QoS-aware NaaS Architecture

Gladys DIAZ

Inès AYADI – (Thèse soutenue le 17 mars 2014)
Noemie SIMONI – Equipe AIRS

Rencontres Vichy 2014

Outline

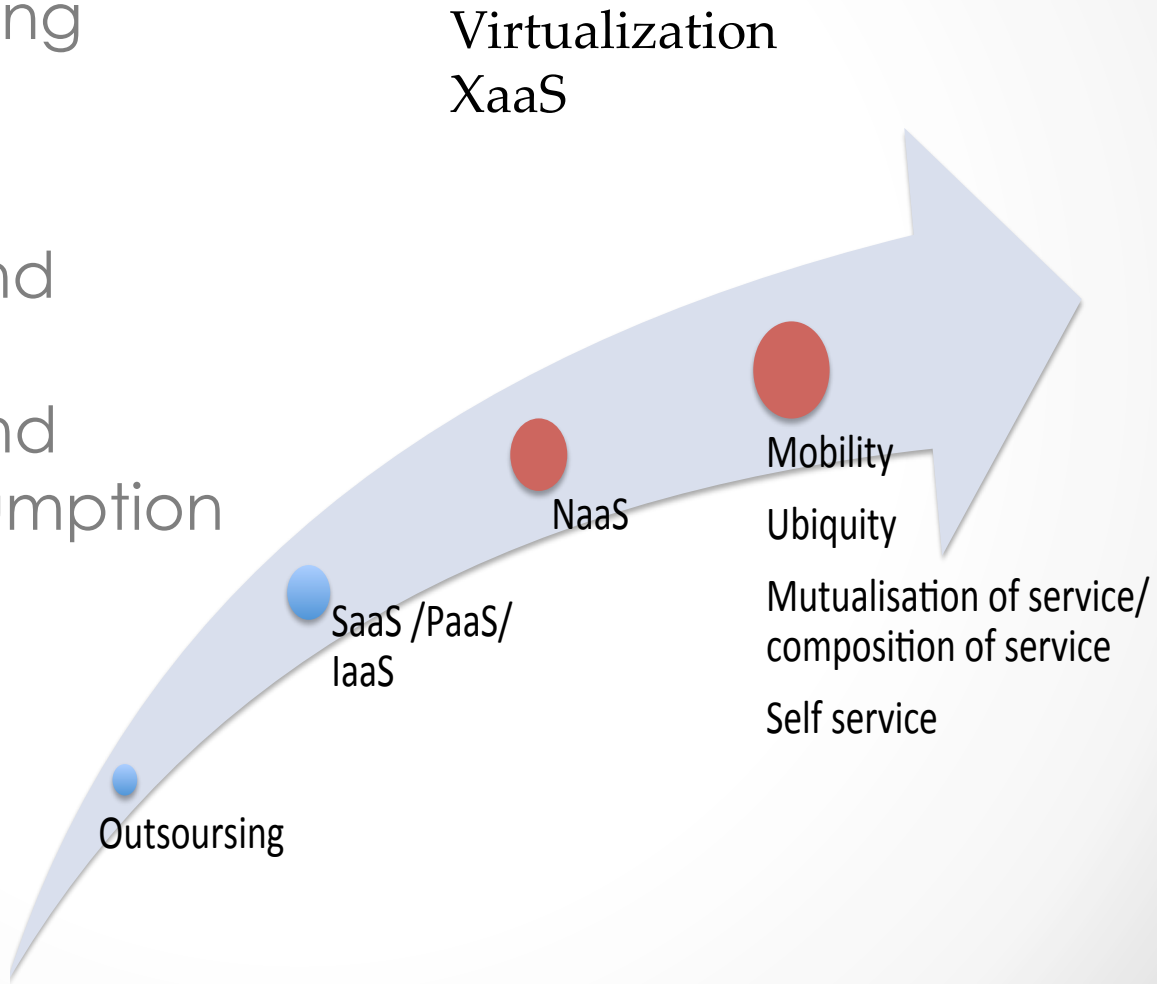
1. Introduction
2. NaaS : Network-as-a-Service
   - Network virtualization
   - SDN
   - NFV
3. QoS-aware for NaaS architecture proposition
   - QoS generic model / Solution: CoS classification
   - Constraints Placement of virtual network
4. Conclusion and perspectives

1. Introduction

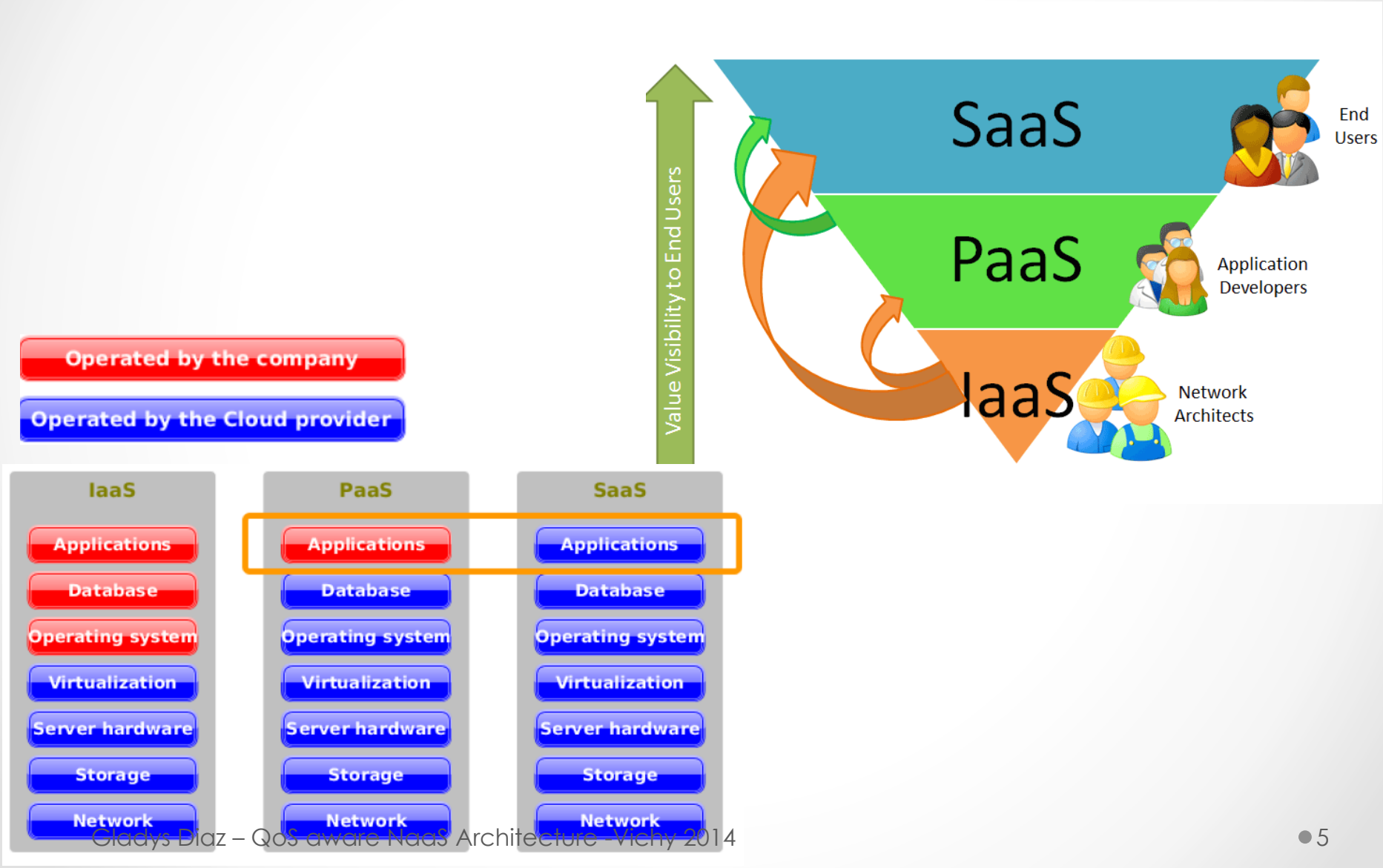
- Contexte : Cloud Networking
- Cloud Networking :
  - Network resources virtualization
- NaaS : Network as a Service
  - the network becomes a utility

Introduction

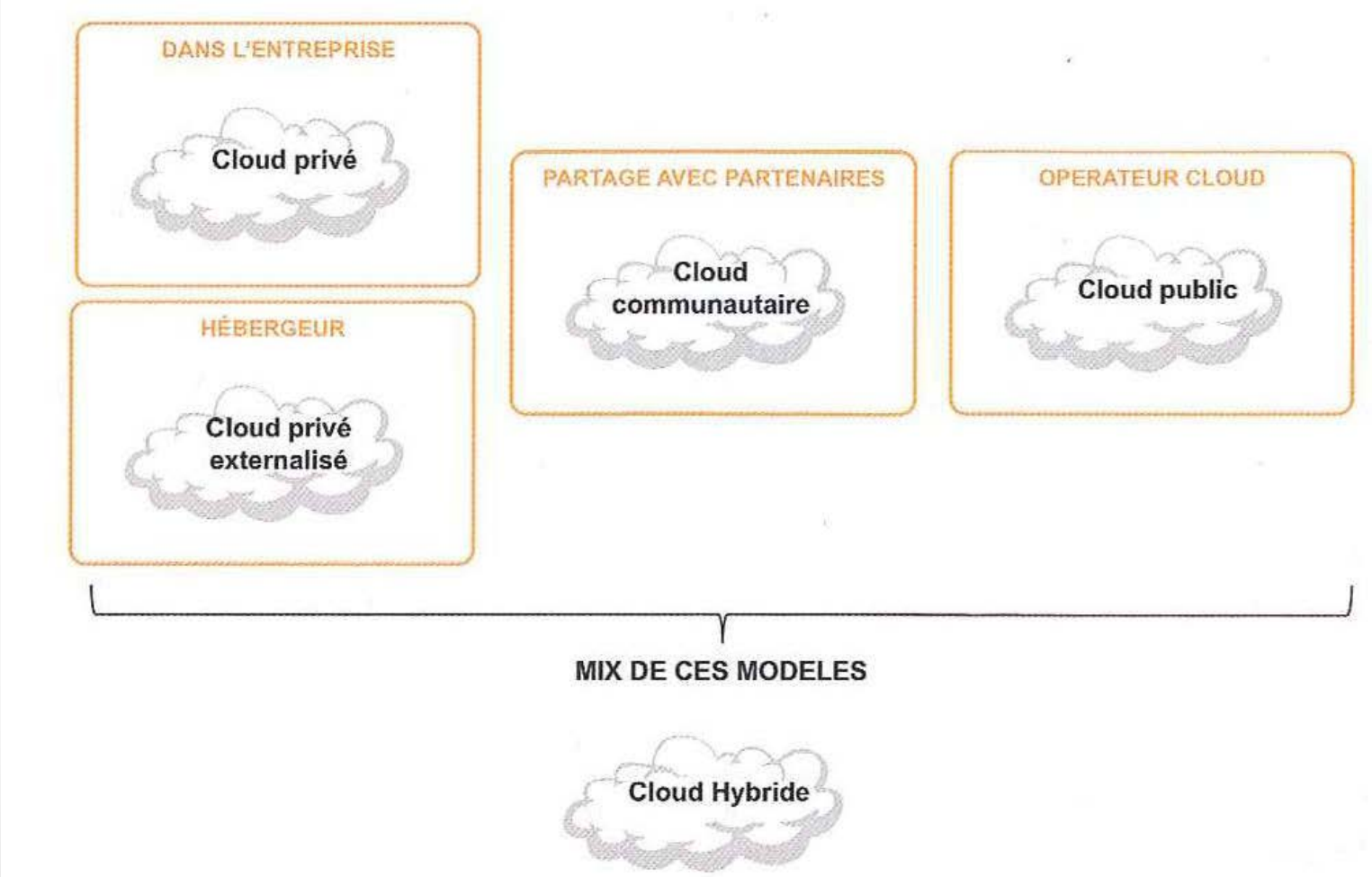
- Cloud computing
- Dynamicity, adaptability and flexibility in deployment and resource consumption



