Seven Rules for Guaranteeing Quality of Service in Multi-Domains Multi-Technologies Networks

Michel Diaz
LAAS – CNRS
Michel.Diaz@laas.fr

NexComm – Athens - June 2010

INTRODUCTION

Best-Effort et QoS

QoS/QoE Requirements

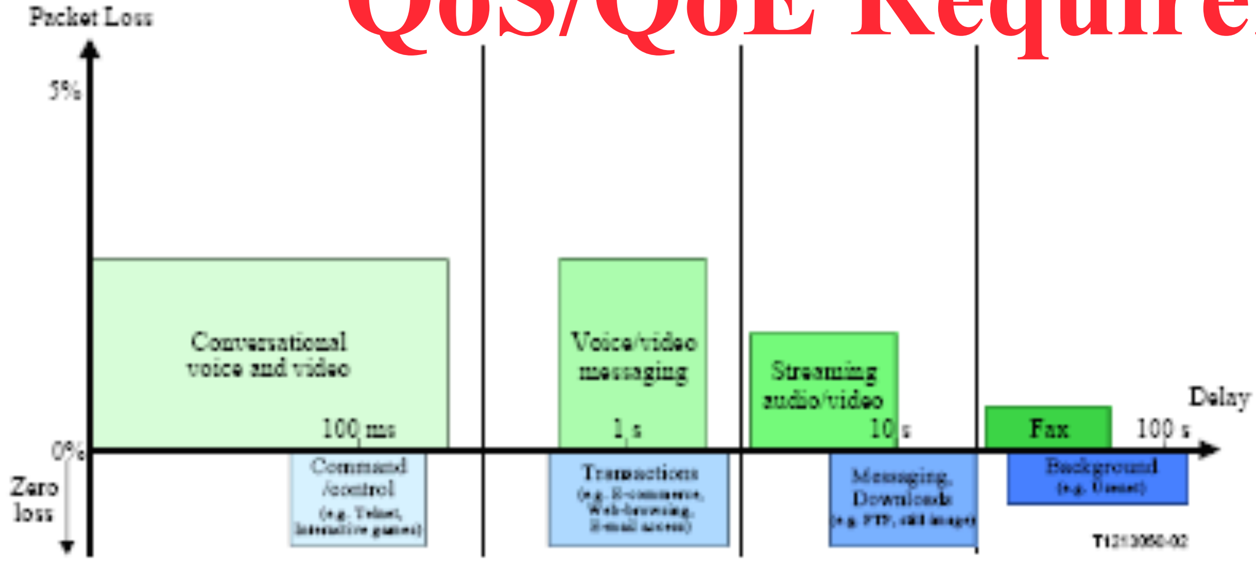


Figure 1/G.1010 – Mapping of user-centric QoS requirements

Applicative QoE and QoS are :

- **-Bandwidth**
- **-Partial or total Reliability**
- **-Temporal Limits**

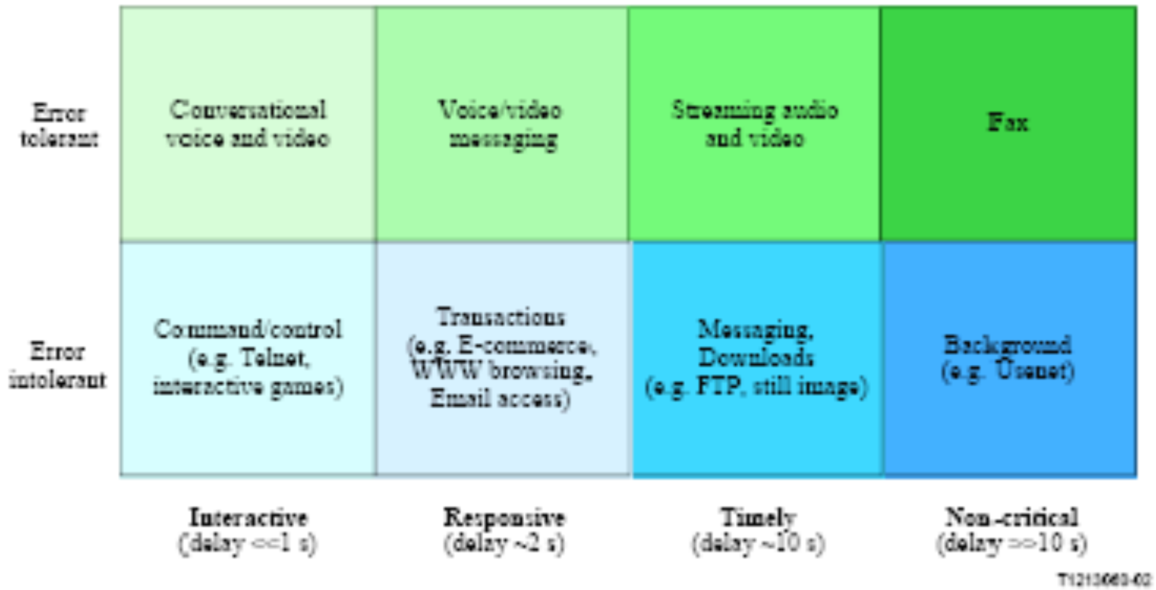


Figure 2/G.1010 – Model for user-centric QoS categories

The global Approaches

- QoS in the Networks

  can be provided :

  – By Over-Provisionning

  – By Optimisation (Best Effort)

  – By Guarantee

From BE to Optimised QoS

- **Starting from the BE Internet**
  - Improve the ressource utilisation

  But by keeping all basic present solutions

  And WITHOUT modifing the architecture

- **Optimise all composants**
  - Adapt les applications (new efficient codecs,…)
  - Adapt les architectures (use proxys,…)
  - Develop better mechanisms (congestion control, …)
  - Develop more adapted protocols (ex in Transport Layer: DCCP,…)

- **BUT ALWAYS Best-effort, WITHOUT guarantee**

From BE to Garanteed QoS

New Requirements :

- **master all ressources of the Internet BUT**
- **BEING as General & Open as the present Internet**

⇒ Define NEW functions,

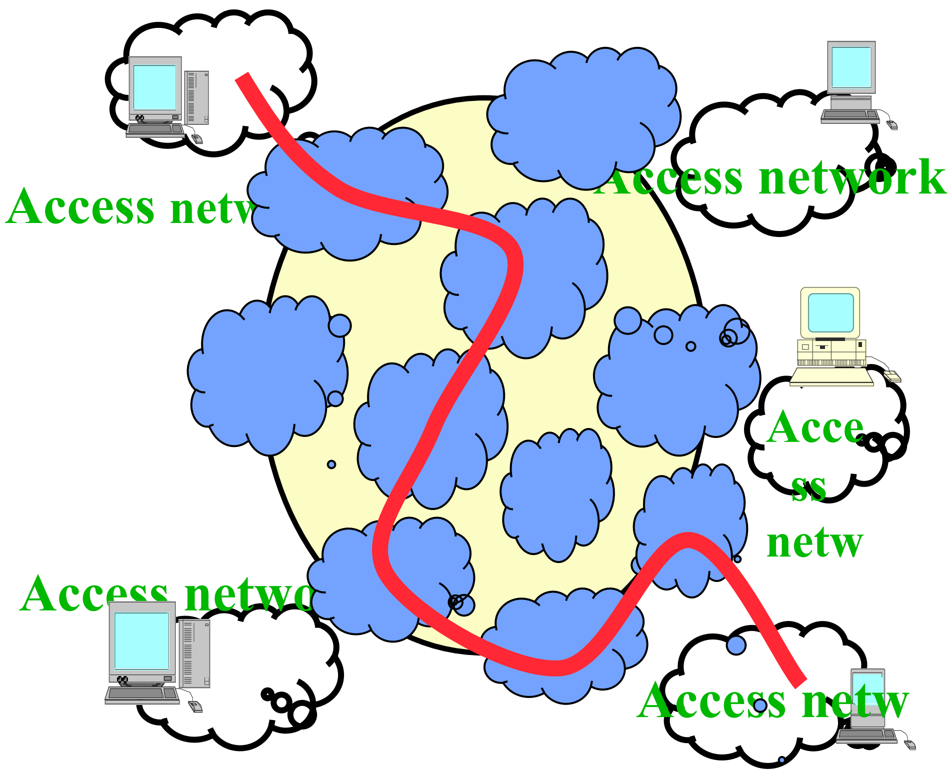
NEW mechanisms and NEW protocols

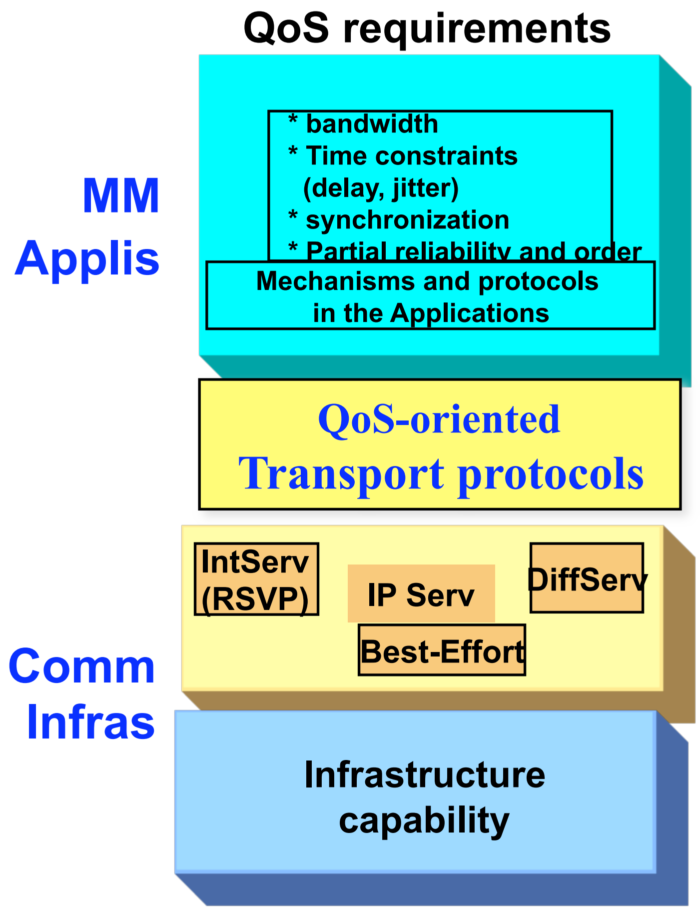
THE problems:

- **What functions, mechanisms, protocols ?**
- **What global architecture ?**
- **What is the resulting complexity ?**
- **What are the resulting deployment difficulties ?**

as this has to be done

Horizontaly accross the networks and Verticaly in the nodes





A lot of work has been done :

- MANY Mechanisms, protocols et architectures
- But all Partial, so

HOW to design and integrate (fully coherently)

from Users to Users

– The most efficent mechanisms
– Their compatible compositions in protocols
– All these protocols (signaling, data, …)

in an adequate architecture ?

