a given rate.
	NodeReliability	Indicates that the reliability rate of virtual node must be at least equal to a given rate.
	NodeCapacity	Indicates that the capacity type (several measures can be used CPU, storage, I/O interfaces, etc.) must be at least equal to given rate.

Figure 6.9 – Extension OVF pour décrire le VirtualLinkSection

VirtualQoSNetSection	Type of extension	Description
NetAvailability	Section	Indicates a total rate of availability of virtual network. This constraint is representing by the sum of availability rates considering at virtual nodes and links involved in the virtual network.
NetDelay	Section	Take into consideration that the set of times of treatment to give the service of media delivery. This total time is representing by the sum of transmission times in virtual links and the times (treatment, waiting) consumed by each of virtual nodes traversed.
NetReliability	Section	Indicates a total rate of reliability of virtual network. This constraint is representing by the sum of reliability rates considering at virtual nodes and links involved in the virtual network.
NetCapacity	Section	Indicates a total rate of capacity of virtual network to be considered.

Figure 6.10 – OVF extension - VirtualQoSNetSection

```

<NetworkSection>
  <VirtualNetworkSection ovf:VirtualNetName="VirtualNetwork VoIP" ovf:VirtualNetDescription="Virtual Network to deploy the
  service of VoIP">

    <VirtualLinkSection ovf:VirtualLinkName="VL 1" >
      <VirtualNodeEdge> VNode 1 </VirtualNodeEdge>
      <VirtualNodeEdge> VNode 2 </VirtualNodeEdge>
      <PLinkConstraints>
        <TransitionNodeNumber> 2 </TransitionNodeNumber>
        <IntermediateNodeList> Vnode12, VNode13 </IntermediateNodeList>
      </PLinkConstraints>
      <QoSLinkConstraints>
        <LinkAvailability>
          <Rate>80%</Rate>
        </LinkAvailability>
        <LinkAvailability>
          <MaxLinkDelay>
            <Rate>100 ms</Rate>
          </MaxLinkDelay>
        </LinkAvailability>
        <LinkReliability>
          <Rate>2%</Rate>
        </LinkReliability>
        <LinkCapacity>
          <Rate>100Kbps</Rate>
        </LinkCapacity>
      </QoSLinkConstraints>
    </VirtualLinkSection >

    <VirtualNodeSection ovf:VirtualNodeName="VNode1" ovf:VirtualNodeFunctions="VNF1, VFN22">
      <NodeType> Edge </NodeType>
      <VNIC> Veth1 </VNIC>
      <VirtualLinkList> VL1, VL3 </VirtualLinkList>
      <PNodeConstraints>
        <IsolationNodeConstraint> VNode1 </IsolationNodeConstraint>
        <AffinityNodeConstraint> </AffinityNodeConstraint>
        <Anti-AffinityNodeConstraint> VNode2 </Anti-AffinityNodeConstraint>
      </PNodeConstraints>
      <QoSNodeConstraints>
        <NodeAvailability>
          <Parameter> Accessibility time </Parameter>
          <Value> >0.01ms </Value>
        </NodeAvailability>
        <NodeDelay>
          <Parameter> Delay of traitement for each request </Parameter>
          <Value> [0.015ms, 0.03 ms] </Value>
        </NodeDelay>
        <NodeReliability>
          <Parameter> Loss rate </Parameter>
          <Value> [0.8%, 1.5%] </Value>
        </NodeReliability>
        <NodeCapacity>
          <Parameter> Maximun number of requests </Parameter>
          <Value> 100 </Value>
        </NodeCapacity>
      </QoSNodeConstraints>
    </VirtualNodeSection>

    <VirtualQoSNetSection>
      <NetAvailability> Availability rate =95% </NetAvailability>
      <NetDelay> end-to-end delay = [180ms, 300 ms] </NetDelay>
      <NetReliability> Reliability rate = [1.8%, 3.5%]</NetReliability>
      <NetCapacity> 100 simultaneous connections </NetCapacity>
    </VirtualQoSNetSection>

  </VirtualNetworkSection>
</NetworkSection>

```

Figure 6.11 – Description du réseau virtuel : exemple de Fichier en OVF++ - Virtual Network Section



# 7

## Use case : vers une personnalisation autonome continue

Nous allons illustrer dans les paragraphes suivants un use case sur le Cloud Gaming permettant de mettre en place notre démarche pour la définition d'un « service réseau personnalisé et à la demande ». Pour ce faire des questions relatives au comment représenter, stocker, interconnecter et organiser les informations sur les composants de services à différents niveaux sont à étudier. Nous avons préconisé l'utilisation de notre modèle de QoS pour donner une réponse à cette représentation en intégrant une vue fonctionnelle et non fonctionnelle (la QoS) des composants de service. Le même modèle est appliqué au niveau des services applicatifs et au niveau des services réseau.

Notre modèle n'étant pas attaché à un langage particulier, son implémentation pourra être réalisé en utilisant les technologies actuelles de description des services, tels que USDL (*Unified Service Description Language*) [57] ou par exemple OWL-S (*Ontology Web Language*) [58], proposés dans le contexte des web services.

Une fois le déploiement dynamique réalisé, l'allocation des ressources se fait sous contrainte de QoS au moment de la consommation effective du service par l'utilisateur.

Nous montrons dans la figure 7.1 un exemple d'Use Case dans le contexte du Cloud gaming. Au niveau applicatif le service est assuré par le fournisseur de Cloud gaming qui fournit les jeux disponibles en ligne.

Nous considérons dans un premier temps, que le joueur se connecte à Internet à travers un accès sans fils avec son profil de session de jeux 3D, qui est stockée dans le serveur de Cloud Gaming. Le *Cloud Gaming provider* est le responsable qui compose et provisionne les composants de services applicatifs. Le *Cloud NaaS provider* doit lui fournir les services réseau nécessaires pour supporter les flux applicatifs de la session de jeu.