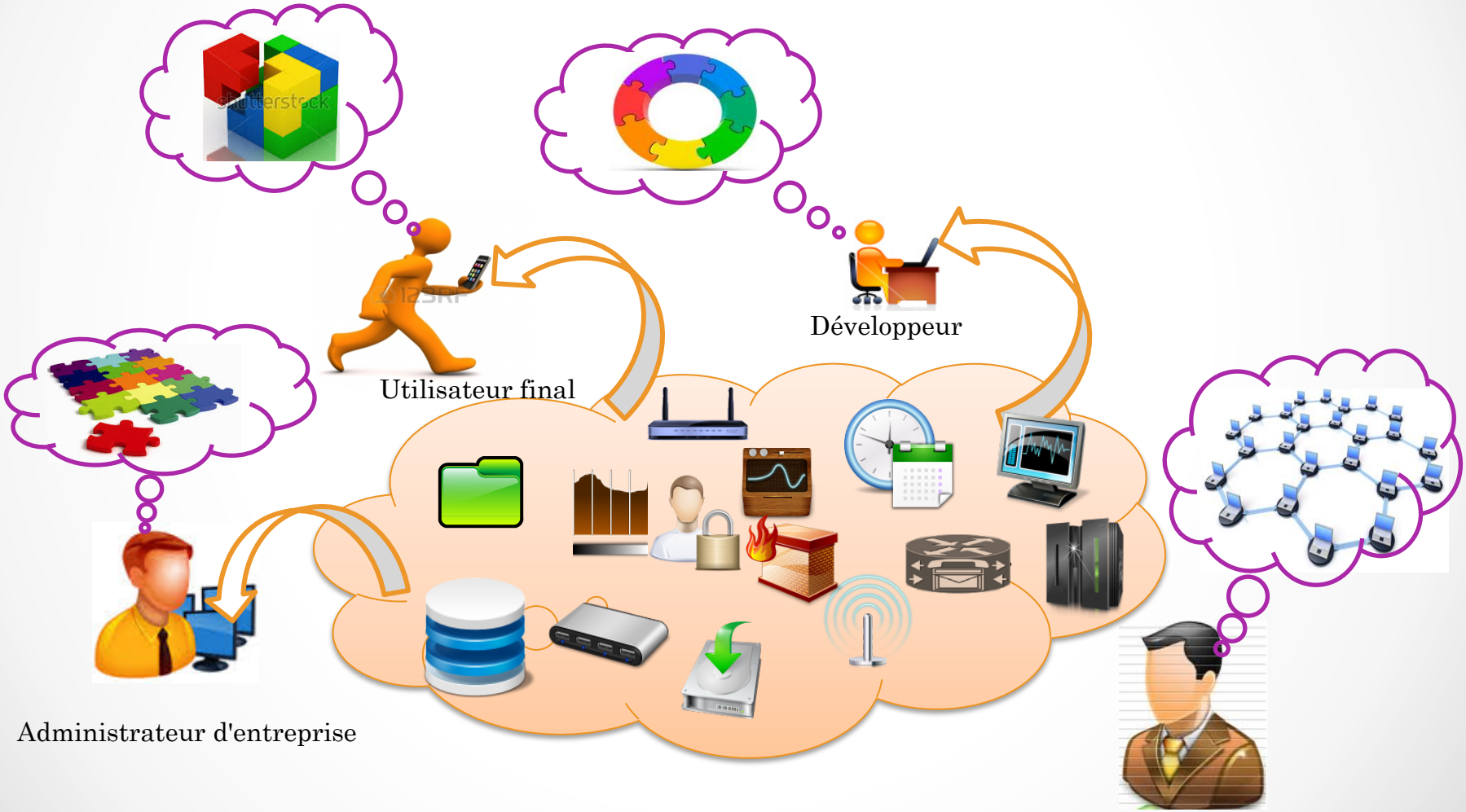
Cloud solution levels



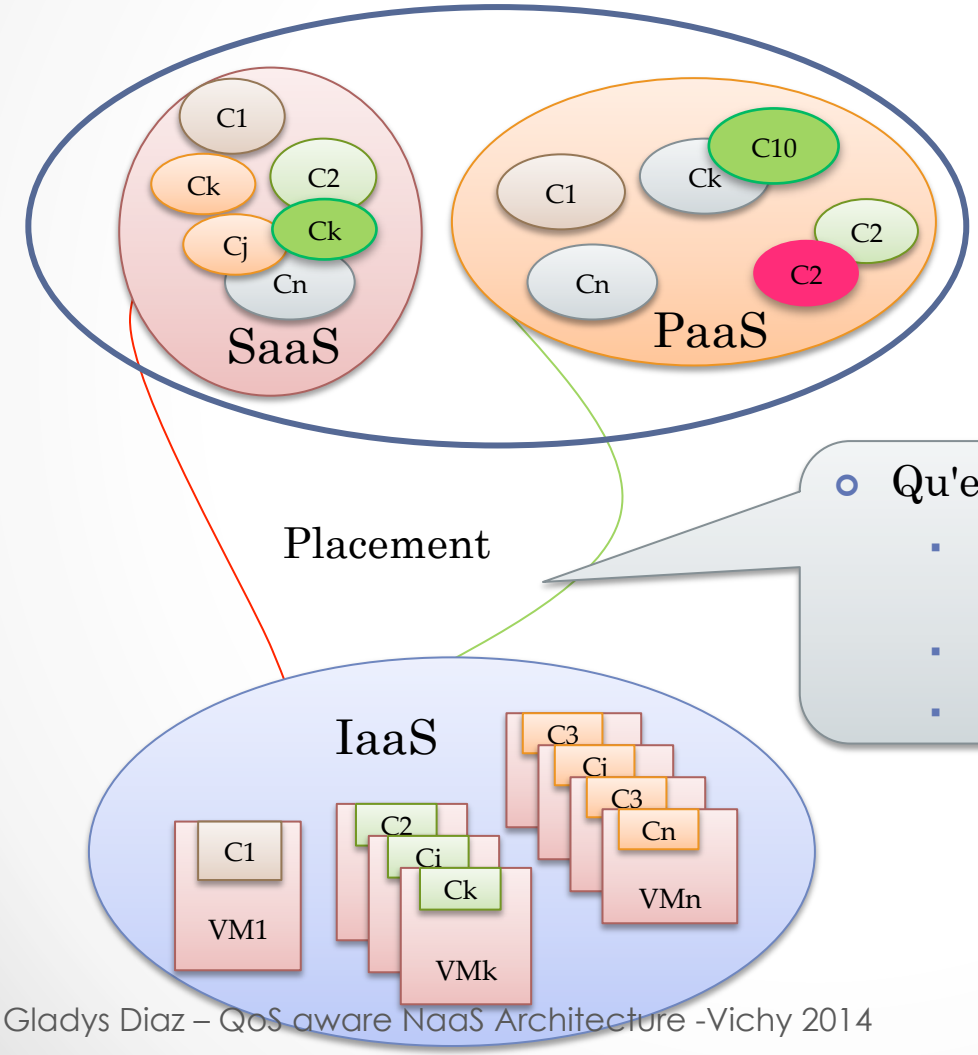
Cloud Types


DANS L'ENTREPRISE
Cloud privé
HÉBERGEUR
Cloud privé externalisé
PARTAGE AVEC PARTENAIRES
Cloud communautaire
OPERATEUR CLOUD
Cloud public
MIX DE CES MODELES
Cloud Hybride