Many ideas for this work : Projet IST EuQoS

End-to-end QoS support over heterogeneous networks

Ref:

http://www.euqos.eu/

T. Braun, M. Diaz, J.E. Gabeiras, T. Staub Editors, *End-to-end Quality of Service Over Heterogeneous Networks*, Springer, ISBN 978-3-540-79119-5, 2008

EuQoS Objectives

1. Design End-to-end QoS architecture and software (a QoS system) for

- multiple heterogeneous networks
- multiple heterogeneous technologies

2. Demonstrate the QoS System

for a large set of access networks :

Ethernet, xDSL, WiFi, UMTS, MPLS-GMPLS and Satellite

What Methodology ?

Proposal : Use two design levels

Level 1: A Meta-Level => (Design) Meta-Rules

Level 2 : A Classical Level => (Design) Rules

Design META-RULES

- **Design a global/full architecture**

If Mechanisms are designed independently of their context, there is a low probability for them to be good choices

- **Avoid all end-to-end homogeneous solutions**

Given the Internet complexities, topologies and technologies, any acceptable architecture must integrate DIVERSITY

- **Define Only Key Interfaces, Independent of the Technologies**

Give total freedom to designers, in all domains and technos, to develop the most efficient solutions (perf, cost, etc)

The seven

Design Rules

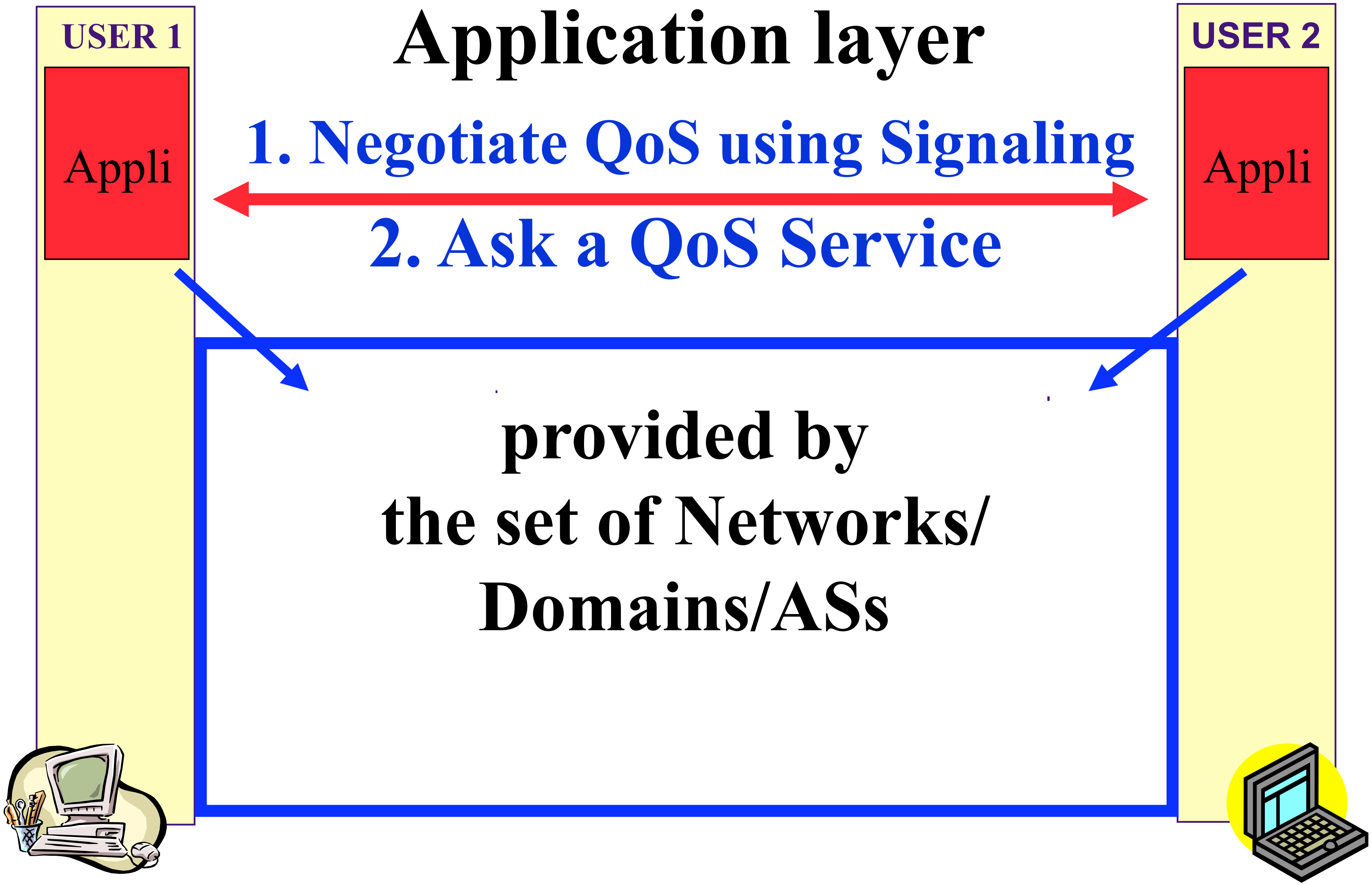
Rule 1 :

APPLICATIONS
must be independent of
NETWORKS

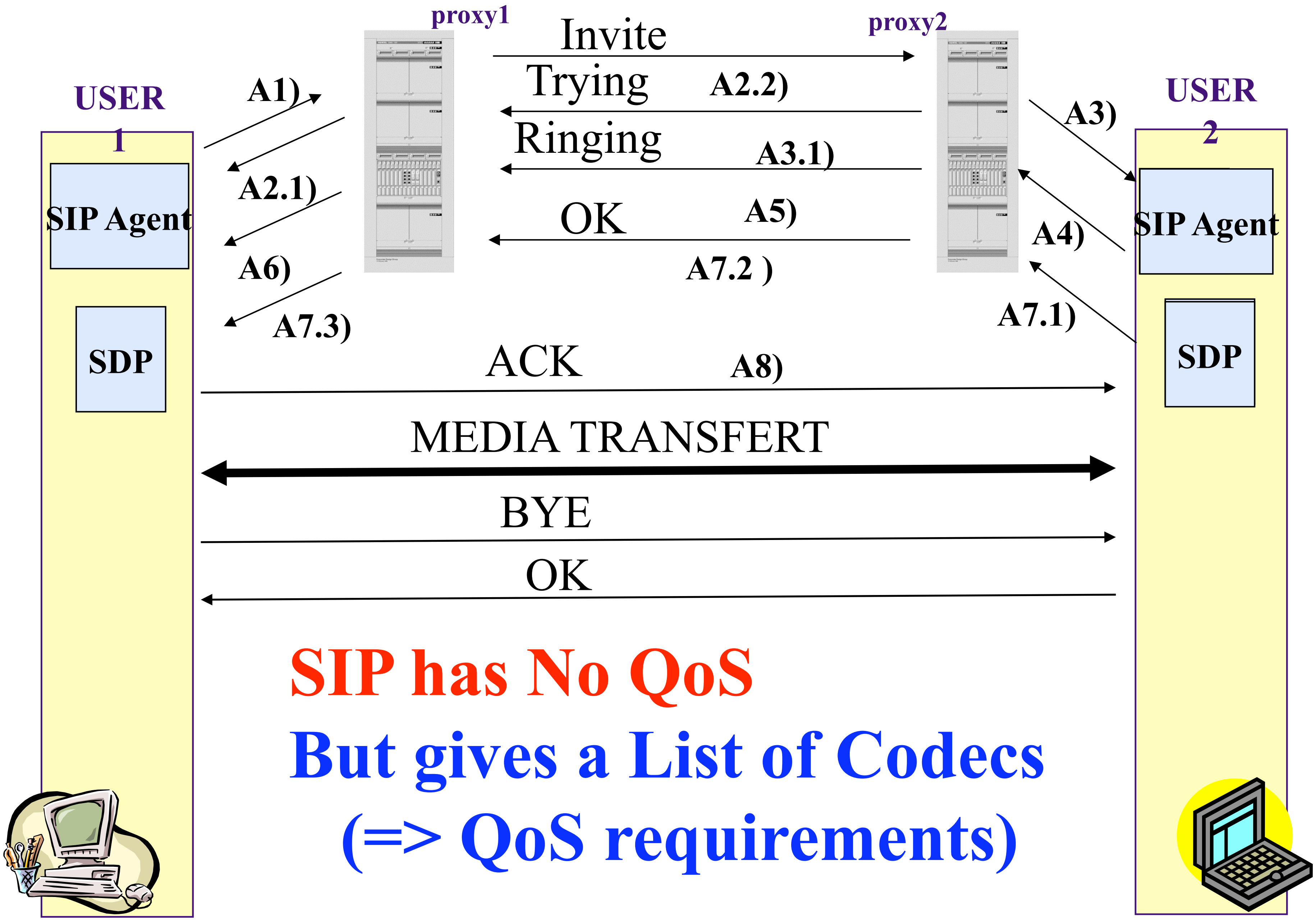
↓

ARCHITECTURE v1

ARCHITECTURE v1



Note: many applications use SIP



SIP has No QoS

But gives a List of Codecs (=> QoS requirements)