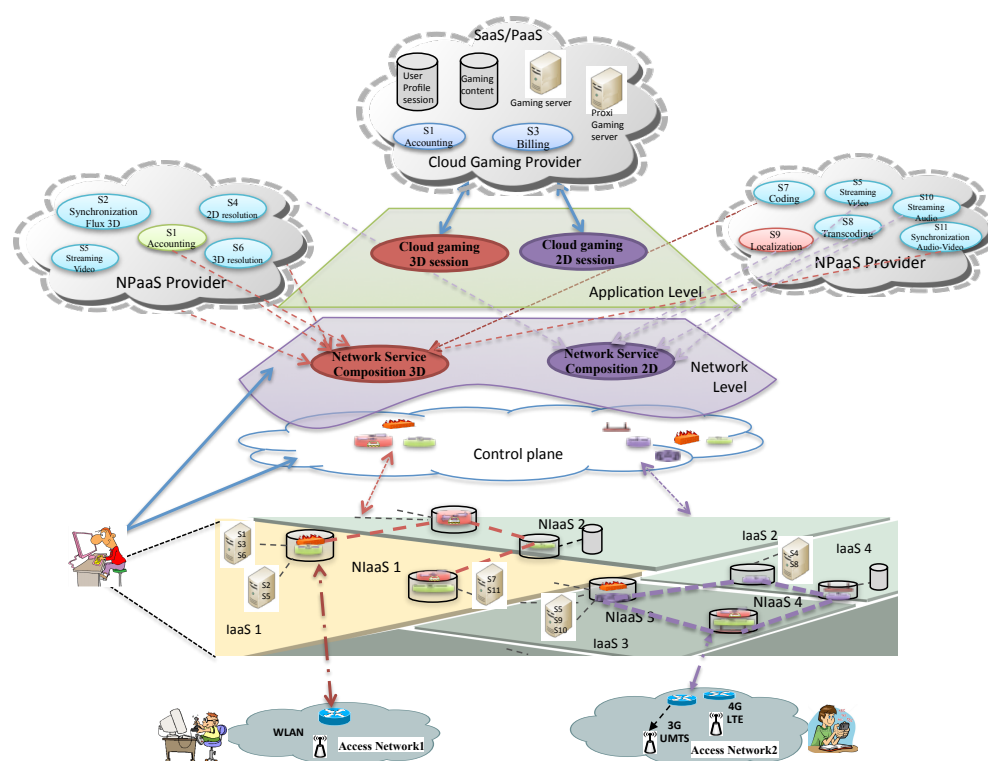


Figure 7.1 – Use case Cloud Gaming - NaaS Cloudifié, Virtualisé et programmable pour un service personnalisé

Un ou plusieurs fournisseurs de Cloud Gaming peuvent intervenir pour fournir par exemple l'accès aux codes spécifiques de jeux proposés. Les jeux proposés à l'utilisateur dépendent de sa configuration et des ressources pour afficher les flux vidéo – audio envoyés. A cet effet le Cloud service provider (le fournisseur de Cloud Gaming, dans notre exemple) analyse les besoins de l'utilisateur en fonction des services et des ressources (le profil d'utilisateur) et compose le VPSN (les services associés au jeu vidéo requis selon la QoS demandée par l'utilisateur). La composition de services inclut au niveau du PaaS les services applicatifs, et au niveau du NaaS les services de réseau qui devront supporter l'acheminement des flux des services applicatifs. Ainsi pour supporter les sessions de jeux, plusieurs services réseau peuvent composer le VPCN, parmi eux :

- Pour la connectivité : DHCP server, NAT router, APL (*Application Level gateway*), etc
- Pour le media streaming : coding, streaming video, streaming audio, synchronisation des flux 3D, synchronisation vidéo-audio, multi-screen, media cache, etc.
- Pour la sécurité : firewall, IPS, Parental control, etc.

Ces services réseau pourraient être fournis par des fournisseurs différents. Au niveau réseau le NaaS provider doit analyser/dimensionner en fonctions des caractéristiques QoS des flux, quels services distribuer pour définir le réseau virtuel qui supportera la session de jeu.

Finalement l'architecte NaaS orchestrera la composition des services et réalise le déploiement nécessaire au niveau NaaS pour supporter la session. Dans cet exemple nous retrouvons le réseau rouge qui supporte la session lorsque l'utilisateur se trouve à son domicile.

Dans un deuxième temps, on considère que l'utilisateur change d'équipement et se déplace. Ces changements se répercutent sur la composition du service de Cloud Gaming. On passe par exemple d'une session 3D vers une session 2D, la session 3D n'est plus supportée par le nouvel équipement. En outre l'utilisateur se déplace, et le réseau d'accès change. Le NaaS doit recomposer le service réseau pour s'adapter aux changements et supporter le nouveau flux de l'application. L'architecte NaaS doit orchestrer la nouvelle composition et déployer les services réseau nécessaires. Dans la figure 7.1, le service réseau mobile adapté est présenté par le réseau violet.

Le niveau «Application Level» de notre exemple fournit la logique de service applicatif final à l'utilisateur, la composition des services applicatifs à travers le VPSN.

Le niveau «Network Level» fournit la composition de services réseaux liés à l'application de Cloud Gaming (le Network Service Delivery). La vue «control plane» à ce niveau fournit à son tour l'assemblage de ressources réseaux (VPCN) nécessaires au support et traitement du trafic du média correspondant aux demandes des services applicatifs choisis (le Media Delivery) et qui seront placés au niveau du plan de données.



# 8

## Expérimentation

Nous présentons dans cette section une implémentation de l'architecture NaaS proposée dans ce document. Pour réaliser cette implémentation des choix sur les outils/solutions open source disponibles ont du être effectués (8.1). Afin d'étudier les faisabilités de déploiement (8.2) nous avons réalisé différentes expérimentations (8.3). Nos premiers résultats sont analysés dans (8.4).

Cette implémentation a été réalisée sur la plate-forme GRID 5000 [59].

### 8.1 Choix des composants

Table 8.1 – Caractéristiques des nœuds GRID 5000

Machine type	x86 64
Operating System	Linux
Operating System version	3.2.0-4-amd64
Number of CPUs	2 CPUs
CPU speed	2390 MHz
Memory Size	2060472 KB

Notre architecture comporte plusieurs niveaux de composants :

- L'infrastructure physique : permettant de réaliser les expérimentations est fournie par Grid5000. Les ressources allouées pour chaque nœud physique sont consignées dans le

tableau 8.1.

- L'infrastructure virtualisée du NaaS : nous considérons dans notre étude que le NaaS fournit ses ressources à travers une infrastructure physique virtualisée : le NIaaS (*Network IaaS*). Pour ce faire nous avons choisi d'utiliser Open vSwitch (Version 2.3.1) [60] pour définir les nœuds virtualisés de l'infrastructure du NaaS. Nous avons choisi Open vSwitch à la place de Mininet [61] (souvent choisi dans ce type de configuration) qui demande l'installation de l'ensemble de la topologie sur une même VM. Open Vswitch, lui, nous permet de déployer séparément chaque nœud virtuel sur les machines physiques différentes. Ainsi nous avons obtenu des informations propres à chaque VM.
  - Des réseaux virtuels basés SDN : l'enjeu majeur est de permettre à plusieurs réseaux virtuels de partager la même infrastructure virtualisée, le NIaaS, chaque réseau virtuel ayant son propre contrôleur. Pour ce faire nous avons utilisé le contrôleur OpenFlow FlowVisor [62] qui agit tel qu'un hyperviseur réseau en permettant la virtualisation des réseaux SDN, à travers la programmation et l'isolation de tranches des réseaux virtuels (slices), chacune déléguée à un contrôleur différent.
  - Les contrôleurs des réseaux virtuels : nous considérons des réseaux virtuels basés sur SDN. Chaque réseau virtuel possède son propre contrôleur. Après une étude préalable [63] sur les contrôleurs OpenFlow [6] nous avons choisi d'utiliser (dans un premier temps) le contrôleur open source Floodlight [64] pour chaque réseau virtuel implémenté.
- L'Interface Sud : nos Open VSwitch (OVS) communiquent avec le contrôleur FlowVisor en utilisant le protocole OpenFlow.
- L'Interface Nord : pour nos premières expérimentations les descriptions concernant la description des flux, la topologie et les contraintes réseaux à respecter sont programmées manuellement. Pour introduire de la dynamique dans l'application de cette description et sa prise en compte pour le déploiement, nous travaillons actuellement sur la description de ces aspects dans une description plus détaillées sous le format *Open Virtualization Format ++* (OVF++).