Introduction



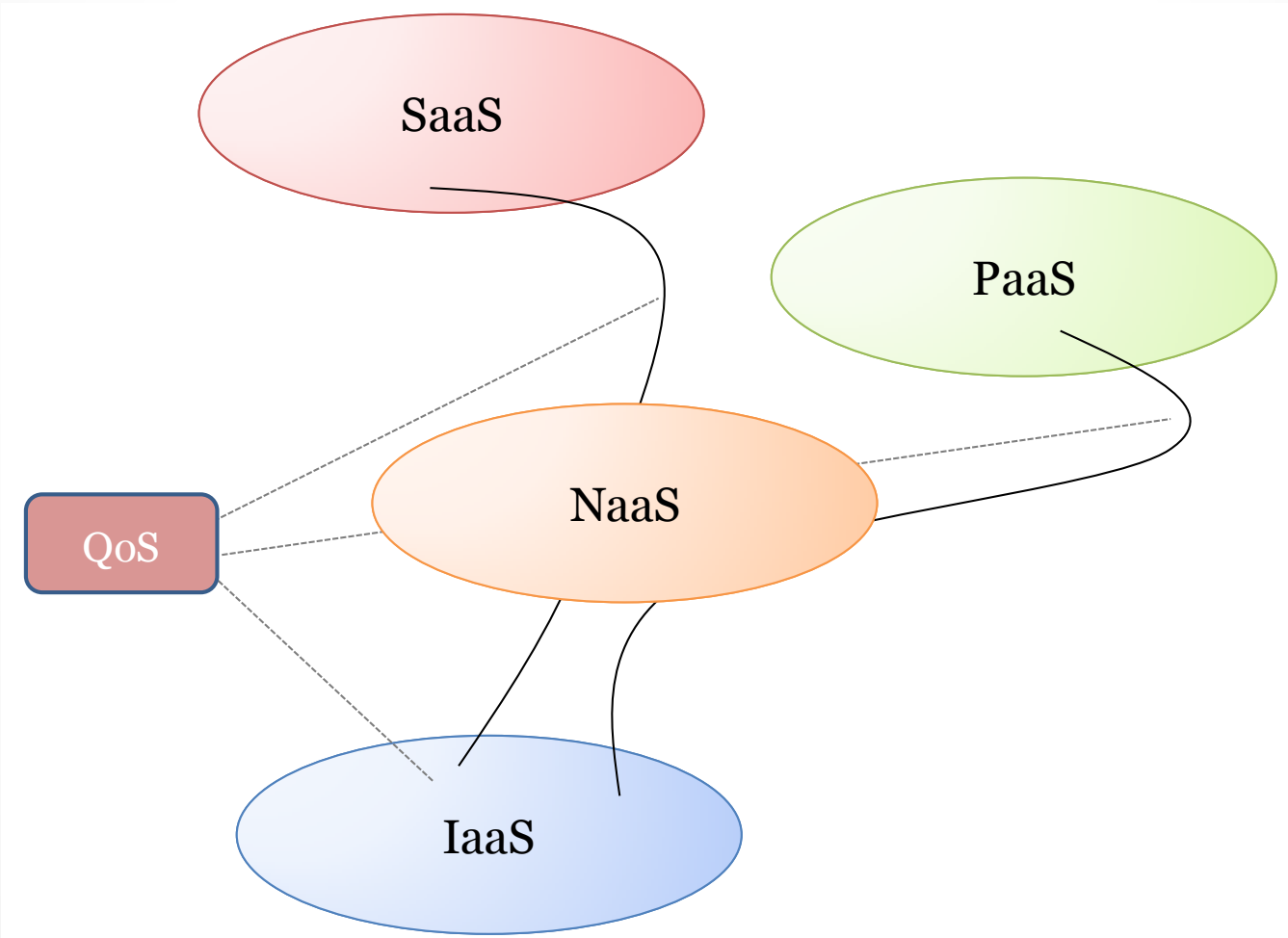
Problems



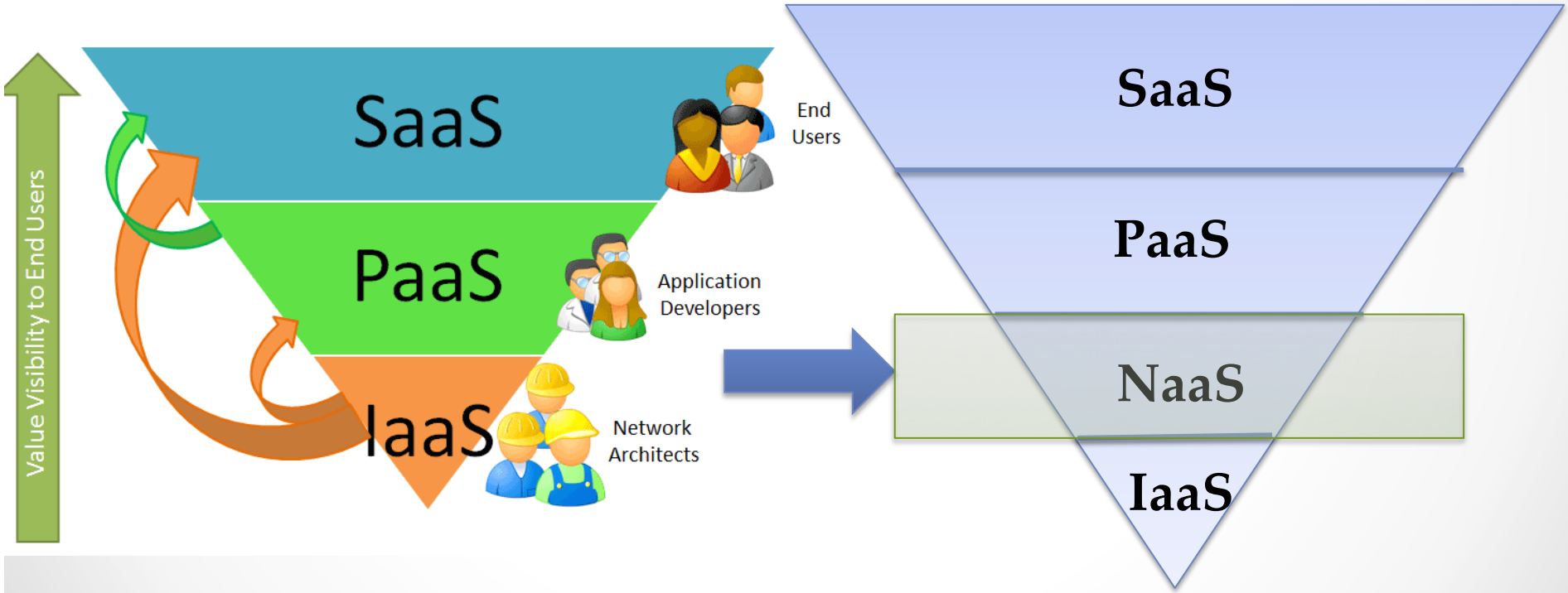
But the network is rarely offered as a service and is rather considered as a part of the infrastructure and thus is not provided "on demand", it is often based on statistical solutions.

- Qu'en est-il du réseau?
  - Configuration statique (VPN, VLAN, VPC)
  - Sur/sous provisionnement
  - Pas de garanties de QoS

Contexte



Contexte: NaaS



3. NaaS

- Network-as-a-Service
- introduces the challenge of provide network functions as a set of cloud networking services
- involves the fact to select appropriate network components' in according to application requests
- approches: virtualization, SDN et VNF

Virtualisation des réseaux

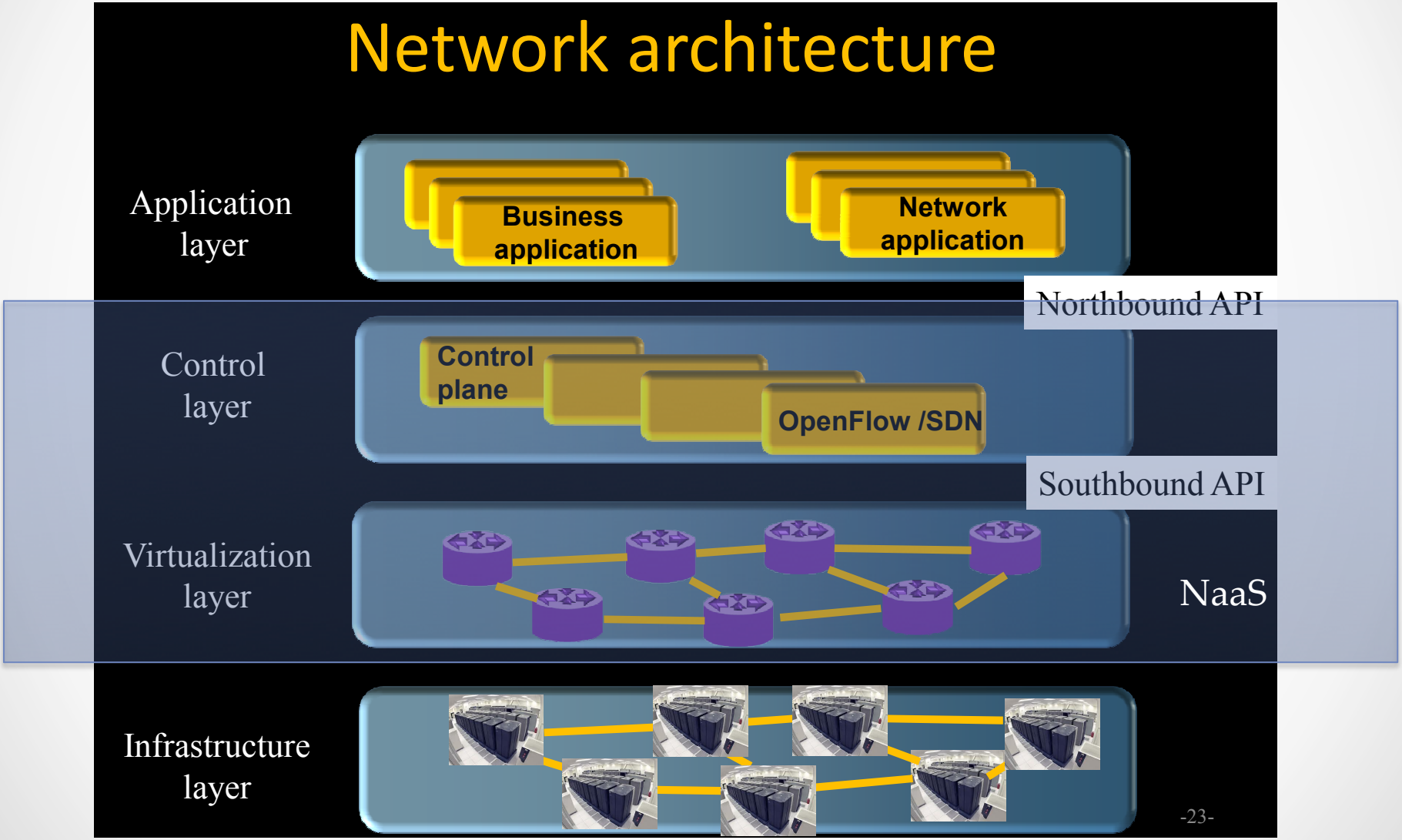
- Aujourd'hui : vision équivalent à la virtualisation des machines
  - Méthode permettant le partage de ressources disponibles (bande passante)
  - Virtualisation des équipement réseaux (routeurs, switch)

- En évolution : réseau programmable
  - Découplage des plan de control et de données (SDN)
  - Virtualisation des fonctions réseaux (VNF)

SDN
Software-Defined Networking

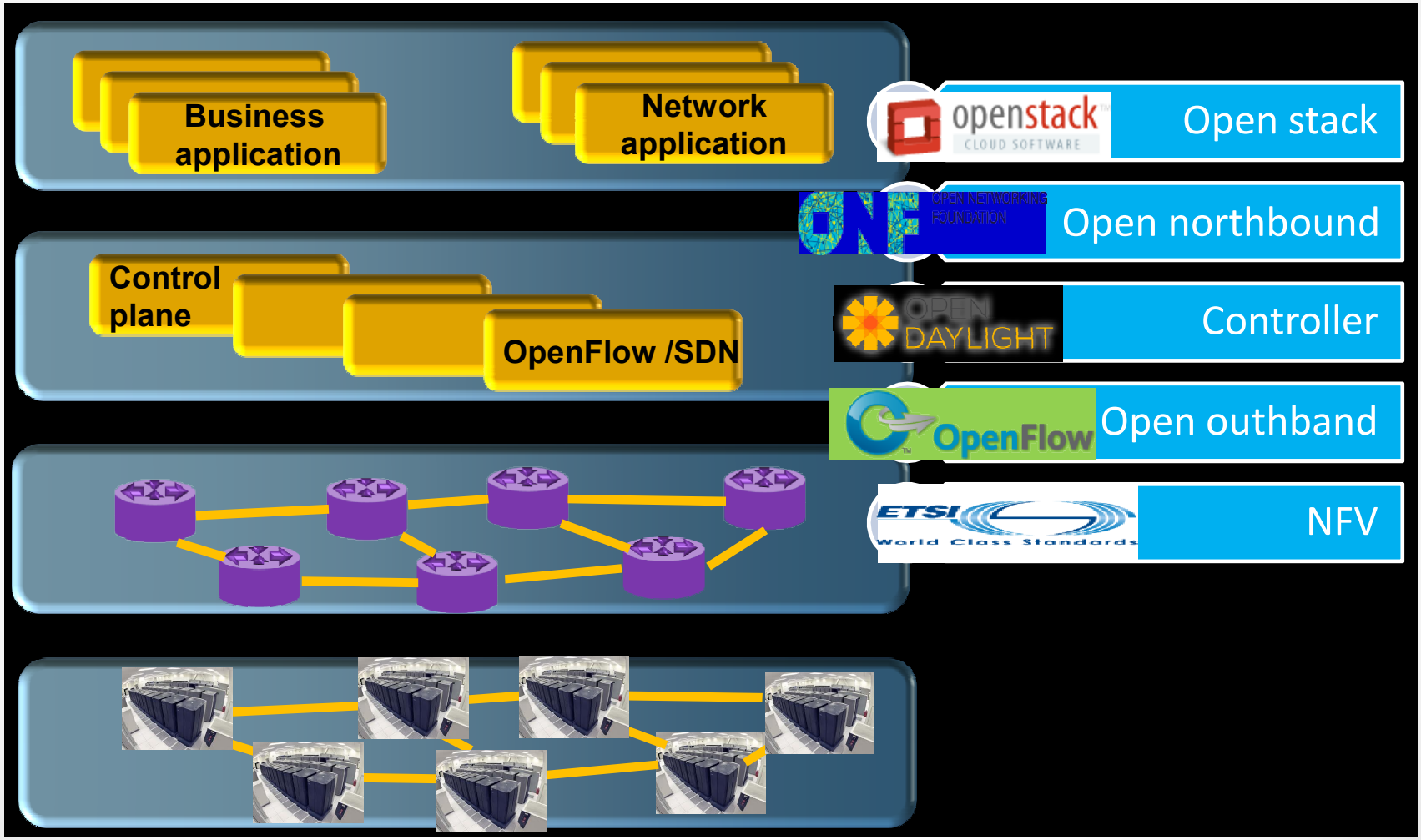
- The control and data planes are decoupled, and is directly programmable
- The underlying network infrastructure is abstracted from the applications and networks services
- Network services can treat the network as a logical or virtual entity.
- Network intelligence is (logically) centralized in software-based SDN controllers that maintain a global view of the network,
- Programmability, automation, and network control, enabling them to build highly scalable, flexible networks that readily adapt to changing business needs.
- Open standards-based and vendor-neutral
- The OpenFlow protocol is a foundational element for building SDN solutions.