Negociate QoS By

Encapsulating SIP (EuQoS EQ-SIP)

- SIP Codec negotiation becomes a building block
- SIP "Selected Codec" used to request QoS

  for the end-to-end connection:

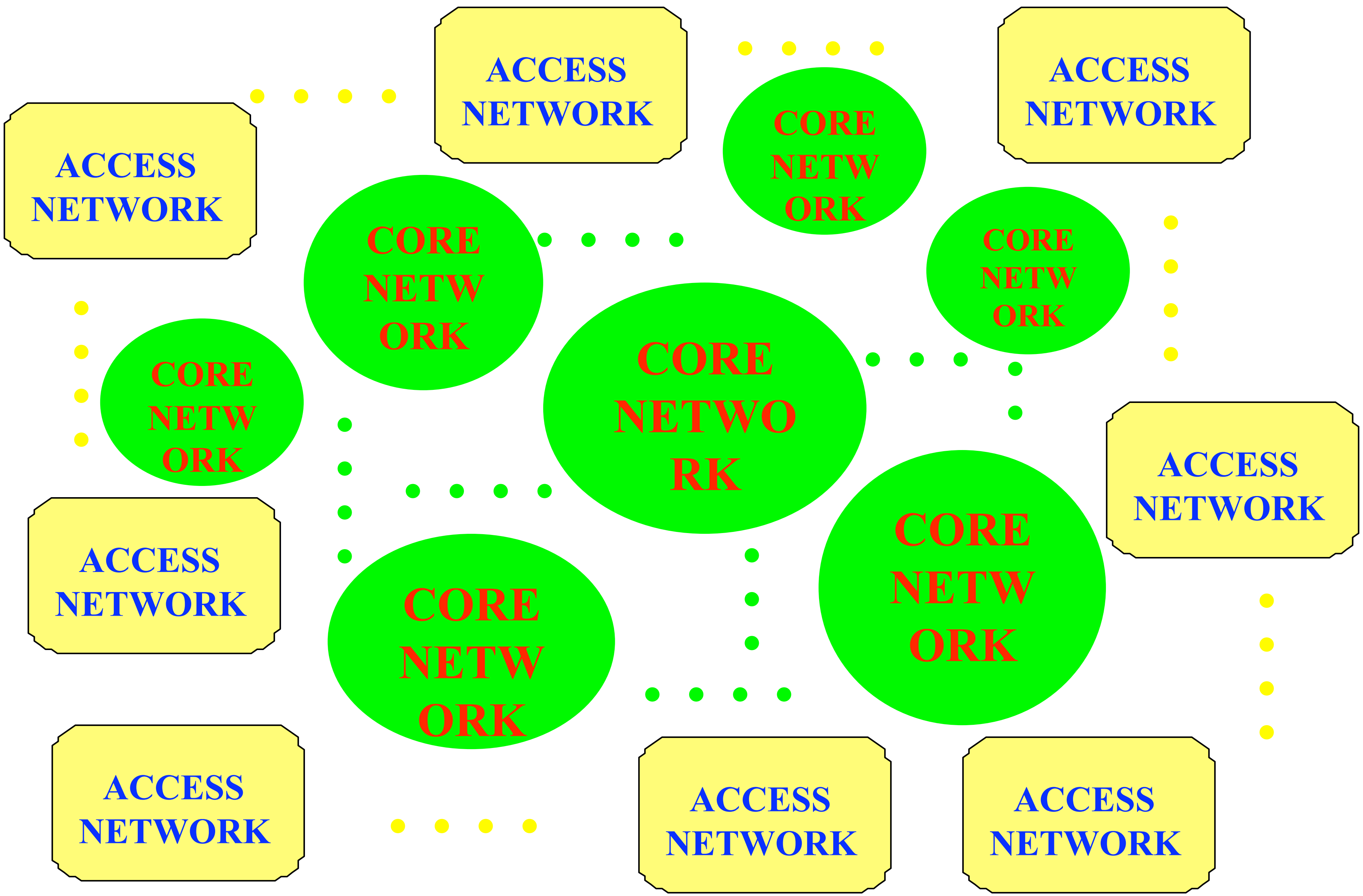
  - EX: From Calling to Called User, Bandwidth Request for "Selected Codec" is 5 Mb/s,…

Rule 2 :

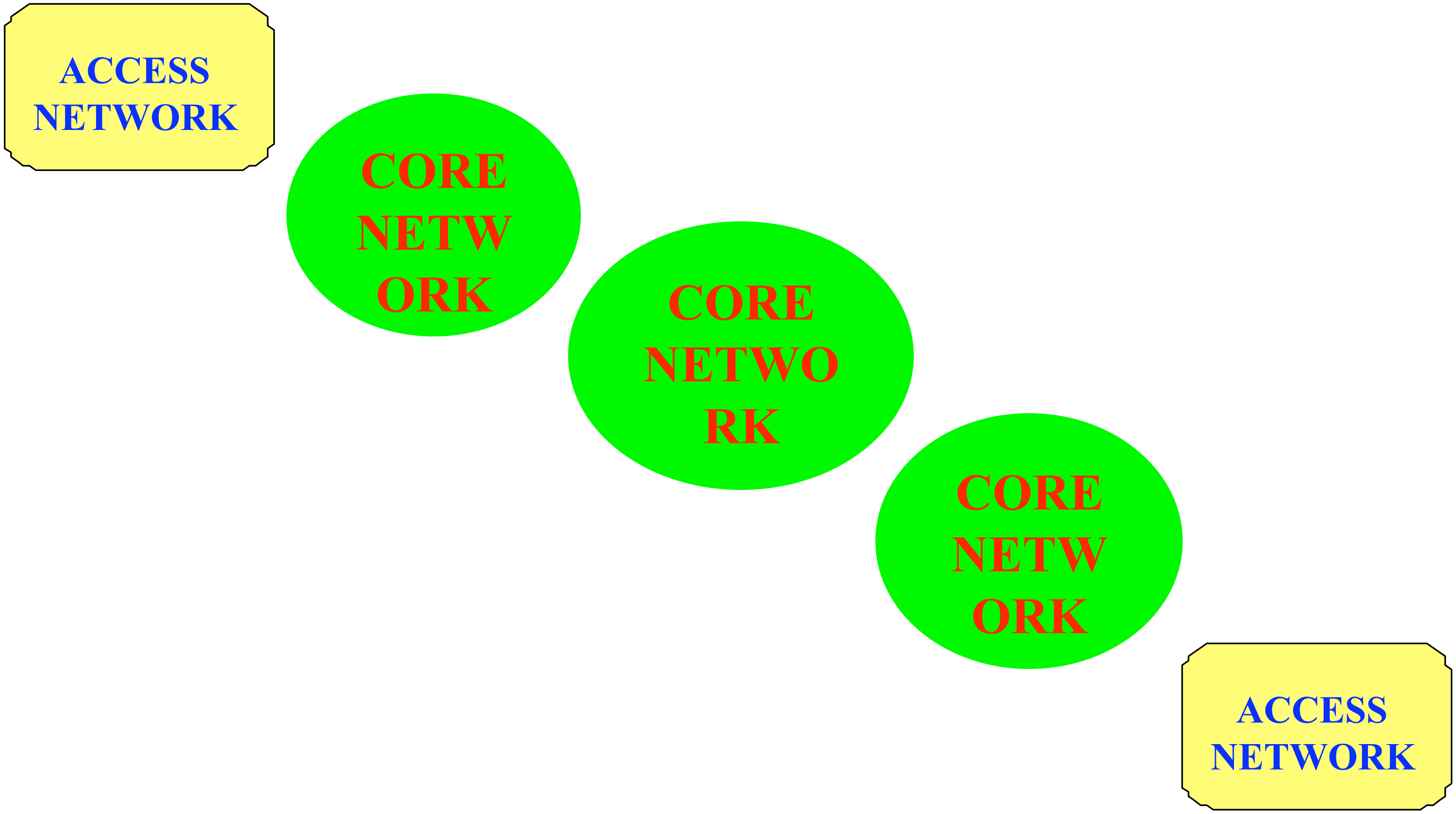
INTRODUCE
Technology Independence
with respect to NETWORKS