La figure 8.1 montre le positionnement des toutes ces niveaux considérés dans notre expérimentation.

## 8.2 But de l'expérimentation

En partant de notre démarche pour la construction des réseaux virtuels VN+VNOS, cette première phase d'expérimentation vise à évaluer la performance des outils open source disponibles, et à réaliser les connexions nécessaires entre les différentes entités invoquées, dans les deux niveaux : le niveau du plan de contrôle et le niveau du plan de données.

Les premiers tests ont été réalisés pour définir les entités qui sont responsables de la prise de décisions et pour comprendre comment fonctionnent-ils lors des décisions sur l'utilisation des ressources, de réaffectation et sur la répartition en cas de changements des contraintes du

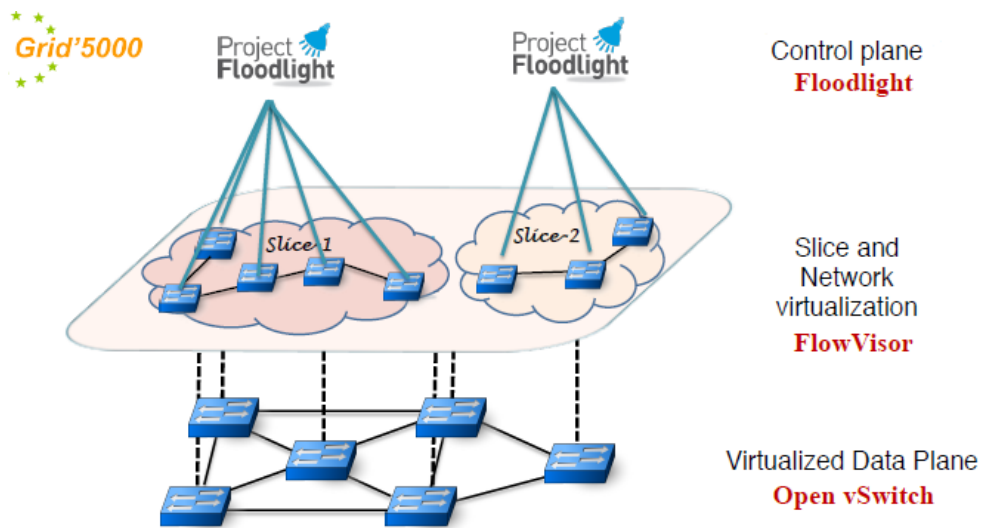


Figure 8.1 – Environnement d’expérimentation sur GRID 5000.

réseau (par exemple le nombre de nœuds à traverser). Ainsi, nous avons étudié les impacts des changements au niveau des nœuds du réseau, en commençant par le nombre de nœuds traversés et leur capacité.

Ces outils permettent aujourd’hui de réaliser des configurations manuellement. Pour introduire de la dynamique, des modules d’automatisme devraient être intégrés dans les différentes configurations.

### 8.3 Expérimentations réalisées

Nous avons effectué les différentes phases préconisées et configuré manuellement nos expériences. Une première étude a été effectuée pour établir la logique de service, puis la phase de déploiement tenant compte du type d’application et donc du type de flux qui va être généré (Network Service Delivery) puis une phase de répartition sur l’abstraction, sur l’image de l’infrastructure virtualisée, pour ensuite faire le placement sur les ressources disponibles (fournies par Grid5000) afin d’assurer le media delivery. Une fois tous les composants activés, nous avons mesuré certains paramètres de QoS pour superviser le respect des contraintes applicatifs au niveau du transport du réseau pour le flux de bout-en-bout. Les premiers tests concernent les paramètres de délai (temps de réponse, gigue).

#### 8.3.1 Use case 1 : Mesure du temps de réponse

L’objectif de cette expérience est de mesurer le temps de réponse ou le RTT (*Round Trip Time*) pour un flux d’application lors du franchissement du réseau. Du point de vue de l’application, la qualité de service et la performance dépendront de la mesure RTT. RTT est étroitement liée au nombre de nœuds à traverser dans le réseau. Pour ce use case, nous considérons une topologie de réseau linéaire pour effectuer 3 expériences où nous faisons varier le nombre de

nœuds croisés: 3, 7 puis 12 nœuds pour étudier l'impact. Nous générons du trafic en utilisant l'outil de commande Ping pour chacun des trois tests où nous réalisons 20 simulations. Chaque simulation dure 240 secondes. Le Tableau 8.2 montre les résultats.

Table 8.2 – Expériences RTT – Use Case 1

Experiment	RTT (msec)
1	0,276700
2	0,722950
3	1,222000

Dans ce scénario, le temps nécessaire à un paquet pour traverser un nœud de Open vSwitch et une liaison réseau est de 0,05 ms. Nous avons noté une augmentation du temps RTT chaque fois que nous augmentons le nombre de nœuds. Le temps nécessaire pour l'application des flux de passer par les liens du réseau et des éléments de transmission (*Open vSwitch*) est plus important quand le nombre de nœuds augmente. Si une contrainte d'application est de respecter une certaine valeur RTT, le fournisseur de réseau par l'intermédiaire du contrôleur de réseau devrait programmer les nœuds du réseau virtuel pour cette application en conséquence. Par exemple, l'application VoIP utilisant le protocole RTP est une application sensible au temps qui ne tolère pas un RTT supérieur à 150 ms. Cela devrait caractériser la QoS attendue du réseau aux demandes exprimés dans le SLA de l'application pour la prestation de services aux niveaux du Media Service Delivery et du Network Service Delivery.

### 8.3.2 Use case 2 : Mesure de la gigue

Pour ce deuxième use case, nous considérons deux applications différentes générant deux types de flux différents. Un trafic vidéo générant des flux UDP (*User Datagram Protocol*) et une application VoIP générant des flux RTP (*Real Time Protocol*).

Table 8.3 – Configuration du Trafic - Use Case 2

Configurations	UDP Traffic	RTP Traffic
Traffic generator	Iperf	Mausezahn
Number of simulations	10	10
Duration for each simulation	120 seconds	120 seconds
Number of paquets	102042	112000
Paquet size	1470 bytes	172 bytes
Link bandwidth	10 Mbps	50 Mbps
Throughput	10 Mbps	
Codec		G711
Transmission Delay		1 msec

Nous avons configuré deux réseaux virtuels chacun traitant le media delivery des flux (UDP, RTP) pour ces deux applications. Nous avons utilisé deux générateurs de flux [65] : Iperf [66] pour la génération des flux UDP et Mausezahn [67] pour la génération des flux RTP. Le tableau 8.3 résume les configurations des expériences et les caractéristiques de chaque flux et donc de chaque réseau virtuel.