Network Architecture Background



Source: Systematic-SDN&NFV-KeyNote-UMPC-Pujolle

Background -technologies



Source: Systematic-SDN&NFV-KeyNote-UMPC-Pujolle

Background-technologies

- OpenStack :
  - (https://www.openstack.org)
  - is an open and scalable operating system for building public and private clouds.
- OpenDaylight: Open Source Programmable Networking Platform
  - (http://www.opendaylight.org)
  - open platform for network programmability to enable SDN and create a solid foundation for NFV for networks at any size and scale
  - The Northbound (programmatic) and Southbound (implementation) interfaces are clearly defined and documented APIs.
- Openflow:
  - (https://www.opennetworking.org)
  - Standard Protocole to SDN
  - Communication: control plane and forwarding plane

NFV
Network Functions Virtualization

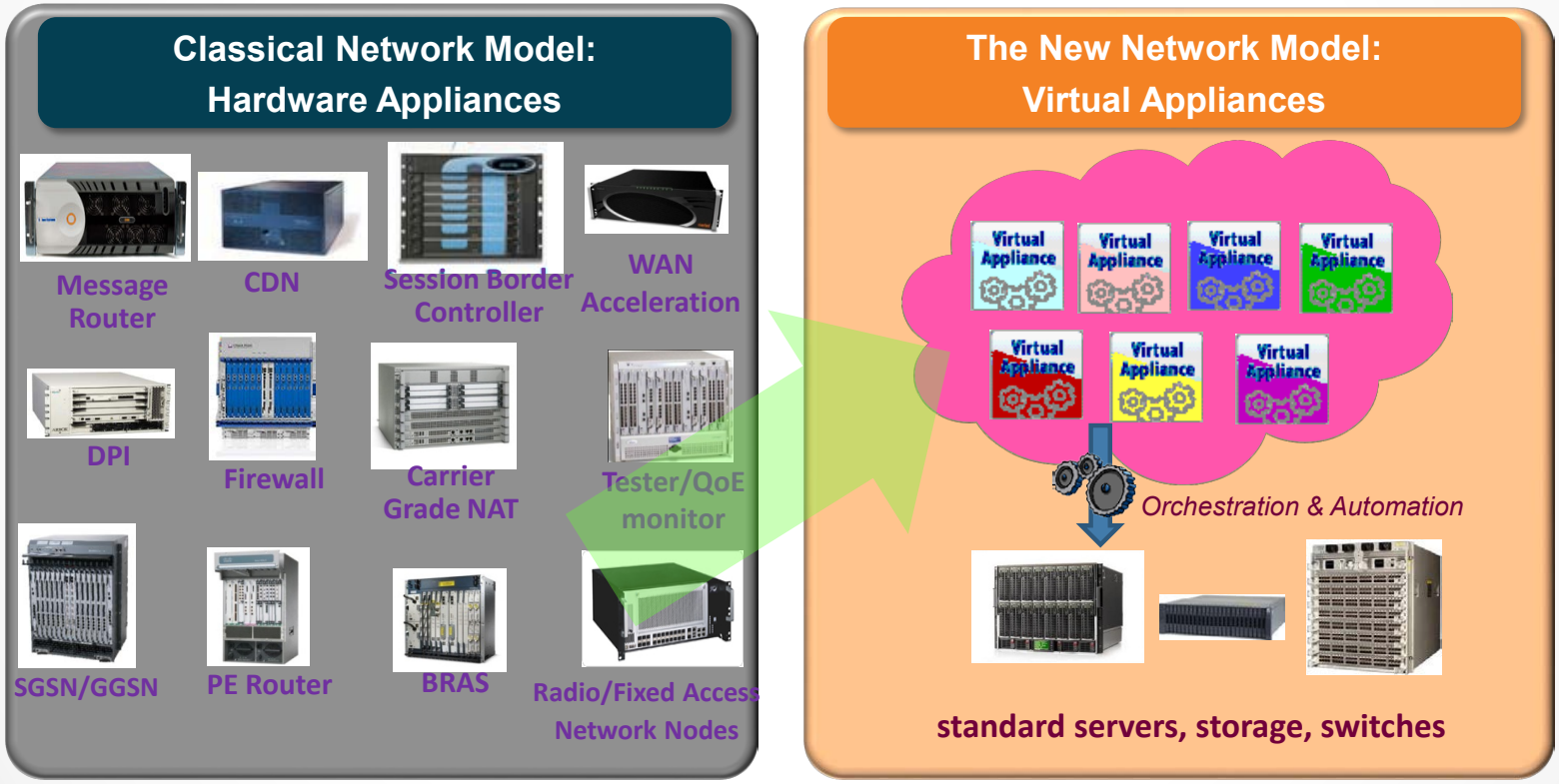
- Proposed in 2012 by **ETSI NFV ISG**
- By virtualizing and consolidating network functions traditionally implemented in dedicated hardware, using cloud technologies, network operators expect to achieve greater agility and accelerate new service deployments
- applicable to any data plane packet processing and control plane function in fixed and mobile network infrastructures.
- Utilize resources more effectively;
- Virtualization allows providers to allocate only the necessary resources needed by each feature/function.

Source: www.etsi.org/nfv

Source: ETSI NFV

NFV

- *implementing network functions in software - that (today) run on proprietary hardware - leveraging (high volume) commodity servers and IT virtualization*

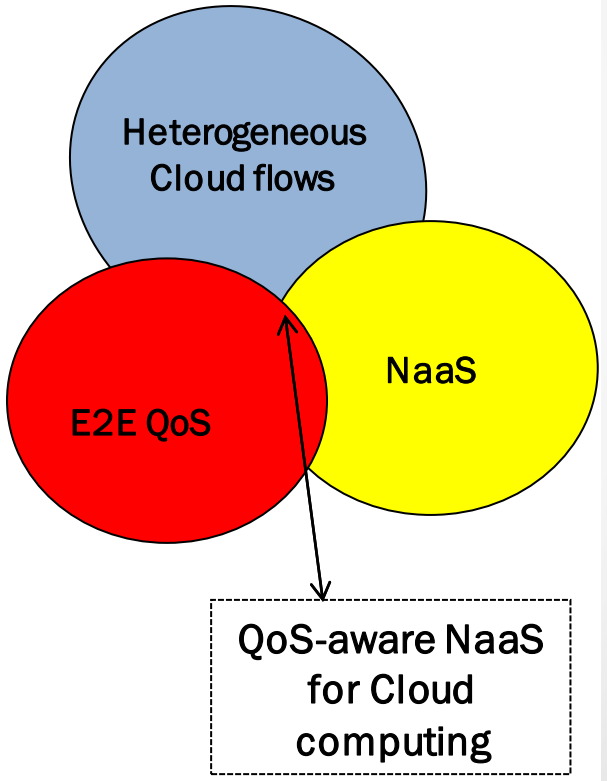


- Network Functions are **based on specialized hardware**
- **One physical node per role. Physical install per site**
- **Static. Hard to scale up & out**

- Network Functions are **SW-based**
- **Multiple roles** over **same HW. Remote operation**
- **Dynamic. Extremely easy to scale**

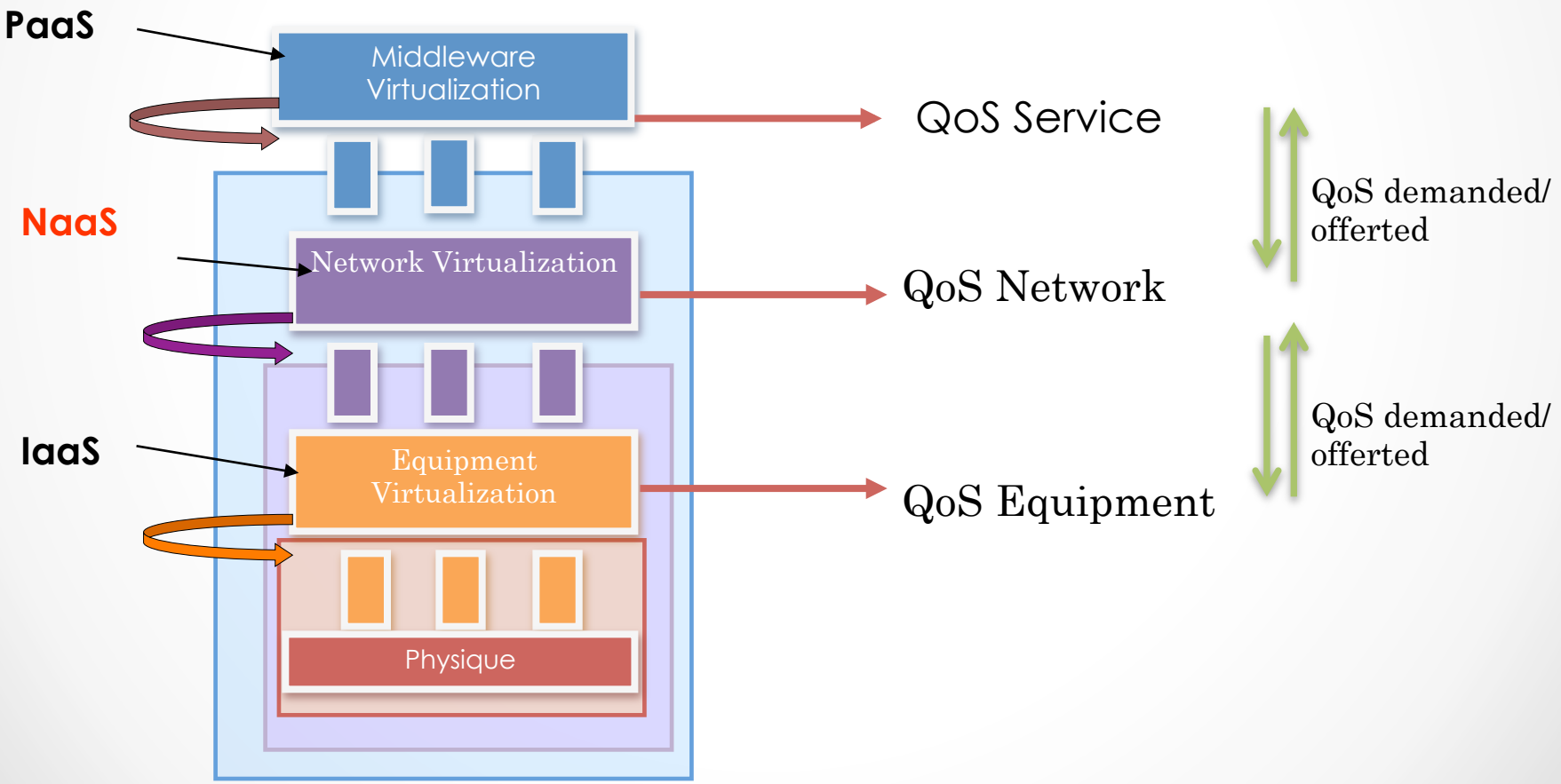
Objectives

- Network virtualization NaaS: recent Cloud model
  - SDN/NVF complementary
  - As-a-service model
- E2E QoS aware
  - User-side: SLA
  - Provider-side: avoid violations
- Heterogeneous Cloud flows:
  - Application flows, aggregation flows caracterization
  - QoS sensitivity: availability, delay, capacity, reliability



4. QoS-aware NaaS Proposition

End-to-end Virtualization

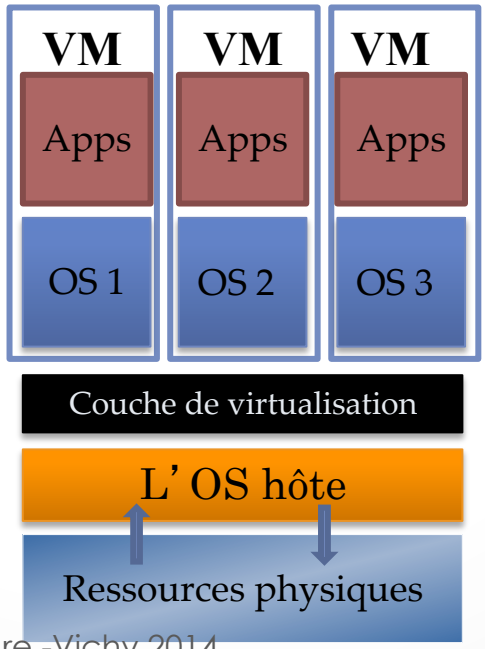


Levels of virtualisation

Niveaux	Virtualisation
Equipment	Virtualisation des machines, des mémoires, de CPU, etc.
Network	Virtualisation des routeurs, des points d'accès, des liens (projets: 4ward, CABO, PlanetLab, VINI, etc.)
Middleware	Virtualisation des serveurs