ARCHITECTURE v2

Over GEANT & the NRNs



The Simplest case

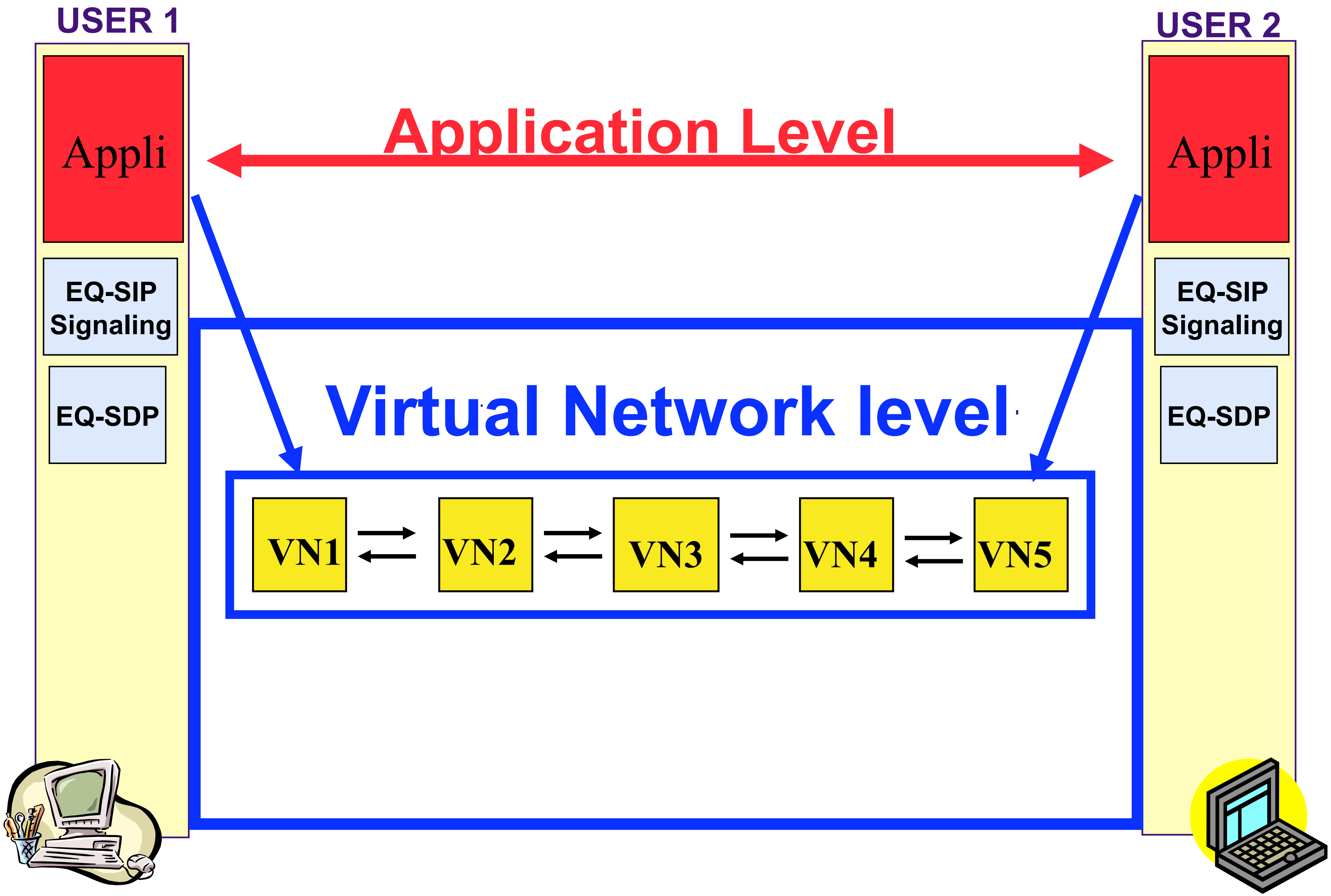


Then, Independence means :

a) The different **technologies must not appear** in the architecture: make it **virtual**

b) **Define and Add a technology independent virtual network (VN) level** for guaranteeing an **abstract** end-to-end QoS

ARCHITECTURE v2



For QoS and Ressources, TWO PLANES

1. **PLANE 1 - CONTROL, how to RESERVE the resources: SIGNALING to be defined to enforce the guarantee**

2. **PLANE 2 - DATA, how to USE the resources : how to send the DATA when the resources are allocated**

PART 1

CONTROL PLANE

SIGNALING

Rule 3 :

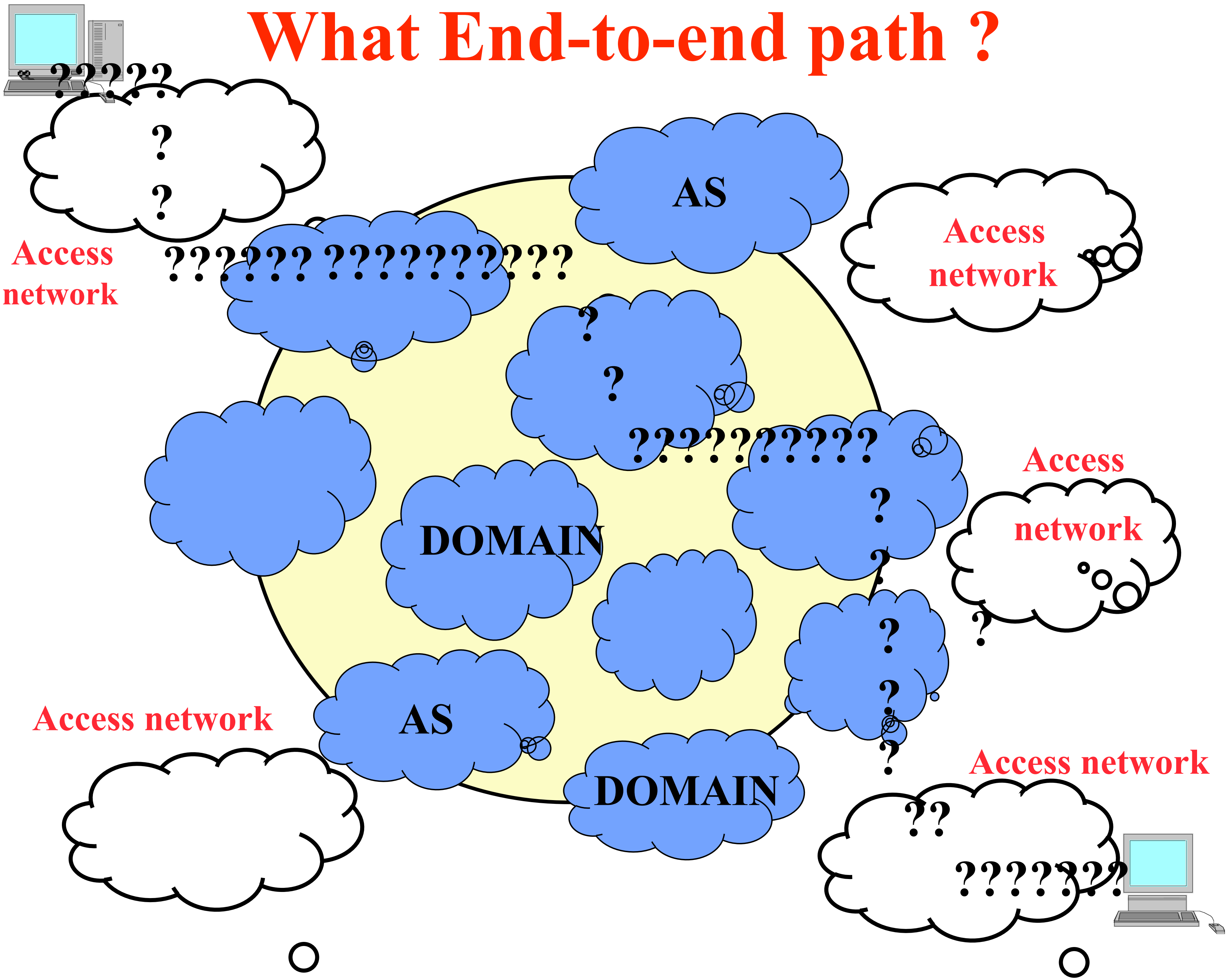
Define (in the CONTROL Plane)

as a High-Level SIGNALING

beween the VNs

ARCHITECTURE v3

What End-to-end path ?



SIGNALING and DATA Paths

a) Classical solution : in the Control Plane

- Define a SIGNALING path (e.g. using RSVP)
- Use SIGNALING path to send the DATA
  and the Data follows this QoS Data path

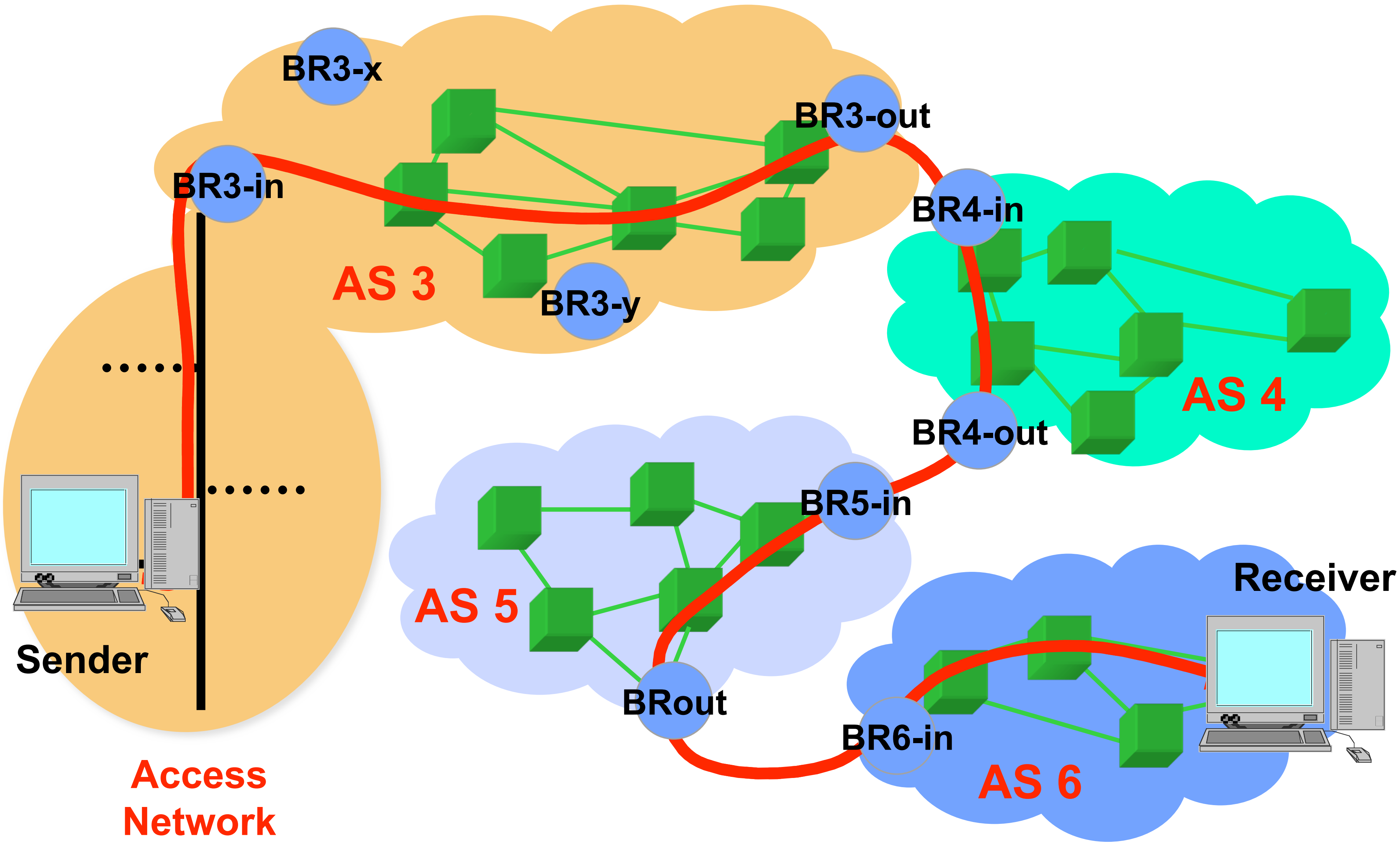
  =>> deployment ???