Nous avons fait varier la topologie de chaque réseau virtuel, avec deux configurations ( (a) et (b) ), et nous avons fait varier le nombre de nœuds traversés pour chaque configuration, comme le montre la figure 8.2.

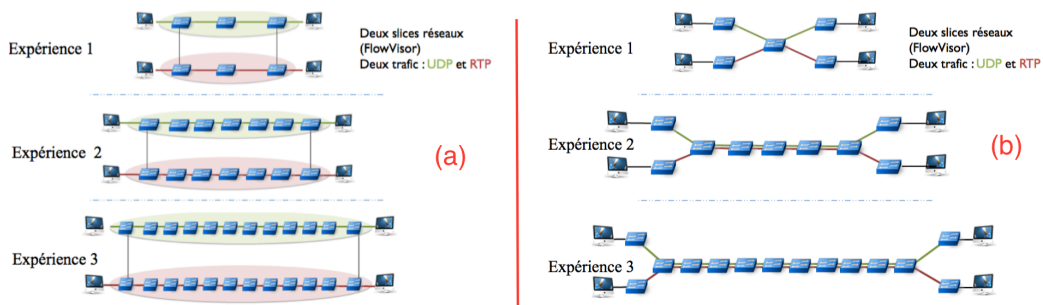


Figure 8.2 – (a) Scénario 1 : configuration sans nœuds partagés  
(b) Scénario 2 : configuration avec des nœuds partagés

Les premiers résultats concernant les deux scénarios de topologie sont présentés dans les tableaux : (a) pour le traitement des flux RTP (tableau 8.4) et (b) pour le traitement des flux UDP (tableau 8.5). On peut noter que la gigue se voit affectée par la modification de la topologie du réseau et par le nombre de nœuds traversés. Mais ces changements concernant la gigue ne sont pas significatifs et, comme attendus, ils sont plus importants dans le cas du partage d'un nœud commun qui est utilisé par deux slices réseaux. Les flux RTP pour l'application VoIP rencontrent de plus grandes variations que l'application vidéo générant des flux UDP, dont la gigue moyenne reste stable. Nous procédons actuellement à de nouveaux tests pour corroborer ces premiers résultats.

Table 8.4 – (a) Flux RTP

RTP Flow	Scenario 1	Scenario 2
Experiment	Jitter ( $\mu$ )	Jitter ( $\mu$ )
1	1.907	3.7322
2	3.2799	34.0190
3	8.1415	75.2890

Table 8.5 – (b) Flux UDP

UDP Flow	Scenario 1	Scenario 2
Experiment	Jitter ( $\mu$ )	Jitter ( $\mu$ )
1	80.4	79.95
2	80.9	80.59
3	81.9	81.17

## 8.4 Analyse des résultats

Nous avons pu constater que les changements intervenant dans le niveau Media Service Delivery, par exemple le nombre de nœuds traversés en changeant la topologie du réseau virtuel, se



répercutent au niveau du Network Service Delivery (par exemple le temps de réponse mesuré ici à travers le RTT).

L'étude réalisée montre que les outils d'aujourd'hui agissent au niveau du Media delivery, dont les opérations consistent très souvent au calcul de meilleurs chemins. Pour que les choix faits en amont impactent les performances, le service de Network Delivery qui se veut «User Centric» voir «Content Centric» doit orchestrer une vue globale et dynamique du service à rendre en s'appuyant sur l'hyperviseur réseau. C'est l'hyperviseur réseau qui devrait maintenir dynamiquement le service du Media Delivery en supervisant la topologie du réseau virtuel, la capacité des liens, les délais acceptables pour les liens, etc.

Cette première expérimentation nous a permis également de mettre en place une phase de calibrage pour les prises de décisions.

# 9

## Conclusion

Pour presque tous les acteurs, la transformation numérique est un précurseur nécessaire à l'Internet hyper-connecté envisagé à l'avenir. Cela nécessite de la virtualisation, de la programmabilité et du «as a Service» des IT et des fonctions de réseau.

L'écosystème réseau est en pleine mutation et les opérateurs doivent être en capacité d'offrir de la flexibilité, adaptabilité et dynamique. En effet, à l'instar de l'IT, nous assistons à la cloudification du réseau (NFV), et à sa softwarisation (SDN) afin d'améliorer les gains financiers et l'efficacité opérationnelle.

Dans ce contexte évolutif dont les usages et les technologies sont complémentaires les questions auxquelles nous avons essayé d'apporter de réponses concernent la définition et fourniture du service réseau : Comment offrir un service réseau à la demande en accord avec les besoins des applicatifs ? Quelles fonctionnalités devraient permettre et favoriser la personnalisation des services réseau? Comment incorporer les bénéfices et avantages offerts par la virtualisation, la cloudification et le SOA à la définition d'un tel service réseau?

Ce document présente nos réflexions et contributions pour définir une personnalisation du service réseau à la demande.

Nous avons commencé par examiner les changements apportés par la virtualisation et la programmabilité au niveau réseau. En partant de l'évolution menée par l'application des ses différents paradigmes au niveau du réseau, nous sommes arrivés au nouveau modèle NaaS où l'on distingue la «virtualisation du service réseau» et la «virtualisation des contrôleurs des flux réseaux (VNOS)» de la «virtualisation des équipement réseau».

Notre proposition préconise ainsi un changement au niveau du cycle de vie des services réseau du NaaS. La phase de déploiement de réseau se rapporte aux réseaux virtuels associés à leur plan de contrôle et la phase de placement sur l'infrastructure physique relève d'une problématique de placement des VMs. La nouvelle phase de déploiement de réseau virtuel

ainsi proposé introduit deux nouveaux processus, celui de la construction des VN+VNOS et celui de la distribution des composants du NaaS pour permettre la construction et la programmation des réseaux virtuels, avant leur placement.

Notre proposition introduit ainsi un mécanisme de personnalisation et à la demande pour les services réseau.

*Des services réseau personnalisables* car nous intégrons dans notre démarche les approches SDN et NFV permettant de mettre en place en place une programmabilité de composants au niveau du réseau virtuel (fonctions réseau virtualisées) et de son plan de contrôle associé.

*Des services réseau à la demande* car à travers la virtualisation le processus de déploiement devient continu et dynamique, pouvant prendre en charge les changements au niveau du service delivery et répercutant ces changements au niveau de la définition du réseau virtuel qui supportera le media delivery.