Physical resources virtualization

~~System-centrique~~

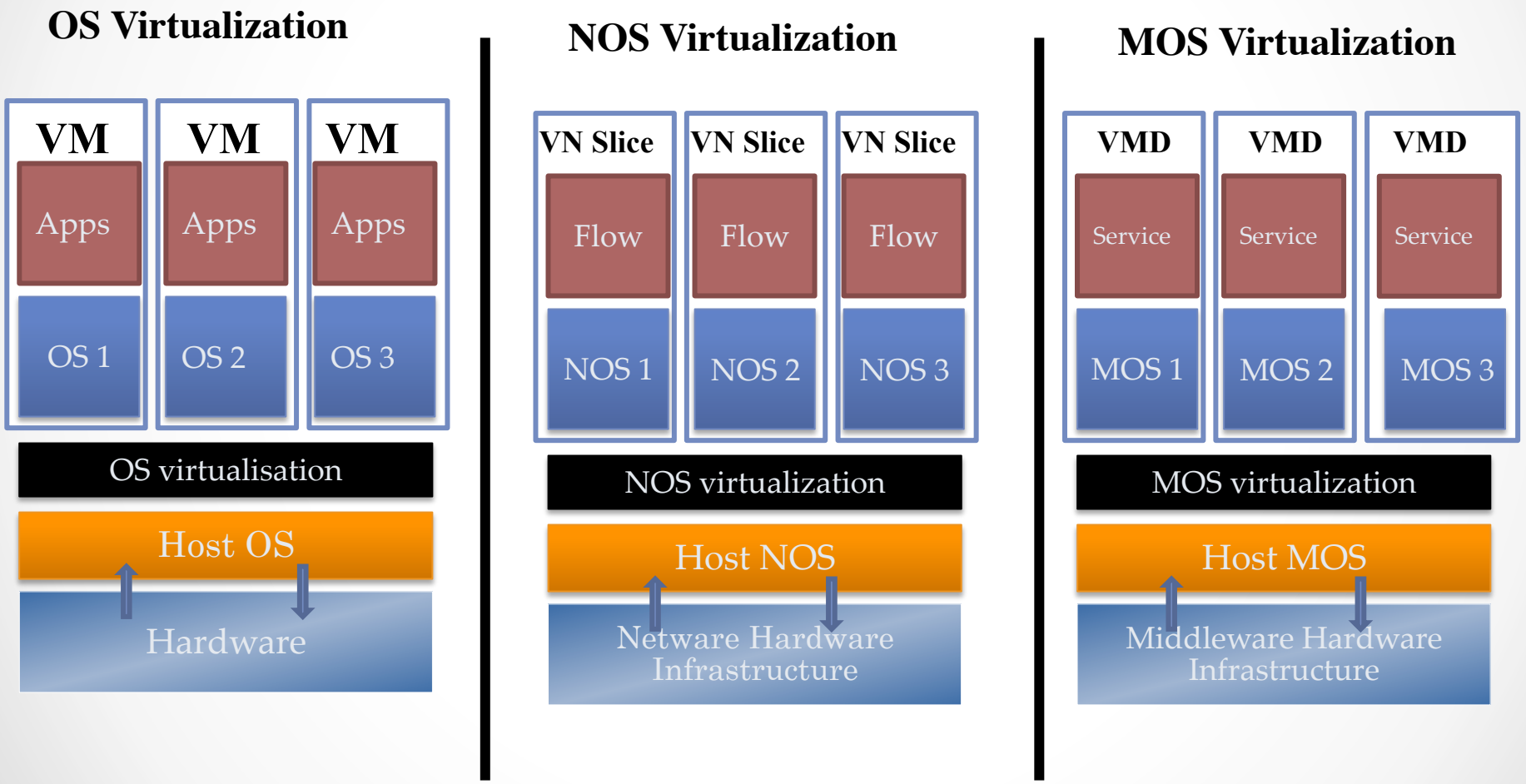


~~Network-centrique?~~

~~Application-centrique?~~

Proposition

End-to-end Virtualization

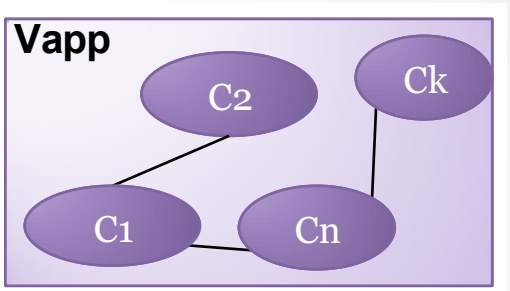


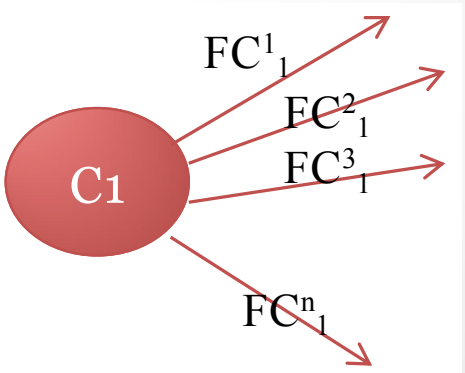
Proposition Network Virtualization

Virtualization	OS	NOS
Shared resources	CPU, memory, I/O	NIC, Link, network equipment
Unit	VM	VN Slices
Isolation	Betweent VMs	Betweent network flows
Customization	Applications	Protocole Stacks

Proposition Network Virtualization

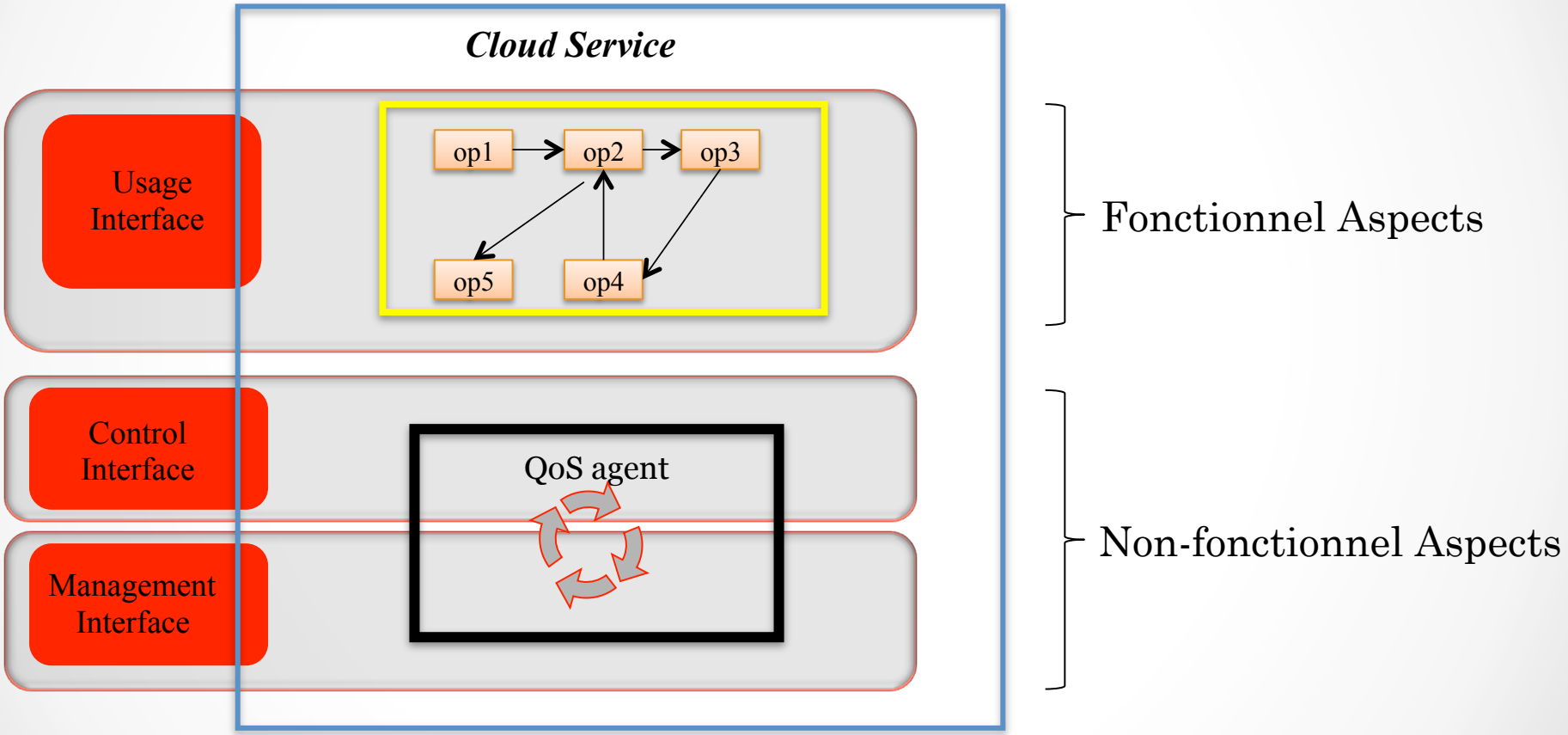
- Cloud application: a set of components
- Application's component: various output flows
- Flow: is a service
  - Behavior: QoS sensitive
  - Differentiate classes of service (VN-CoS)
- Cloud networking: Flows aggregation
  - Network functions: are as services
  - Behavior: QoS criteria