b) Hortogonal solution : in the Data Plane

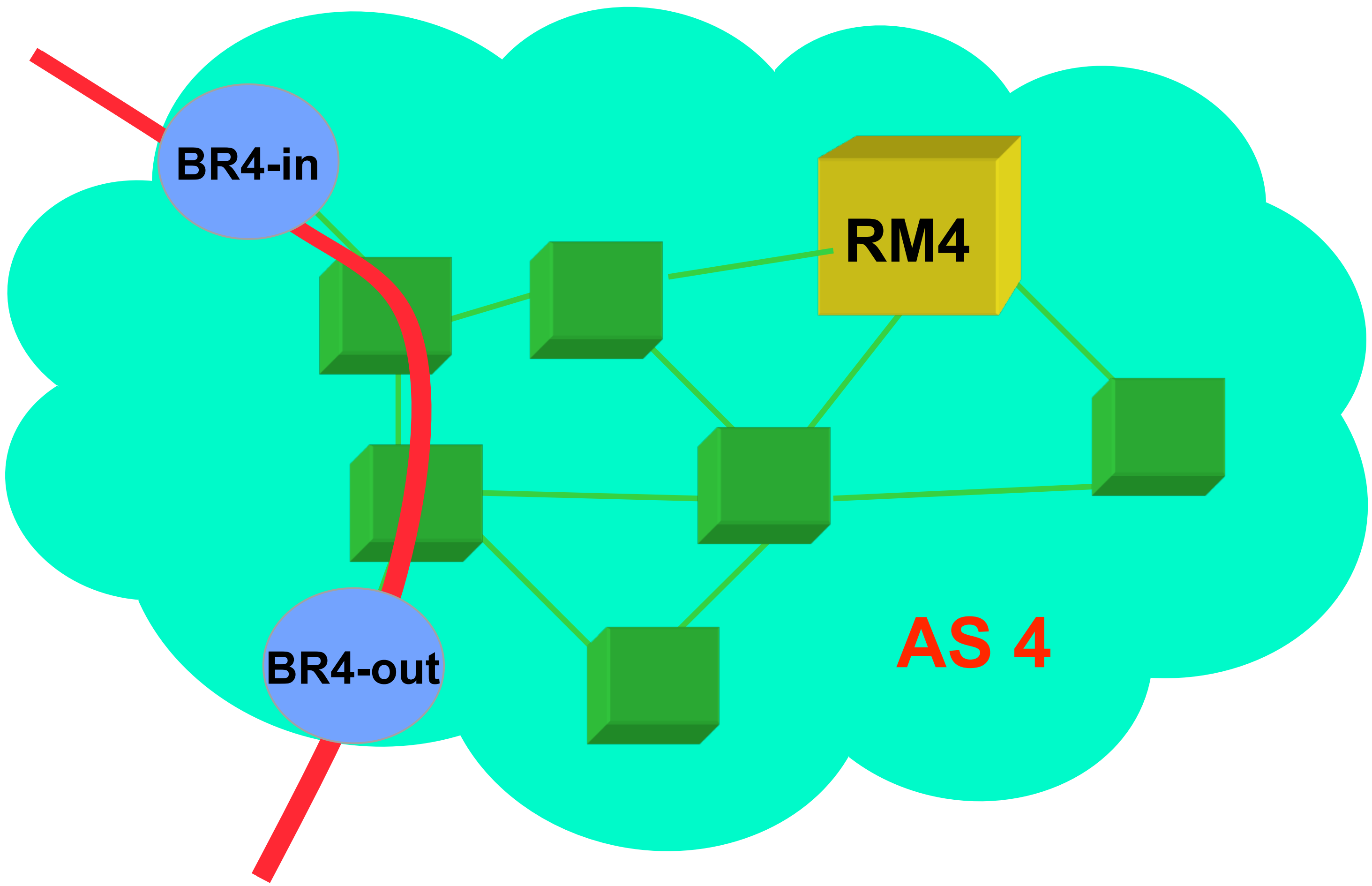
- Use the present DATA path : BGP or BGP-based
- Reserve the resources when needed (Res Managers)
- Using a SIGNALING path derived from DATA path
  and Send Data along this QoS Data path

Start from Data Path : BGP-based

(e.g. q-BGP, with QoS classes)