L'automatisation est l'objectif final permettant de maintenir un processus de déploiement dynamique et continue des ressources réseaux. Cette automatisation permettant aussi une convergence entre le contrôle et la gestion. Pour permettre l'automatisation des processus de déploiement virtuel nous avons proposé de décrire le réseau virtuel à déployer en utilisant une description basée sur le langage OVF ++.

Nous incluons dans notre description les contraintes (QoS et placement) que le NaaS doit prendre en compte pour chaque niveau (réseau, lien et noeud). Notre description montre deux points de vue: la qualité de service demandée et la qualité de service offerte. Cette description proposée en OVF++ a été intégrée dans le cadre du projet OpenCloudware.

Une première mise en oeuvre de l'architecture NaaS nous a permis de comprendre le fonctionnement des différents outils existants. Nous avons constaté que l'interface sud répond aux promesses faites par les éditeurs, mais pour avoir plus d'agilité il nous faut encore développer un certains nombres de fonctionnalités et d'automatismes. Ainsi on pourrait avoir l'interface nord à la hauteur d'un service réseau delivery satisfaisant pour les différents demandeurs.

La conception du modèle d'information viendra compléter la vue globale nécessaire pour l'intégration des processus de déploiement et de gestion. Le modèle d'information permettra une automatisation du monitoring et un déploiement continue concernant les services réseaux nécessaires pour supporter le service de media delivery.

# 10

## Perspectives de recherche

Mon projet de recherche constitue une suite logique des activités menées ces 8 dernières années, en particulier sur le déploiement dynamique dans le contexte des services et réseaux virtualisés.

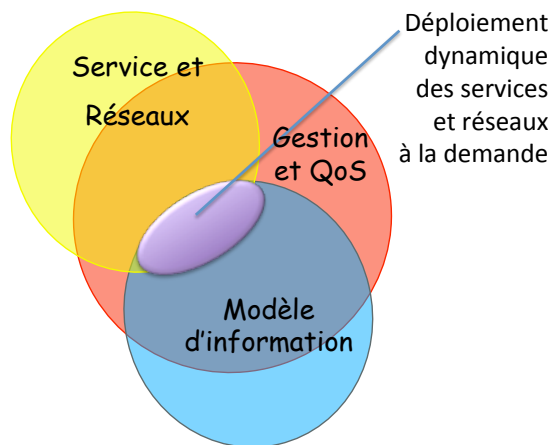


Figure 10.1 – Contexte de la recherche  
Les perspectives

La thématique proposée par mon projet de recherche se situe à l'intersection des trois domaines (Services et Réseaux, gestion de la QoS et Modèle d'information), pour un déploiement dynamique efficace et à la demande.

En effet, le déploiement dynamique doit être guidé par un modèle d'information permettant de réaliser une traduction de la composition de services vers une réalité de déploiement. Pour les architectures virtuelles le déploiement implique l'instanciation des composants de services nécessaires sur une infrastructure virtualisée, qui s'appuie à son tour sur les ressources physiques les mieux indiquées, choisies selon un plan

d'optimisation. Le déploiement devient ainsi intelligent grâce aux connaissances représentées et permettant de gérer des actions nécessaires au niveau du plan de contrôle et de gestion.

Nous avons décrit dans ce document nos propositions concernant le déploiement des réseaux virtuels personnalisables et à la demande. Ces propositions sont basées sur la définition du réseau comme un service et sur l'introduction d'une nouvelle phase dans le cycle de

vie des services réseaux en milieu virtualisé.

Nous avons présenté notre point de vue architecturale avec la définition de l'Architecture NaaS, fonctionnelle à travers la description des composants SCC et de la session réseau et informationnelle à travers la description des réseaux virtuels à déployer. Ce dernier point doit être enrichi avec la définition d'un modèle d'information plus complet et applicable aux différentes phases du cycle de vie proposé.

Nous devons compléter notre vision en intégrant les points de vue organisationnelle et en complétant l'aspect informationnelle. Ces deux derniers aspects permettront de mettre en place une convergence entre contrôle et gestion.

Les travaux envisagés concernent trois axes de recherches avec des objectifs complémentaires :

Axe 1: Gestion et orchestration pour un déploiement on demande

Axe 2: Application de l'approche Content Centric

Axe 3: Domaine Véhiculaire

## 10.1 Axe 1 : Gestion et orchestration pour un déploiement on demande

Dans le cadre des réseaux et services du futur, les nouvelles organisations devront prendre en considération tout un ensemble de critères (mobilités, personnalisation de services, composition dynamique, etc.) qui permettront de répondre aux besoins des utilisateurs et fournisseurs de services, en déployant des solutions flexibles et adaptables.

Les services changent en fonction du contexte applicatif et de leur besoin ; l'organisation doit permettre de déployer les services selon des stratégies différentes. Ceci revient à définir différentes stratégies de déploiement des services avec l'intégration des usages, d'une façon intelligente et dynamique et à des fins d'efficacité pour tous les acteurs. Ainsi, pour mettre en place des conceptions et déploiements dynamiques et permettre une vraie personnalisation des réseaux il faut aussi considérer les aspects organisationnels.

L'aspect organisationnel fait référence à la distribution des services dans des composants de l'architecture afin de construire un service final personnalisé. Des mécanismes d'orchestration doivent être intégrés dans les nouvelles architectures de services guidant une intégration autant verticale que horizontale des solutions.

Dans ce contexte nous devons travailler sur la spécification des mécanismes nécessaires à mettre en place au niveau de l'orchestrateur et de l'hyperviseur réseau proposés par l'architecture NaaS, et faire ainsi converger les processus de contrôle et de gestion en utilisant un modèle d'information.

Le modèle d'information devrait permettre de représenter les informations pertinentes par phase du cycle de vie du service réseau, permettant de définir, déployer et garantir un contrôle du service réseau delivery. Basés sur les modèles de QoS et de profils [9] nous intégrerons les informations nécessaires pour garantir un contrôle et une gestion continue des services réseau virtualisés (Figure 10.2).

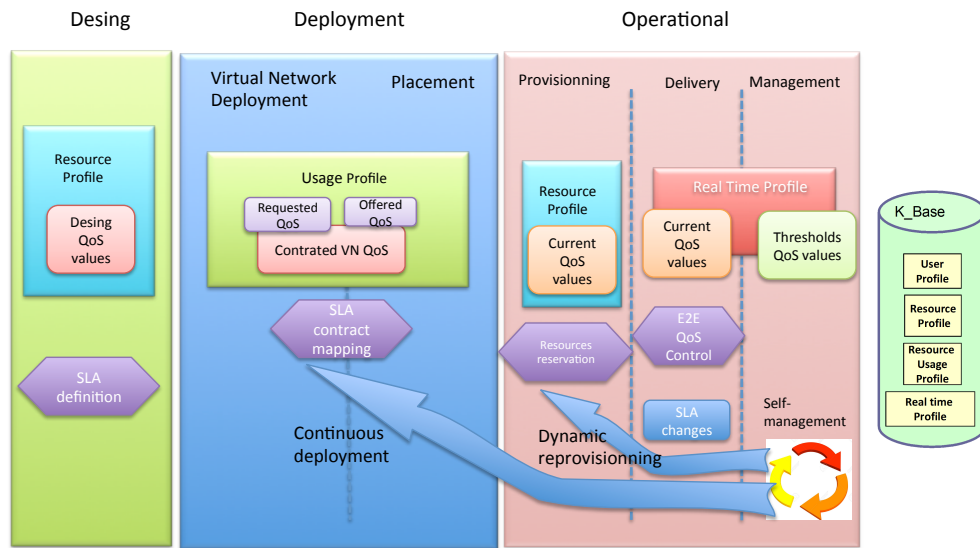


Figure 10.2 – Les modèles de QoS et de profils appliqués au contexte du déploiement des réseaux virtuels