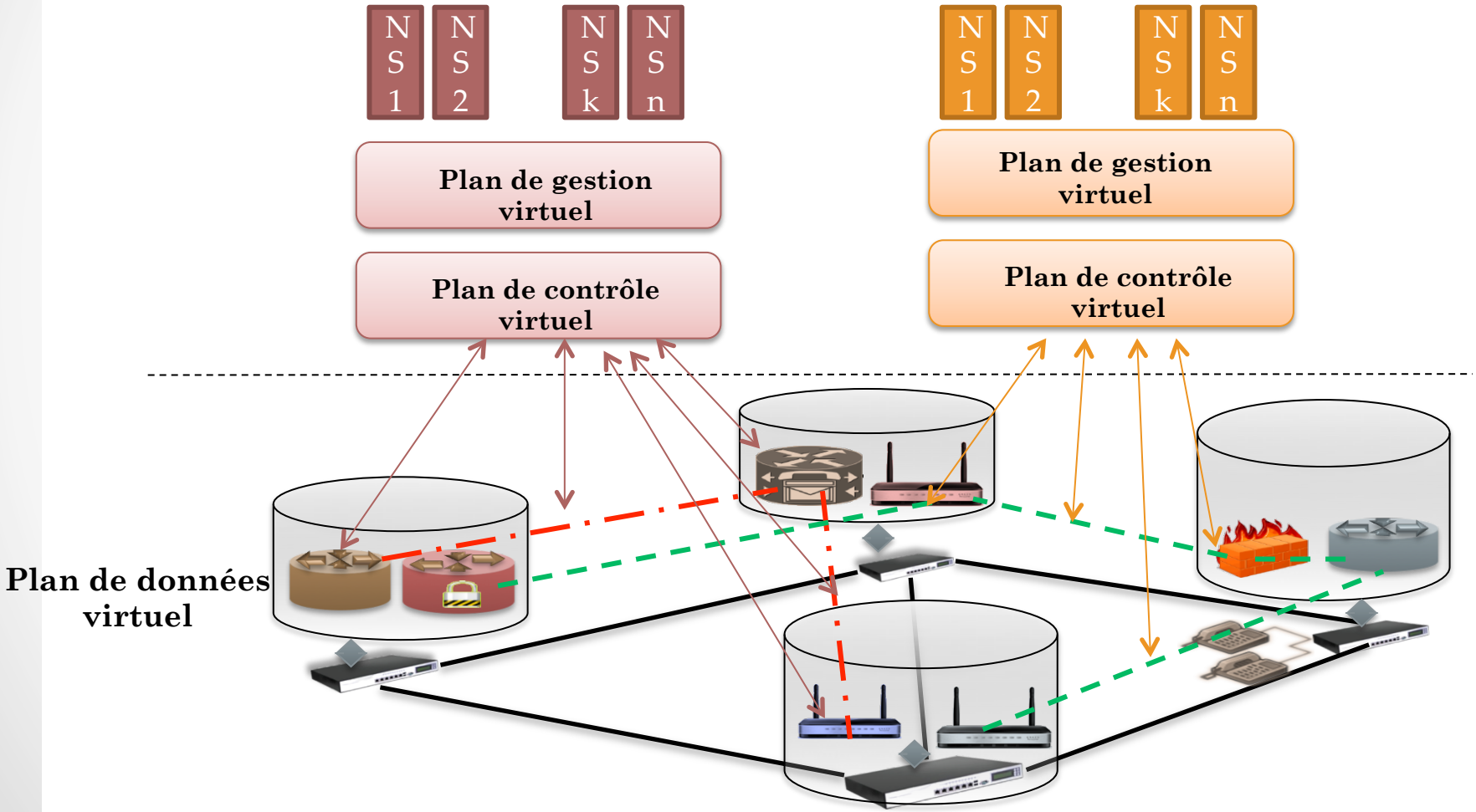
QoS Aware NaaS

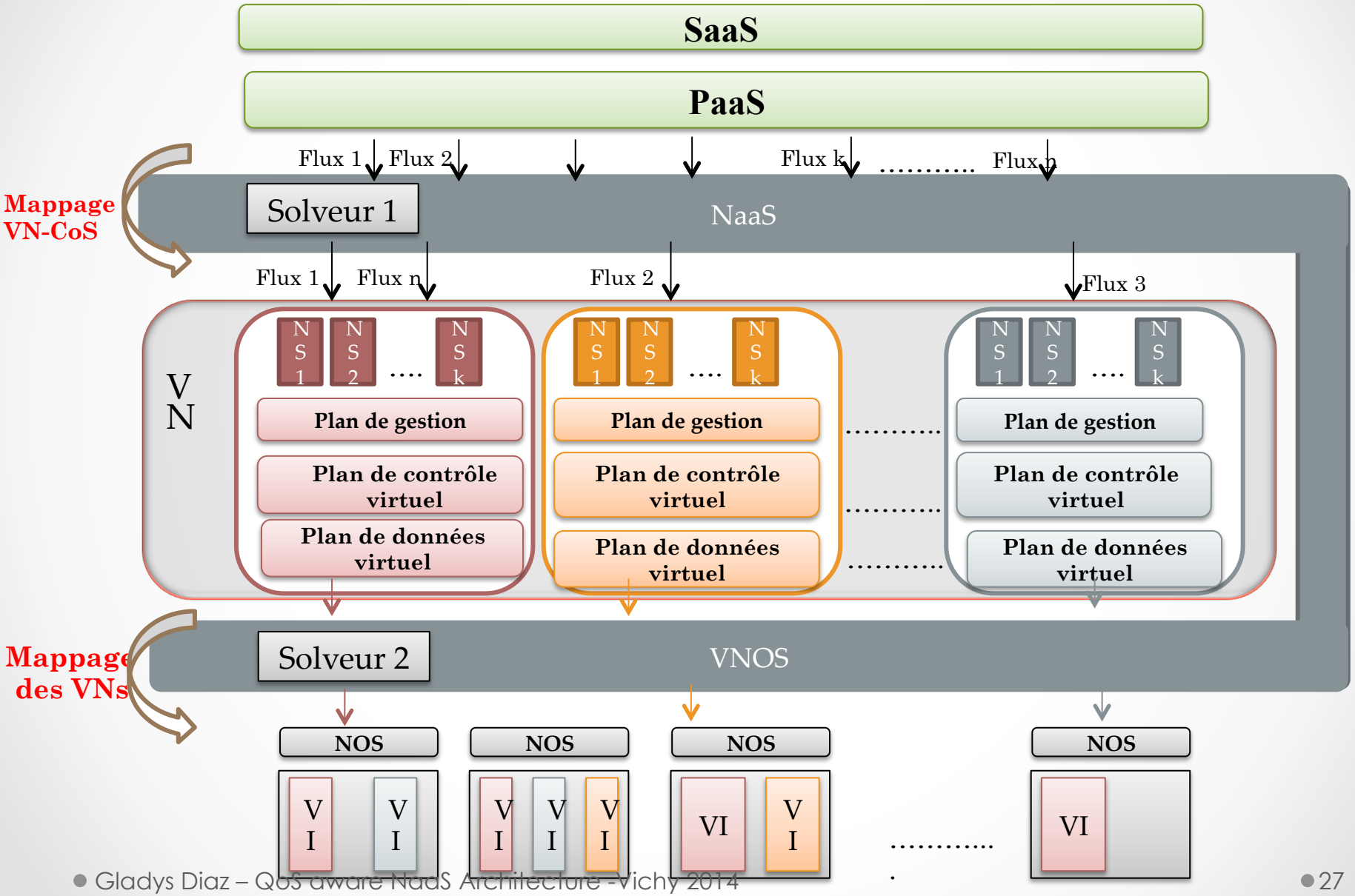
RE-thinking Cloud Services: Service Model



Independant, QoS-aware et self-management

NaaS Proposition




SaaS
PaaS
Flux 1
Flux 2
Flux k
Flux n
Mappage
VN-CoS
Solveur 1
NaaS
Flux 1
Flux n
Flux 2
Flux 3
V
N
NS 1
NS 2
NS k
Plan de gestion
Plan de contrôle virtuel
Plan de données virtuel
Mappage
des VNs
Solveur 2
VNOS
NOS
VI

QoS-Aware for NaaS

QoS generic model

- A behavior consists in the non-functional aspects of cloud networking services It is defined by four QoS criteria (sufficient and necessary)

Transparency levels	QoS criteria
• **Temporal transparency:** a given information can be delivered anytime	• **Availability**: accessibility rate of the network service component
• **Distance transparency:** a given information can be delivered regardless of the distance between end-nodes	• **Delay** : time for request processing
• **Spatial transparency:** a given information can be delivered regardless of its volume	• **Capacity:** processing capacity of network service during a unit of time
• **Semantic transparency**: a given information can be delivered without alteration of its content	• **Reliability:** the compliance rate of running without alteration of information

QoS criteria are useful for: management, security, monitoring, performance evaluation

QoS-Aware for NaaS

Class Of Service for NaaS

- Cloud applications' flows:

- Sensitivity to QoS criteria:
- Availability
- Delay
- Capacity
- Reliability

- Sensitivity degree:
- Low→x
- Medium→xx
- High→xxx

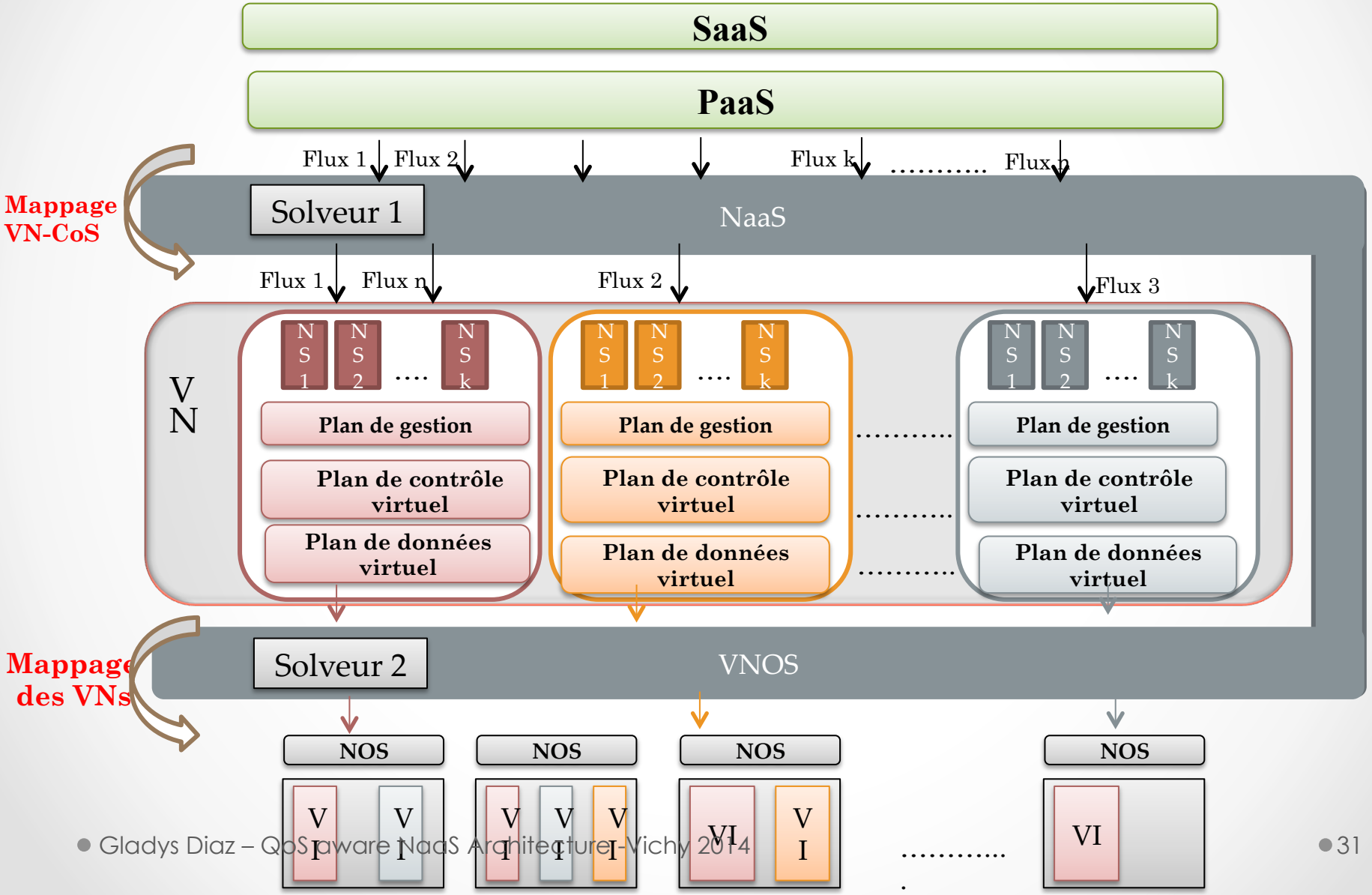
Flow sensitivity			
Availability	Delay	Capacity	Reliability
x	x	x	x
x	xx	x	x
x	xxx	x	x
x	x	xxx	x
xxx	x	xx	xxx
x	xxx	xxx	x
...	...	...	...
xx	xx	xx	x
...	...	...	...
xxx	xxx	xxx	xxx

VN-CoS Protocol stack selection

	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

Proposition

NaaS Architecture



Proposition

Catalogue of placement constraints (node)

Contraintes de haut niveau	
NodeAffinity (v : set <VNE>)	Tous les nœuds virtuels qui appartiennent au groupe v doivent être mappés sur le même nœud physique
NodeAnti-Affinity (v : set <VNE>)	Tous les nœuds virtuels au sein du groupe v doivent être mappés sur des nœuds physiques distincts
NodeIsolation(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	
NodeTreatmentTime(v1 : VNE, d : time)	Le temps de traitement de v1 doit être inférieur à d
NodeAvailability(v1 : VNE, α : rate)	Le taux de disponibilité de v1 doit être au moins égal à α
NodeReliability(v1 : VNE, β : rate)	Le taux de fiabilité de v1 doit être au moins égal à β
NodeCapacity(v1 : VNE, c : String, z : number)	La capacité de type c du nœud v1 doit être au moins égale à z

Syntaxe: Btrplace (OpenCloudware project)