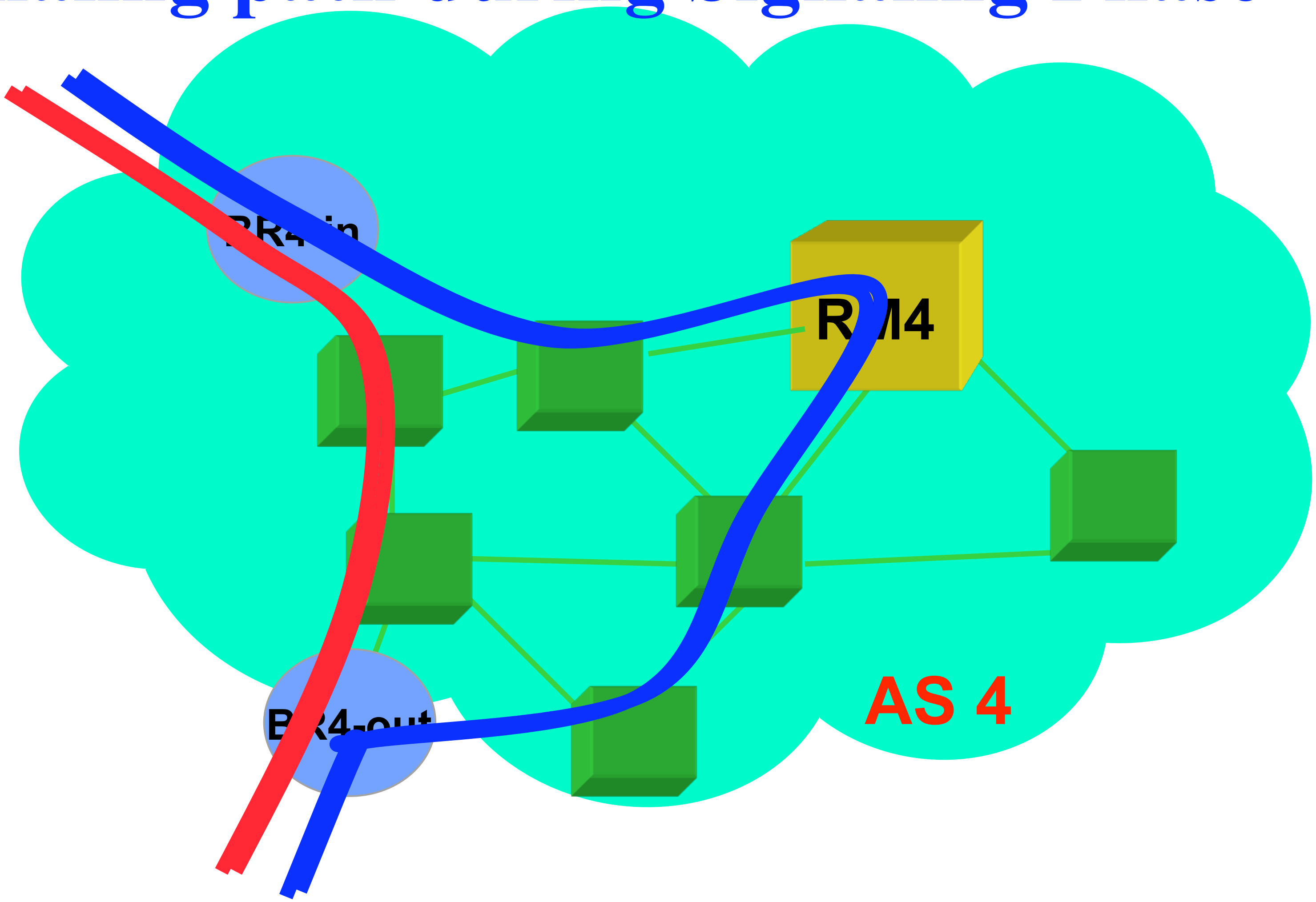


Add a Resource Manager per domain



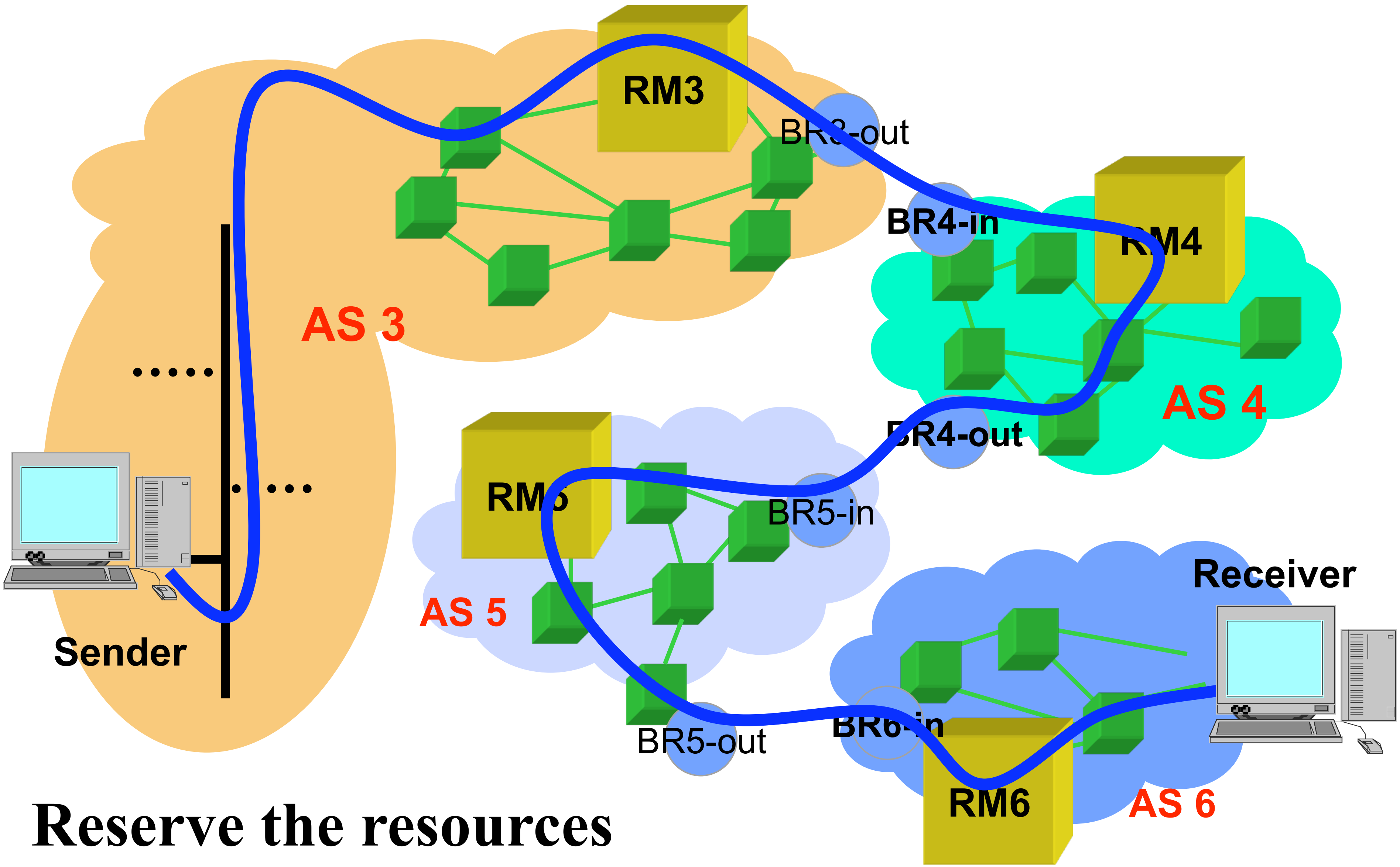
So, Use :