## 10.2 Axe 2 : Content Centric

Les limitations importantes inhérentes aux architectures actuelles en silo et le manque d’agilité dans nos plates-formes et services d’infrastructure nous imposent de penser autrement les services pour les rendre disponibles. Un processus de déploiement à la demande a besoin d’une base de connaissances (Système d’Information) pour automatiser le processus de gestion de l’instanciation d’une organisation continue (architecture physique) convenant aux demandes des utilisateurs et aux stratégies des fournisseurs.

L’aspect informationnel est très important, la convergence des nouvelles architectures de services passe par une structuration des informations (Big Data, IoT) et des modèles d’information pertinente.

Le *Content Centric* (CC) constitue l’approche complémentaire à l’*User Centric* en fournissant des outils et des stratégies de placement de services (et de distribution de serveurs) de façon à fluidifier et à accélérer l’accès et la consultation de contenus. Les avantages et principes de cette approche sont déjà appliqués dans le cadre du Cloud (CCDN) permettant de fournir une solution viable et rentable pour la réalisation de contenus réseaux et services de livraison.

Dans ce contexte nous envisageons l’intégration des approches CC comme mécanisme permettant de faciliter le choix de composants en fonction des contenus des flux qu’ils peuvent traiter. Ceci permettra d’utiliser des stratégies pour organiser et déployer les services en optimisant le transfert des flux basé sur le contenu. L’approche CC est ainsi vu comme un outil permettant de mettre en place des stratégies de déploiement différentes en fonction des informations pertinentes (contrôle, gestion, usage, etc.) représentées dans le modèle d’information à proposer.

### 10.3 Axe 3 : Domaine Véhiculaire

Ce dernier axe constitue un champ d'application. Nous allons examiner comment les différentes propositions de l'axe 1 et 2 peuvent être intégrées dans le cadre de réseaux véhiculaires.

Cet axe correspond à la suite de mes travaux menés depuis 2008 sur ce domaine. Une première architecture fonctionnelle a été développée dans le cadre ITS, l'idée est d'intégrer donc la partie organisationnelle pour répondre à des déploiements dynamiques et à la demande des différents services (trafic, parking, jeux, etc.). Deux questions importantes seront traitées dans ce contexte : le déploiement des agents de l'architecture, mais aussi la spécification de la connaissance qu'ils manipulent.

# Liste des publications

## *Journaux Internationaux et chapitres de livre*

- Gladys DIAZ. "Information Model for Dynamic Deployment of Services". The International Journal of Innovative Research in Computer and Communication Engineering. Volume 3, Issue 6, June 2015. Doi 10.15680/ijircce.2015.0306190.
- Arturo GOMEZ, Gladys DIAZ and Khaled BOUSSETTA. "Adaptive Contention Window for Zone-Based Dissemination of Vehicular Traffic". V. Guyot (Ed.): IEEE ICAIT 2012, LNCS 7593, Springer. pp. 330-339, 2013.
- Abdelhak FARSI, Nadjib ACHIR, Khaled BOUSSETTA, Gladys DIAZ, Arturo GOMEZ. "Backhaul Topology Design and Weighted Max-Min Fair Capacity Allocation in Wireless Mesh Networks". IEEE ICAIT 2012. LNCS 7593, Springer. pp. 296-309, 2013.

## *Conférences Internationales avec actes et comité de sélection*

- Arturo GOMEZ, Gladys DIAZ and Khaled BOUSSETTA, "How Virtual Police Agents can Help in the Traffic Guidance?" IEEE/WCNC, WVCN Workshop wireless vehicular communication and networks 2012, April 1-4, 2012- Paris France.
- Arturo GOMEZ, Gladys DIAZ and Khaled BOUSSETTA. "Toward the Integration of IVC-based Services into ITS Architecture Standards" IEEE GIIS 2012 – The 4th Global Information, Infrastructure and Networking Symposium 2012, 17-19 December 2012, Choroni – Venezuela.
- Inès AYADI, Noémie SIMONI, Gladys DIAZ. "QoS-Aware Component for Cloud Computing". ICAS 2013, The Ninth International Conference on Autonomic and Autonomous Systems. March 24 - 29, 2013 - Lisbon, Portugal.
- Inès AYADI, Noémie SIMONI, Gladys DIAZ. "NaaS: QoS-aware Cloud Networking services". NCA 2013, The 12th IEEE International Symposium on Network Computing and Applications. 22-24 August, 2013. Cambridge, MA, USA.
- Inès AYADI, Gladys DIAZ, Noémie SIMONI. "QoS-based Network Virtualization to Future Networks: An approach based on QoS network constraints". NoF'13, The 4th International Conference Network of Future. 23-25 October, 2013. Pohang, South Korea.



- Silvana Juárez CHALINI, Miguel Angel LEON CHAVEZ, Gladys DIAZ. "UML Model of a Gateway for the Interconnection of IEEE 1609 and Controller Area Network". 14th international CAN Conference, 12-13 november 2013, Paris, France.
- Arslane HAMZA CHERIF, Fedoua DIDI, Khaled BOUSSETTA and Gladys DIAZ. "A State of Art of Modern Intelligent Transport Systems". The 2015 International Conference on Advanced Communication Systems and Signal Processing. ICOSIP 2015. 08-09 November, 2015. Tlemcen, Algeria.
- Arslane HAMZA CHERIF, Khaled BOUSSETTA, Gladys DIAZ, Fedoue DIDI. "A probabilistic convergecast protocol for vehicle to infrastructure communication in ITS architecture". 13th Annual IEEE Consumer Communications and Networking Conference (CCNC 2016). Las Vegas, USA, 9 -12 janvier 2016.