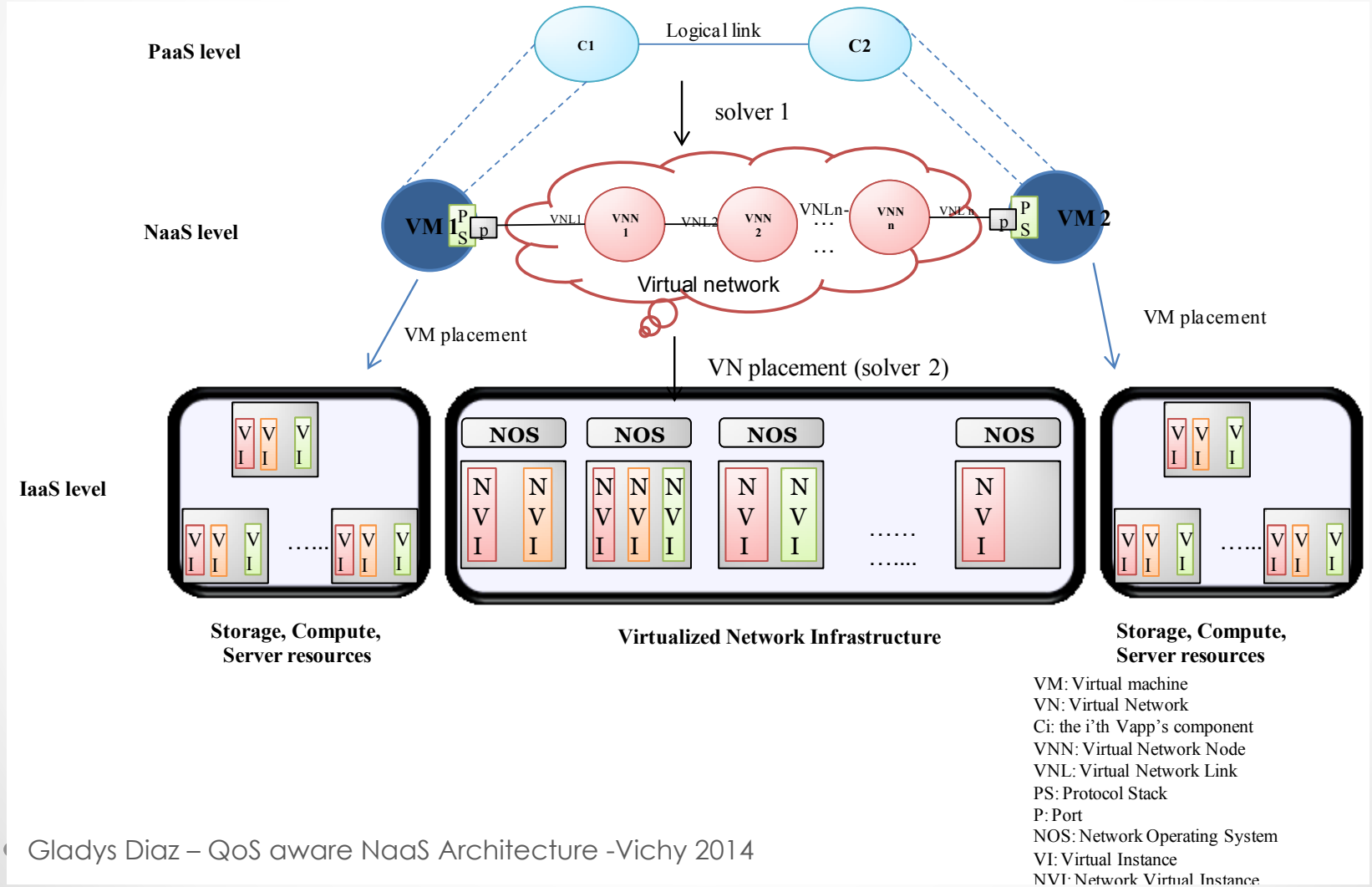
Proposition

Catalogue of placement constraints (link)

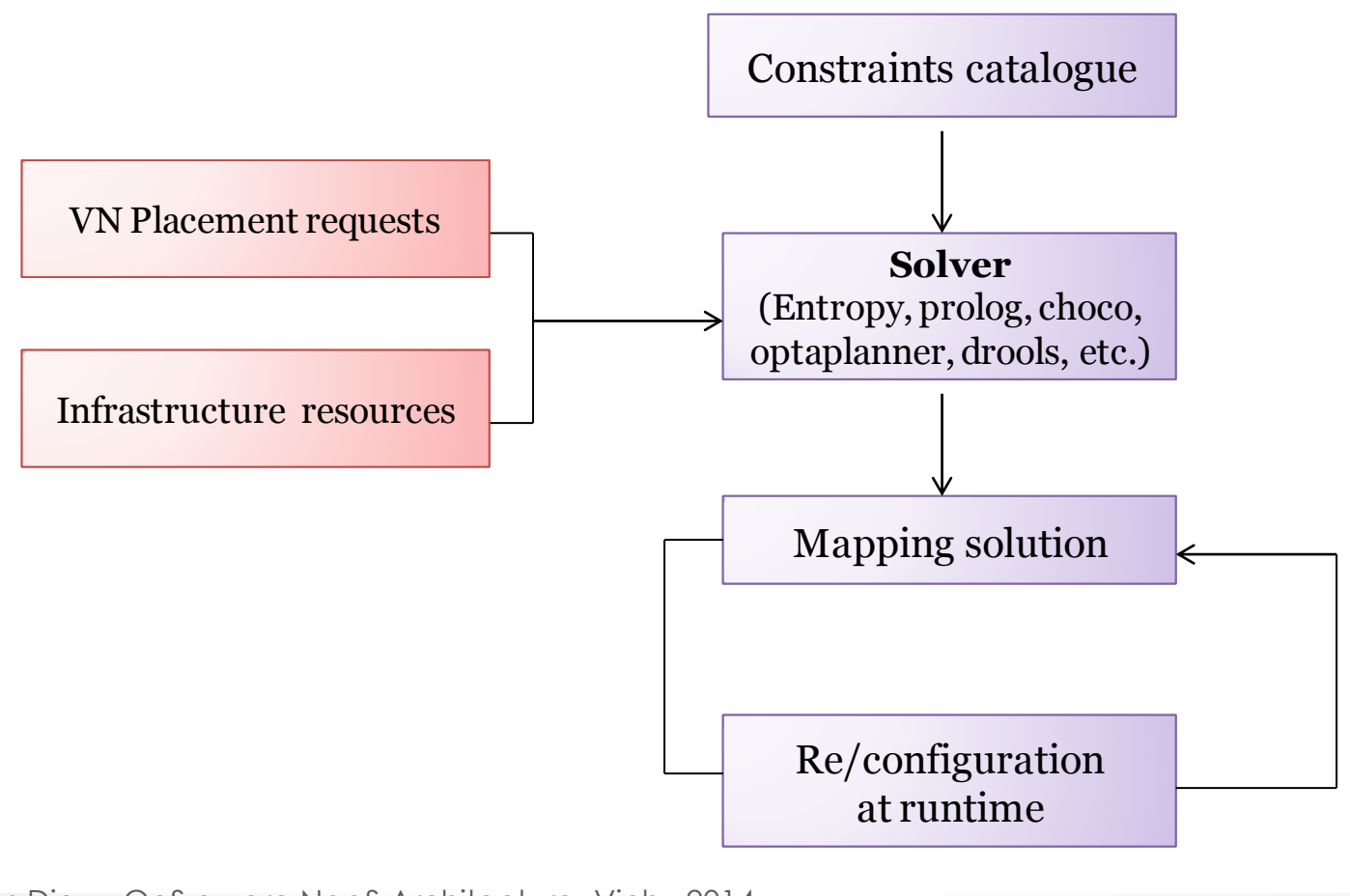
Contraintes de haut niveau	
BanPath(v1 : VNE, v2 : VNE, s : set<INNode>)	Le chemin entre v1 et v2 ne doit pas traverser les nœuds intermédiaires appartenant à s
IsoPath(v1 : VNE, v2 : VNE, nb : number)	Le nombre de nœuds intermédiaires entre v1 et v2 ne doit pas dépasser nb
Contraintes de QoS	
MaxLinkDelay(v1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

Syntaxe: Btrplace (OpenCloudware project)