Signaling path during Signaling Phase



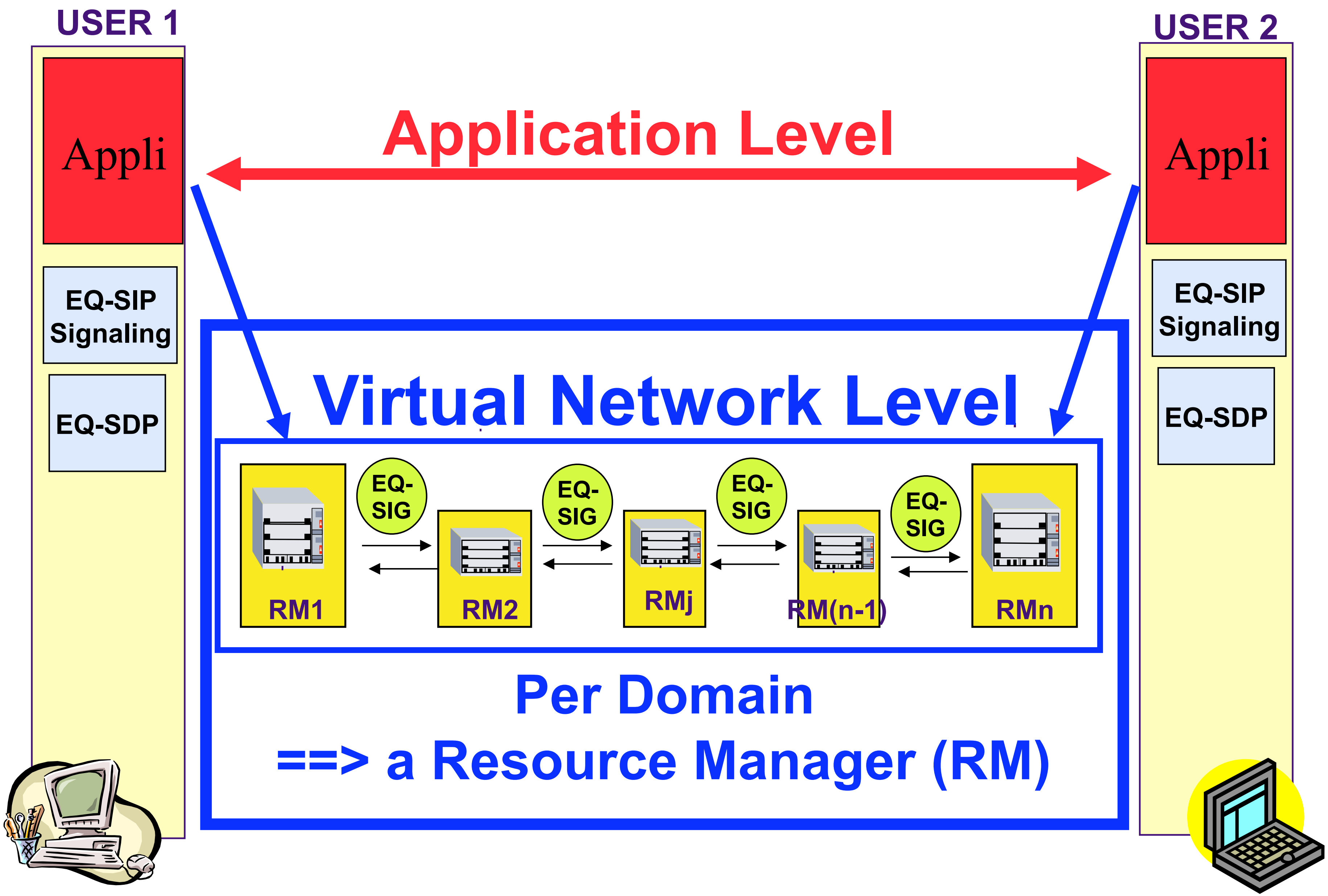
and Data path during Data Phase

PHASE 1 : SIGNALING



Reserve the resources

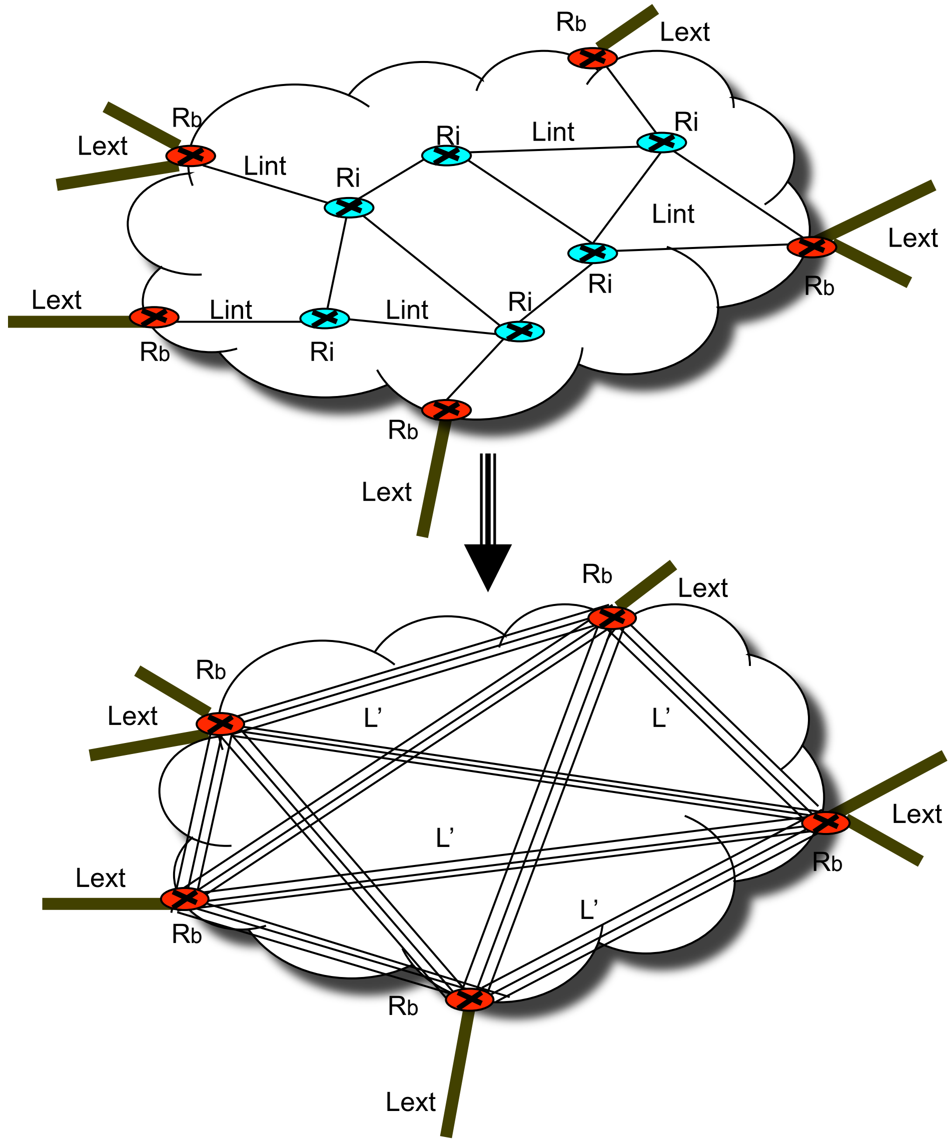
ARCHITECTURE v3



Ex of Admission Control Model : a domain is seen by a RM as pairs of BRs (Abstraction)

Topology T

Model M1



Model M2

Abstract Topology Ta

With the Abstraction Properties

Model 2 : Abstract topology $Ta = \{Rb, Lv\}$

with Lv = {virtual-real links between border routers}

$Ta = Projection (T / \{R_i, L_i\})$

(abstracting internal routers and internal links)

- ***P* being a Property on T and Ta**

Projection must be such that :

***P* (Ta) => *P* (T) <=> if *P* true in Ta, then *P* true in T**

Note : *P*/(Ta) can be not optimal in the sense that some *P* can hold in T and not in Ta

- **Ex of Properties : delay, bandwidth, etc…**

Rule 4 :