# Bibliographie

- [1] "SDN: SOFTWARE-DEFINED NETWORKING: the new norm for networks. white paper." [Online]. Available: <https://www.opennetworking.org>
- [2] "NFV: NETWORK FUNCTION VIRTUALIZATION. white paper." [Online]. Available: [http://portal.etsi.org/NFV/NFV\\_White\\_Paper.pdf](http://portal.etsi.org/NFV/NFV_White_Paper.pdf)
- [3] N. M. K. Chowdhury and R. Boutaba, "A survey of network virtualization," *Computer Networks: The International Journal of Computer and Telecommunications Networking*, vol. 54, no. 5, pp. 862–876, Apr. 2010.
- [4] I. Ayadi, N. Simoni, and G. Diaz, "Naas: Qos-aware cloud networking services," in *Network Computing and Applications (NCA), 2013 12th IEEE International Symposium on*. IEEE, 2013, pp. 97–100.
- [5] I. AYADI, "La virtualisation de bout-en-bout pour la gestion des services cloud sous contraintes de qos," Ph.D. dissertation, Télécom ParisTech, 2014.
- [6] "Openflow." [Online]. Available: <https://www.opennetworking.org/sdn-resources/openflow>
- [7] N. Simoni, S. Znaty, N. Perdignes, and S. Asenis, *Gestion de Réseau et de service: Similitude des concepts, spécificité des solutions*. InterEditions, 1997.
- [8] N. Simoni, *Des réseaux intelligents à la nouvelle génération de services*. Hermès Science publications-Lavoisier, 2007.
- [9] "ETSI TR 102 805-3 V1.1.1 (2010-04). User Group. End-to-end QoS management at the Network Interfaces; Part 3: QoS informational structure." [Online]. Available: [http://www.etsi.org/deliver/etsi\\_tr/102800\\_102899/10280503/01.01.01\\_60/tr\\_10280503v010101p.pdf](http://www.etsi.org/deliver/etsi_tr/102800_102899/10280503/01.01.01_60/tr_10280503v010101p.pdf)
- [10] "ETSI TR 102 805-2 v1.1.1 (2009-11). user group. end-to-end qos management at the network interfaces. part 2: Control and management planes solution - qos continuity." [Online]. Available: [http://www.etsi.org/deliver/etsi\\_tr/102800\\_102899/10280502/01.01.01\\_60/tr\\_10280502v010101p.pdf](http://www.etsi.org/deliver/etsi_tr/102800_102899/10280502/01.01.01_60/tr_10280502v010101p.pdf)
- [11] J. Touch and S. Hotz, "The X-Bone," in *Proceedings of the Third Global Internet Mini-Conference at Globecom, Sydney, Australia, November 1998*, pp. 75–83.
- [12] P. Ruth, X. Jiang, D. Xu, and S. Goasguen, "Virtual Distributed Environments in a Shared Infrastructure," *IEEE Computer*, vol. 38, no. 5, pp. 63–69, 2005.

- [13] "Planetlab." [Online]. Available: <http://www.planet-lab.org/>
- [14] "GENI: Global Environment for Network Innovations." [Online]. Available: <http://www.geni.net/>
- [15] "4WARD project." [Online]. Available: <http://www.4ward-project.eu/index.php?s=overview>
- [16] N. Chowdhury, M. R. Rahman, and R. Boutaba, "Virtual Network Embedding with Coordinated Node and Link Mapping," in *INFOCOM 2009, IEEE*. IEEE, 2009, pp. 783–791.
- [17] H. Show, "GENI design principles," *Computer*, vol. 39, no. 9, pp. 102–105, 2006.
- [18] S. Da Silva, Y. Yemini, and D. Florissi, "The NetScript Active Network System," *Selected Areas in Communications, IEEE Journal on*, vol. 19, no. 3, pp. 538–551, 2001.
- [19] M. E. Kounavis, A. T. Campbell, S. Chou, F. Modoux, J. Vicente, and H. Zhuang, "The Genesis Kernel: A programming system for spawning network architectures," *Selected Areas in Communications, IEEE Journal on*, vol. 19, no. 3, pp. 511–526, 2001.
- [20] "AGAVE: A LIGHTWEIGHT APPROACH FOR VIABLE END-TO-END IP-BASED QOS SERVICES." [Online]. Available: <http://www.ist-agave.org>
- [21] Q. Duan, Y. Yan, and A. V. Vasilakos, "A survey on Service-Oriented Network Virtualization Toward Convergence of Networking and Cloud Computing," *Network and Service Management, IEEE Transactions on*, vol. 9, no. 4, pp. 373–392, 2012.
- [22] I. Ayadi, G. Diaz, and N. Simoni, "QoS-based Network Virtualization to Future Networks: An approach based on network constraints," in *Network of the Future (NOF), 2013 Fourth International Conference on the*. IEEE, 2013, pp. 1–5.
- [23] "Amazon Elastic Compute Cloud (amazon EC2)." [Online]. Available: <http://aws.amazon.com/ec2>
- [24] "Windows azure: Microsoft's Cloud Platform." [Online]. Available: <http://www.windowsazure.com>
- [25] M. F. Bari, R. Boutaba, R. Esteves, L. Z. Granville, M. Podlesny, M. G. Rabbani, Q. Zhang, and M. F. Zhani, "Data Center Network Virtualization: A Survey," *IEEE Communications Surveys Tutorials*, vol. 15, no. 2, pp. 909–928, Second 2013.
- [26] J. Galán-Jiménez and A. Gazo-Cervero, "Overview and Challenges of Overlay Networks: A Survey," *Int J Comput Sci Eng Surv (IJCSSES)*, vol. 2, pp. 19–37, 2011.
- [27] Z. B. Daho and S. Noemie, "Towards Dynamic Virtual Private Service Networks: Design and Self-Management," in *Network Operations and Management Symposium, 2006. NOMS 2006. 10th IEEE/IFIP*. IEEE, 2006, pp. 1–4.
- [28] I. Ayadi, N. Simoni, and G. Diaz, "Qos-aware component for cloud computing," in *ICAS 2013, The Ninth International Conference on Autonomic and Autonomous Systems*, 2013, pp. 24–29.
- [29] "ETSI NFV." [Online]. Available: [http://www.etsi.org/deliver/etsi\\_gs/NFV/001\\_099/002/01.02.01\\_60/gs\\_NFV002v010201p.pdf](http://www.etsi.org/deliver/etsi_gs/NFV/001_099/002/01.02.01_60/gs_NFV002v010201p.pdf)