Virtual network placement

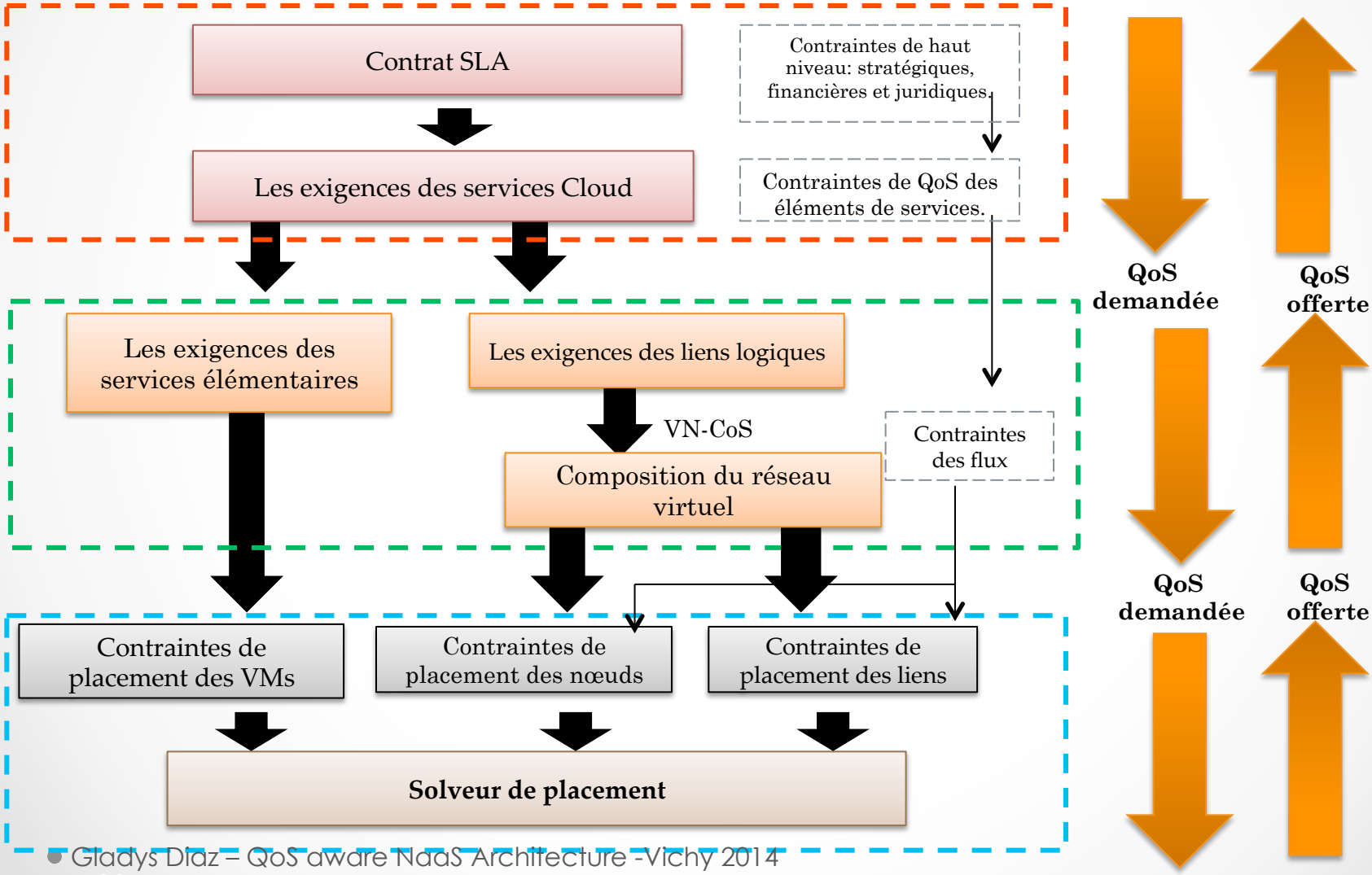


Constraints programming



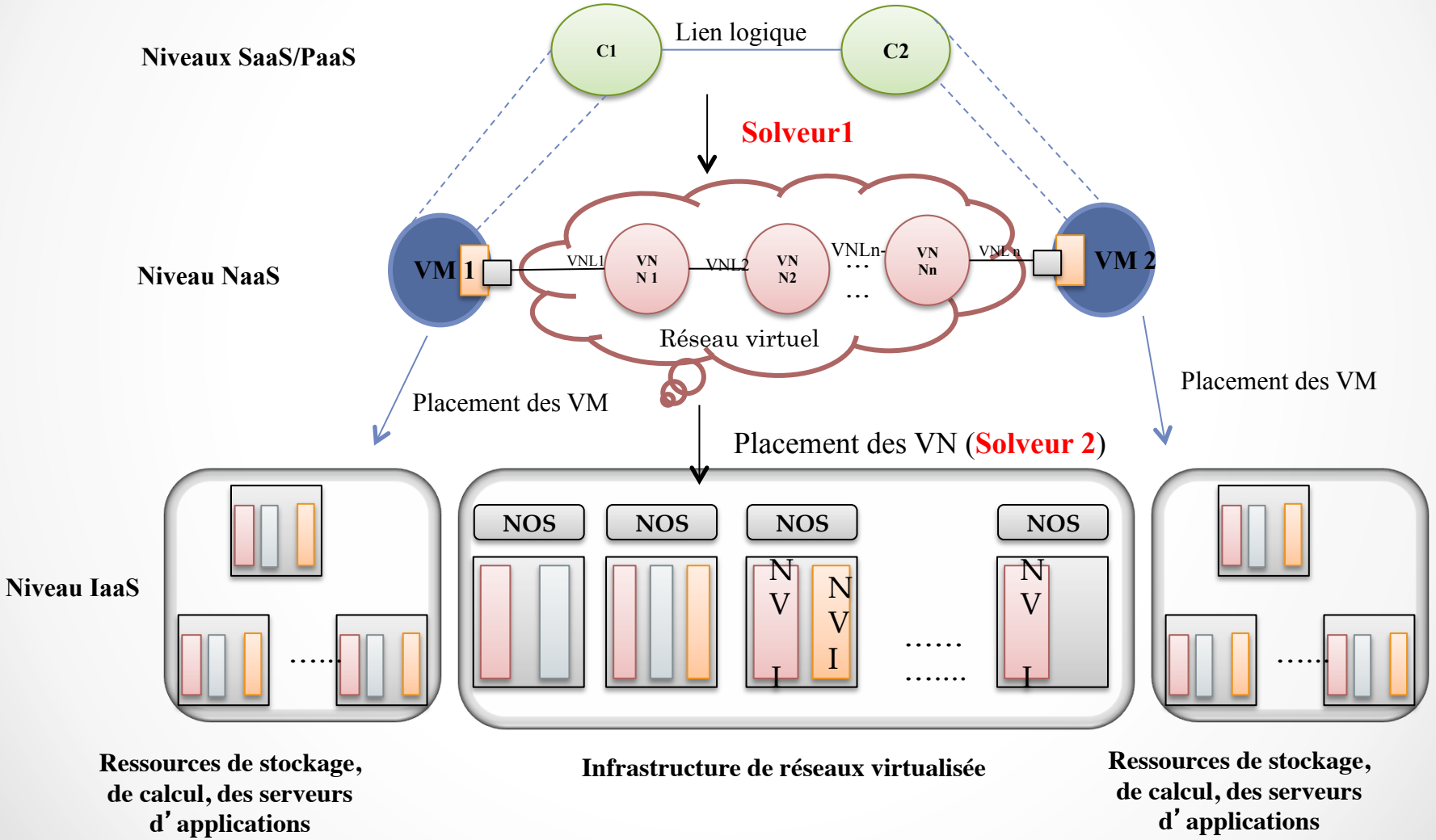
PROPOSITION

INFRASTRUCTURE PLACEMENT



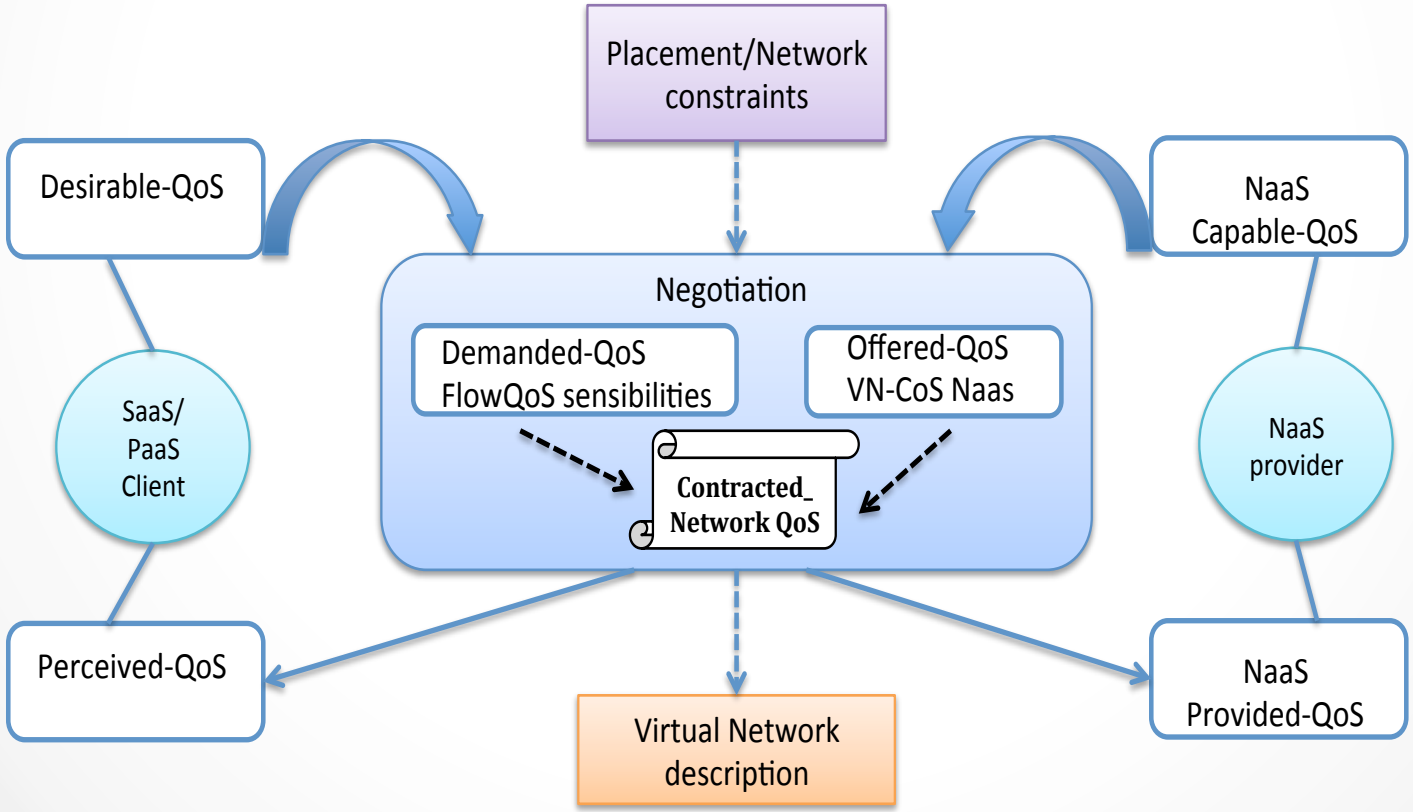
PROPOSITION

INFRASTRUCTURE PLACEMENT



In progress results

- Building and dynamic deployment process of NaaS services



In progress results

- OVF extension – Flow sensibilities (INPUT to NaaS)

```
<ovf:FlowDescription>
<ovf:QoSSensibilities>
      <ovf:Availability ovf: Parameter="Parameter0"  ovf: Value="Value0"
ovf: Degree="HIGH"/ovf:Availability>
      <ovf:Delay ovf:Parameter="Parameter1" ovf:Value="Value1"
ovf:Degree="HIGH"/ovf:Delay>
      <ovf:Reability ovf:Parameter="Parameter2" ovf:Value="Value2"
ovf:Degree="HIGH"/ovf:Reability>
      <ovf:Capability ovf:Parameter="Parameter3" ovf:Value="Value3"
ovf:Degree="HIGH"/ovf:Capability>
  </ovf:QoSSensibilities>
  <ovf:Application>
  </ovf:Application>
</ovf:FlowDescription>
```

In progress results

- OVF extension (VN-OVF++) – Virtual Network description (Output from NaaS)

Virtual Link Section

```
<ovf :NetworkSection>
<ovf :VirtualNetworkSection>
    <ovf :VirtualLinkSection ovf :VirtualLinkName="VirtualLinkName0" ovf :Active="false">
      <ovf :TransitionNodes>
        <ovf :TransitionNodeNumber>2</ovf :TransitionNodeNumber>
        <ovf :IntermediateNodeIsolation>
          <ovf :ListNodes>  VN1, VN2  </ovf :ListNodes>
        </ovf :IntermediateNodeIsolation>
      </ovf :TransitionNodes>
      <ovf :QoSLinkConstraints>
        <ovf :MaxLinkDelay>
          <ovf :Rate>214748364</ovf :Rate>
        </ovf :MaxLinkDelay>
        <ovf :LinkAvailability>
          <ovf :Rate>214748364</ovf :Rate>
        </ovf :LinkAvailability>
        <ovf :LinkReliability>
          <ovf :Rate>214748364</ovf :Rate>
        </ovf :LinkReliability>
        <ovf :LinkCapacity>
          <ovf :Rate>214748364</ovf :Rate>
        </ovf :LinkCapacity>
      </ovf :QoSLinkConstraints>
    </ovf :VirtualLinkSection>
```

In progress results

Virtual Node Section

```
<ovf :VirtualNodeSection ovf :VirtualNodeName="VN1">
    <ovf :NodeType>Edge</ovf :NodeType>
    <ovf :VNIC>   <ovf :VNIC>
    <ovf :VirtualLinkList>     </ovf :VirtualLinkList>
    <ovf :QoSNodeConstraints>
        <ovf :NodeAvailability>
            <ovf :Parameter>Parameter0</ovf :Parameter>
            <ovf :Value>214748364</ovf :Value>
        </ovf :NodeAvailability>
        <ovf :NodeReliability>
            <ovf :Parameter>Parameter1</ovf :Parameter>
            <ovf :Value>214748364</ovf :Value>
        </ovf :NodeReliability>
        <ovf :NodeCapacity>
            <ovf :Parameter>Parameter2</ovf :Parameter>
            <ovf :Value>214748364</ovf :Value>
        </ovf :NodeCapacity>
        <ovf :NodeDelay>
            <ovf :Parameter>Parameter3</ovf :Parameter>
            <ovf :Value>214748364</ovf :Value>
        </ovf :NodeDelay>
    </ovf :QoSNodeConstraints>
</ovf :VirtualNodeSection>
```

In progress results

Virtual Net Constraints Section

```
<ovf :QoSVirtualNetConstraints>
      <ovf :NetAvailability>
         <ovf :FonctionName>FonctionName0</ovf :FonctionName>
         <ovf :Rate>214748364</ovf :Rate>
       </ovf :NetAvailability>
       <ovf :NetReliability>
          <ovf :FonctionName>FonctionName1</ovf :FonctionName>
          <ovf :Rate>214748364</ovf :Rate>
       </ovf :NetReliability>
       <ovf :NetCapacity>
          <ovf :FonctionName>FonctionName2</ovf :FonctionName>
          <ovf :Rate>214748364</ovf :Rate>
       </ovf :NetCapacity>
       <ovf :NetDelay>
          <ovf :FonctionName>FonctionName3</ovf :FonctionName>
          <ovf :Rate>214748364</ovf :Rate>
       </ovf :NetDelay>
    </ovf :QoSVirtualNetConstraints>
 </ovf :VirtualNetworkSection>
</ovf :NetworkSection>
```

Conclusion and Perspectives

- ***Problem***:
  - NaaS model to ensure E2E QoS requirements in Cloud Computing
- ***Contributions***
  - QoS-aware architecture: cloud networking services
  - VN-CoS and flow sensibilities propositions
  - Constraints langage to placement of virtual network
- ***In progress work***
  - NaaS: architectural view (SDN and NFV) ▫
    - VN Mapping: solver 2 (NaaS – IaaS) ▫
    - Configuration control and data planes
    - Virtual network description
  - NaaS : Network services deployment process
  - Performances evaluation (Virtuor)

References

- I. Ayadi, N.Simoni, and G.Diaz. QoS-Aware Component for Cloud Computing ICAS 2013, Lisbonne, Portugal.
- I.Ayadi, N.Simoni, and G. Diaz. NaaS: QoS-aware cloud networking services. In Proceedings 12th IEEE International Symposium on Network Computing and Applications, 22-24 August 2013 Cambridge, MA, USA.
- I.Ayadi, G.Diaz, and N.Simoni. QoS-based Network Virtualization to Future Networks: An approach based on network constraints. In Proceedings 4th IEE International Conference Network Of the Future, 23- 25 October 2013, Pohang, SOUTH COREA.

- Paper submit at CloudNet2014:
- G.Diaz and N.Simoni. Dynamic Deployment process in QoS–aware NaaS architecture.