- [30] J. Batalle, J. Ferrer Riera, E. Escalona, and J. A. Garcia-Espin, "On the implementation of nfv over an openflow infrastructure: routing function virtualization," in *Future Networks and Services (SDN4FNS), 2013 IEEE SDN for.* IEEE, 2013, pp. 1–6.
- [31] G. Monteleone and P. Paglierani, "Session border controller virtualization towards" service-defined" networks based on nfv and sdn," in *Future Networks and Services (SDN4FNS), 2013 IEEE SDN for.* IEEE, 2013, pp. 1–7.
- [32] B. A. Nunes, M. Mendonca, X.-N. Nguyen, K. Obraczka, and T. Turtletti, "A survey of software-defined networking: Past, present, and future of programmable networks," *Communications Surveys & Tutorials, IEEE*, vol. 16, no. 3, pp. 1617–1634, 2014.
- [33] "SP3 : Composants des Services. livrable 3.1 - projet ubis."
- [34] "OpenCloudware Project." [Online]. Available: <http://www.opencloudware.org/bin/view/Main/>
- [35] T. Aubonnet, L. Henrio, S. Kessal, O. Kulankhina, F. Lemoine, E. Madelaine, C. Ruz, and N. Simoni, "Management of service composition based on self-controlled components," *Journal of Internet Services and Applications*, vol. 6, no. 1, pp. 1–17, 2015.
- [36] "Cours ATOMS (N. Simoni)."
- [37] F. Baude, D. Caromel, C. Dalmaso, M. Danelutto, V. Getov, L. Henrio, and C. Pérez, "GCM: a grid extension to fractal for autonomous distributed components," *annals of telecommunications-Annales des télécommunications*, vol. 64, no. 1-2, pp. 5–24, 2009.
- [38] M. Feridan, M. Moser, and A. Tanner, "Building an abstraction layer for management systems integration," in *Proceedings of the 1st IEEE/IFIP International Workshop on End-to-End Virtualization and Grid Management (EVGM2007)*. Citeseer, 2007, pp. 57–60.
- [39] S. K. Ouanouche, "Gestion dynamique de service de bout en bout dans un contexte de mobilité et d'ubiquité: du déploiement au delivery des services," Ph.D. dissertation, Télécom ParisTech, 2011.
- [40] Y. WU, "User-Centric session et QoS Dynamique pour une approche intégrée du NGN," Ph.D. dissertation, Télécom ParisTech, 2011.
- [41] N. Ornelas, N. Simoni, K. Chen, and A. Boutignon, "VPIN: User-session knowledge base for self-management of ambient networks," in *Mobile Ubiquitous Computing, Systems, Services and Technologies, 2008. UBIKOM'08. The Second International Conference on.* IEEE, 2008, pp. 122–127.
- [42] R. Nassar, "Une architecture convergente pour une continuité et personnalisation de services: aspects architectural et fonctionnel," Ph.D. dissertation, Télécom ParisTech, 2012.
- [43] G. A. Lewis, "Role of standards in cloud-computing interoperability," in *System Sciences (HICSS), 2013 46th Hawaii International Conference on.* IEEE, 2013, pp. 1652–1661.
- [44] M. Waschke, "Cloud-specific standards," in *Cloud Standards*. Springer, 2012, pp. 289–331.
- [45] "IETF ACT model." [Online]. Available: <https://datatracker.ietf.org/doc/draft-leebelotti-teas-actn-info/>

- [46] "SDN architecture." [Online]. Available: [https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/onf2015.310\\_Architectural\\_comparison.08-2.pdf](https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/onf2015.310_Architectural_comparison.08-2.pdf)
- [47] "ETSI NFV management and orchestration." [Online]. Available: [https://www.etsi.org/deliver/etsi\\_gs/NFV-MAN/001\\_099/001/01.01.01\\_60/gs\\_NFV-MAN001v010101p.pdf](https://www.etsi.org/deliver/etsi_gs/NFV-MAN/001_099/001/01.01.01_60/gs_NFV-MAN001v010101p.pdf)
- [48] "OVF." [Online]. Available: [http://dmf.org/sites/default/files/standards/documents/DSP0243\\_2.1.0.pdf](http://dmf.org/sites/default/files/standards/documents/DSP0243_2.1.0.pdf)
- [49] F. Galán, A. Sampaio, L. Rodero-Merino, I. Loy, V. Gil, and L. M. Vaquero, "Service specification in cloud environments based on extensions to open standards," in *Proceedings of the fourth international ICST conference on communication system software and middleware*. ACM, 2009, p. 19.
- [50] "OVF SPECIFICATION." [Online]. Available: [http://www.dmf.org/sites/default/files/standards/documents/DSP0243\\_2.1.0.pdf](http://www.dmf.org/sites/default/files/standards/documents/DSP0243_2.1.0.pdf)
- [51] "CIM model." [Online]. Available: <http://www.dmf.org/standards/cim>
- [52] L. Rodero-Merino, L. M. Vaquero, V. Gil, F. Galán, J. Fontán, R. S. Montero, and I. M. Llorente, "From infrastructure delivery to service management in clouds," *Future Generation Computer Systems*, vol. 26, no. 8, pp. 1226–1240, 2010.
- [53] C. Chapman, W. Emmerich, F. G. Márquez, S. Clayman, and A. Galis, "Software architecture definition for on-demand cloud provisioning," *Cluster Computing*, vol. 15, no. 2, pp. 79–100, 2012.
- [54] C. Chapman, W. Emmerich, F. G. Marquez, S. Clayman, and A. Galis, "Elastic service management in computational clouds," in *12th IEEE/IFIP NOMS2010/International Workshop on Cloud Management (CloudMan 2010)*, 2010, pp. 19–23.
- [55] S. García-Gómez, M. Jiménez-Gañán, Y. Taher, C. Momm, F. Junker, J. Bíró, A. Menychtas, V. Andrikopoulos, and S. Strauch, "Challenges for the comprehensive management of cloud services in a paas framework," *Scalable Computing: Practice and Experience*, vol. 13, no. 3, pp. 201–214, 2012.
- [56] "FSN OPENCLOUDWARE. LIVRABLE D1.2. T 1.2 MODELE LOGIQUE AVANCÉ." 2013.
- [57] "UNIFIED SERVICE DESCRIPTION LANGUAGE XG FINAL REPORT." [Online]. Available: <https://www.w3.org/2005/Incubator/usdl/XGR-usdl-20111027/>
- [58] "OWL-S 1.2 RELEASE." [Online]. Available: <http://www.ai.sri.com/daml/services/owl-s/1.2/>
- [59] "GRID5000." [Online]. Available: <https://www.grid5000.fr/>
- [60] "OvS : Open vSwitch." [Online]. Available: <http://openvswitch.org/download/>
- [61] "MININET." [Online]. Available: <http://mininet.org>
- [62] "Flowvisor." [Online]. Available: <https://openflow.stanford.edu/display/DOCS/Flowvisor>

- [63] Q. Shen, "Vers un contrôle programmable pour le futur internet et le cloud computing," Ph.D. dissertation, Télécom ParisTech, 2014.
- [64] "Floodlight v1.1." [Online]. Available: <https://floodlight.atlassian.net/wiki/display/floodlightcontroller/Floodlight+v1.1>
- [65] S. Srivastava, S. Anmulwar, A. Sapkal, T. Batra, A. K. Gupta, and V. Kumar, "Comparative study of various traffic generator tools," in *Engineering and Computational Sciences (RAECS), 2014 Recent Advances in*. IEEE, 2014, pp. 1–6.
- [66] "Traffic generator: Iperf." [Online]. Available: <https://iperf.fr>
- [67] "Traffic generator: Mausezahn." [Online]. Available: <http://www.perihel.at/sec/mz/mzguide.html>