Integrate the technologies of the networks

↓

ARCHITECTURE v4

ADD

a)A technology dependent function

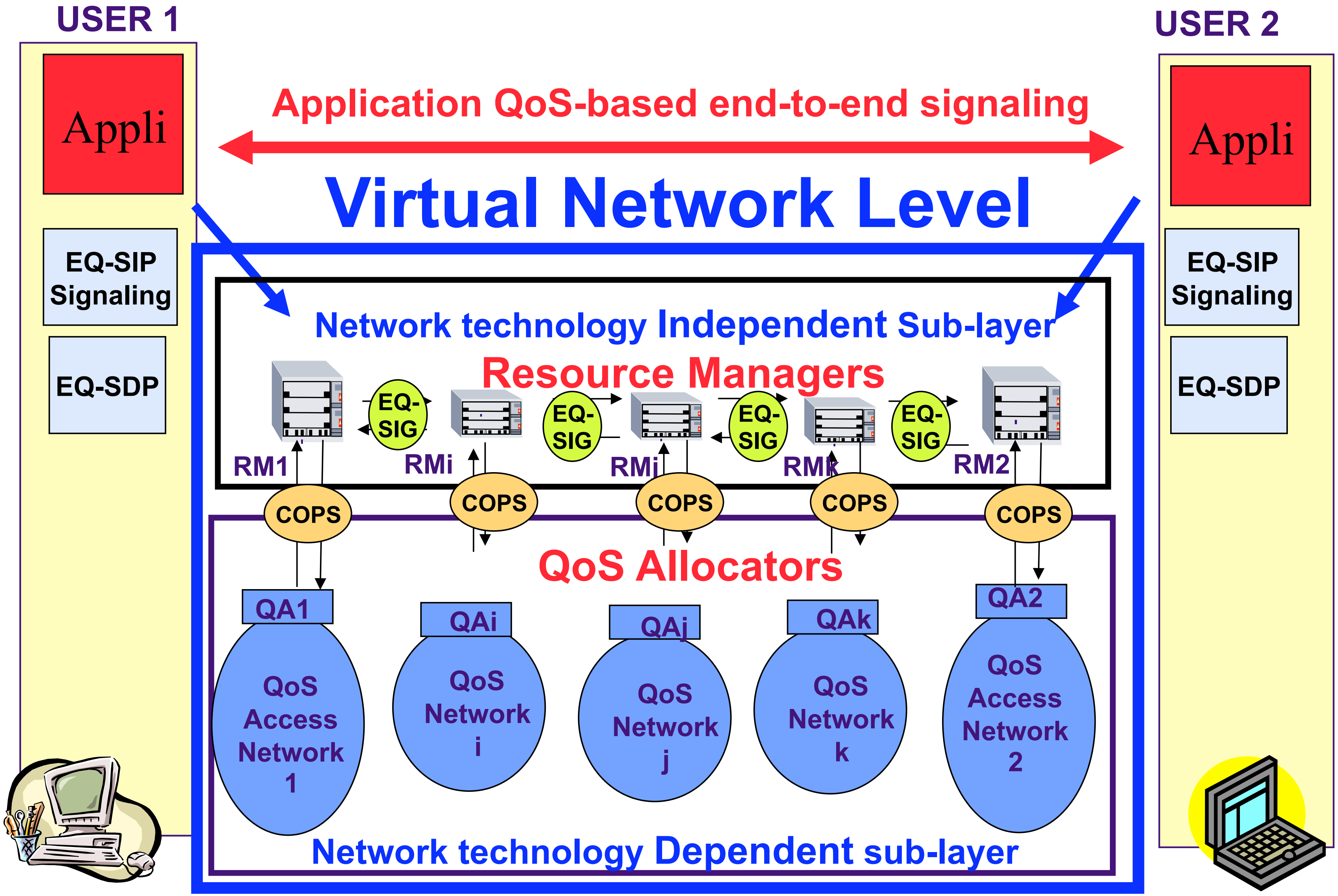
=> A local QoS allocator QA

=> Defined by a
technology-dependent approach

b)and then for each technology

independently
define an optimal mapping
between the RM and the QA

ARCHITECTURE v4



Rule 5 :
Define a simple Connection Admission Control

↓

Architecture v5

Connection Admission Control

When a user asks a QoS communication

Check the RMs

for the availability of the resources

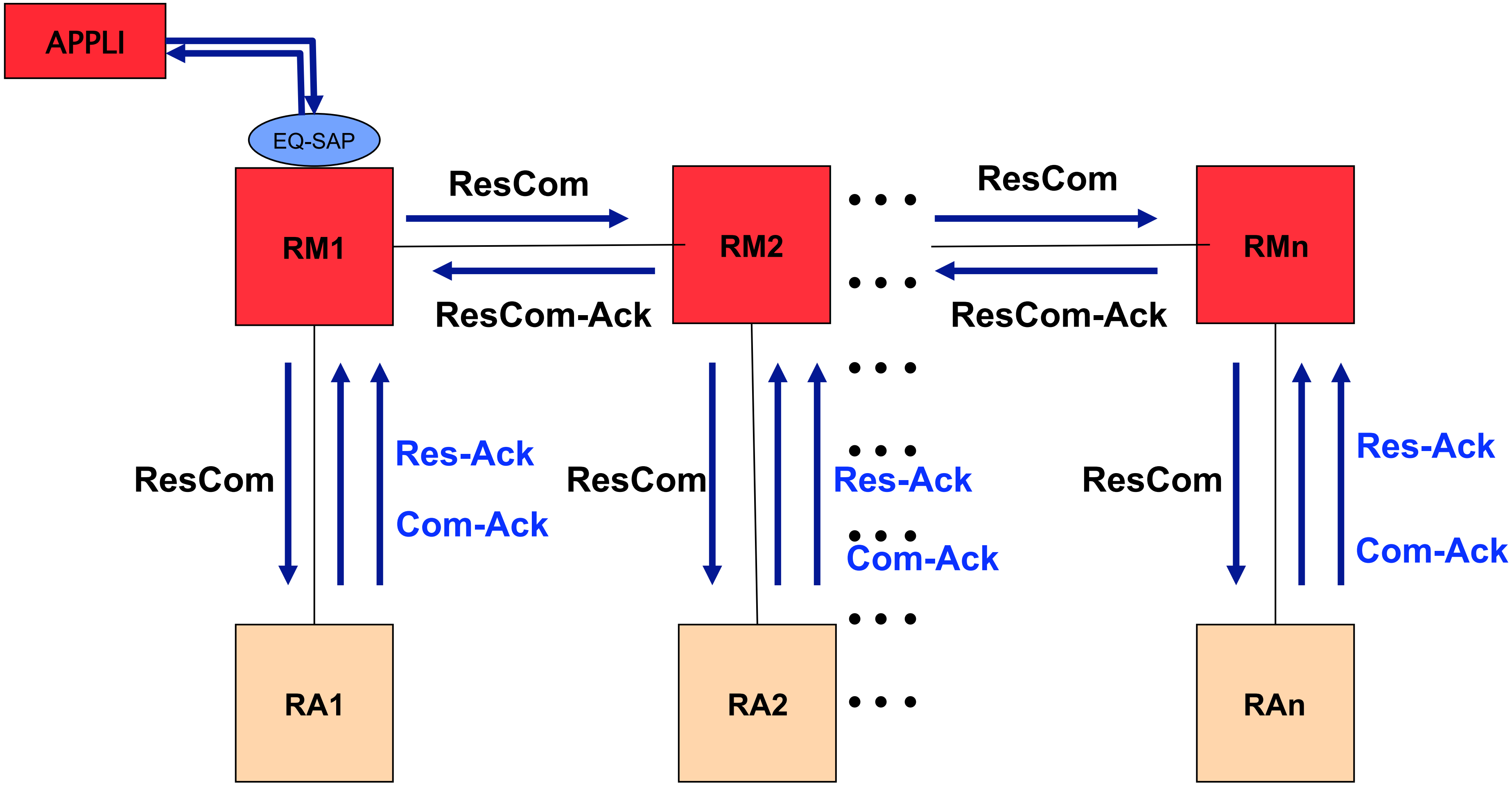
along the path

a)If OK for one RM, ask the next one

b)If not OK, do not accept the call

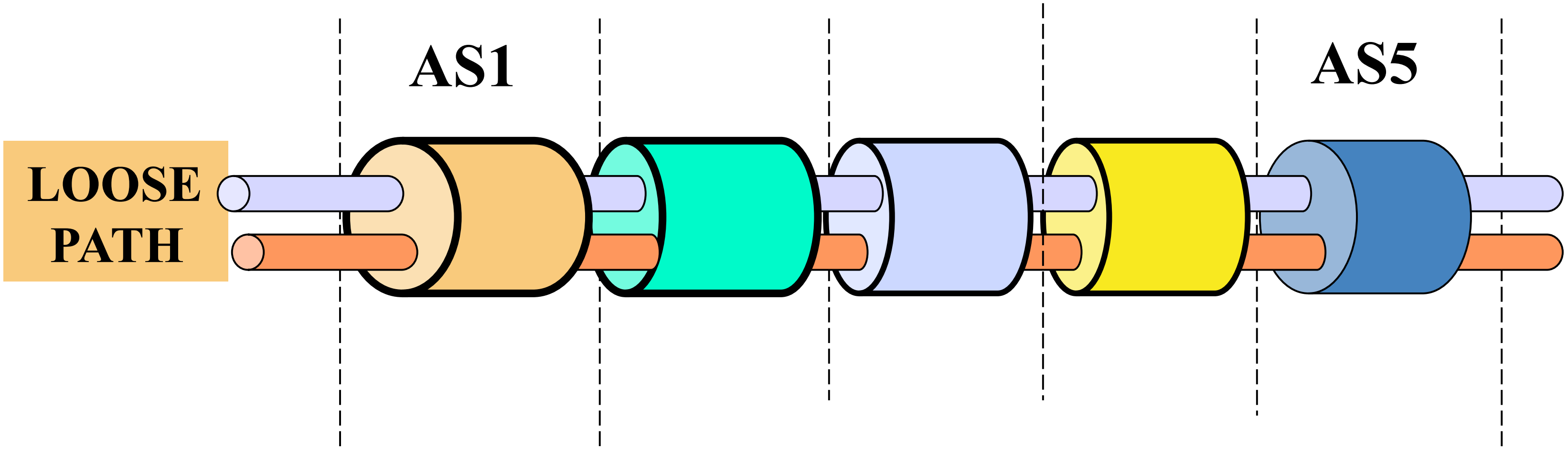
stop or come back and propose less QoS

Ex of CAC in Invocation process



ResCom: Reserve and Commit Message Request
ResCom-Ack : Reserve and Commit Message Ack
Res-Ack : Reserve Message Ack ; Com-Ack : Commit Message Ack

The end-to-end path Signalling

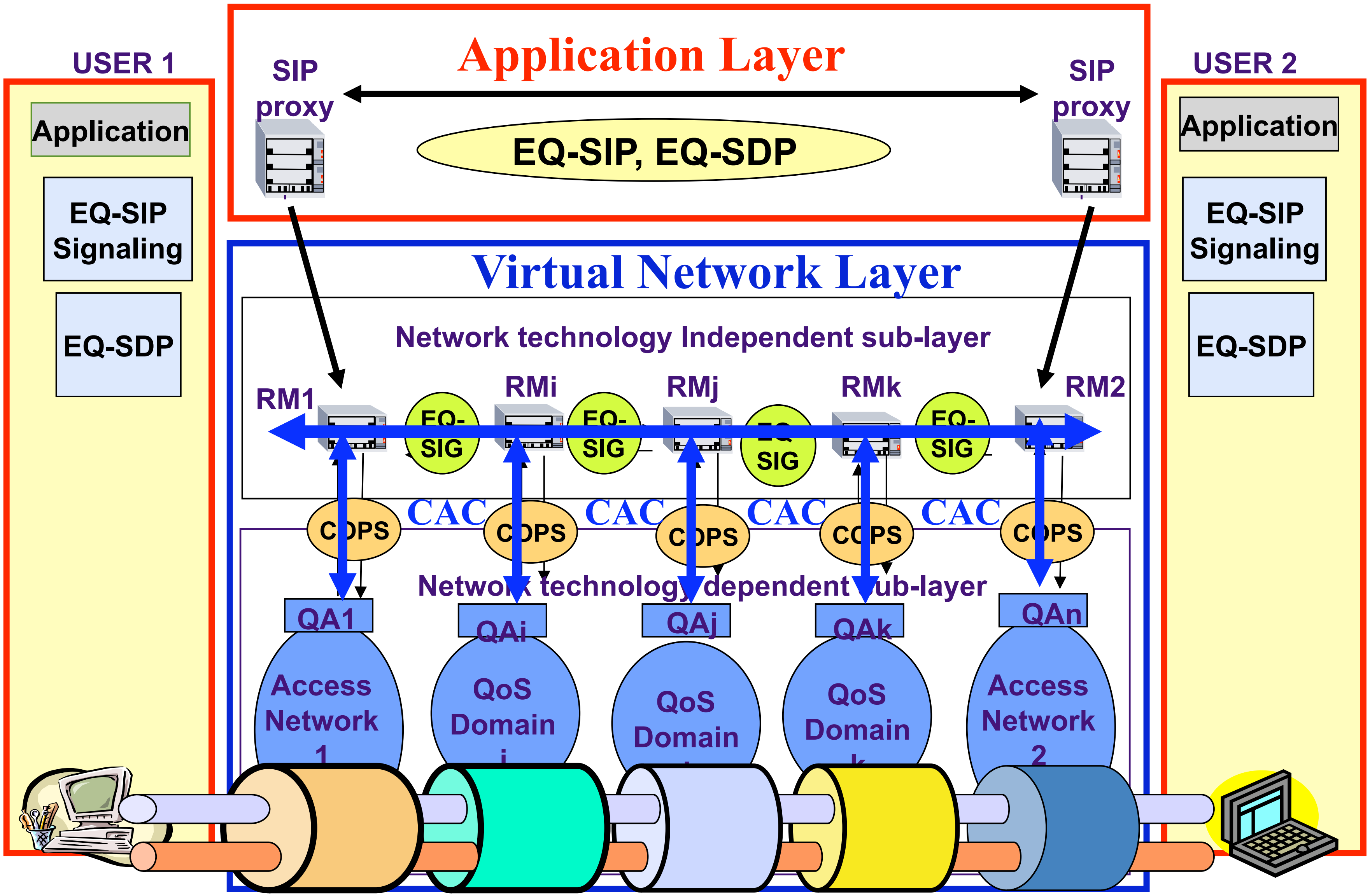


The BGP-based is a LOOSE Path :

Resources are checked in each domain by the RM and its QA (when QA needed)

=> Potential problem of SCALABILITY

ARCHITECTURE v5



Rule 6

Use Recursivity for SIGNALING Scalability and Deployment

Architecture v6

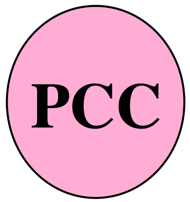
Use Recursivity by Super-Domains

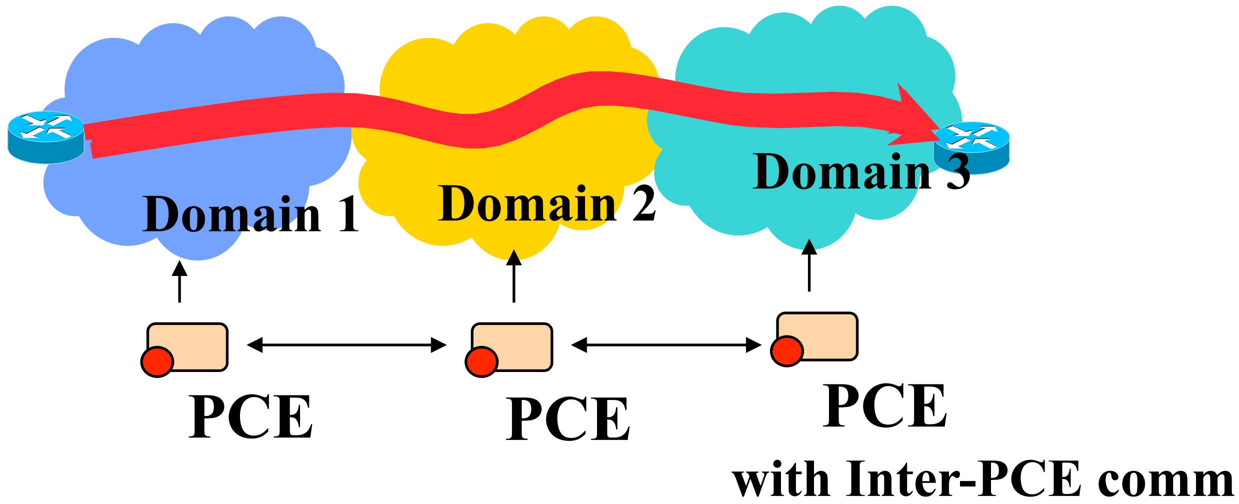
- To SOLVE the scalability problem
  in the CORE networks,
- USE Super-Domains:
  AGGREGATIONS of SETS of DOMAINS
- Example: AGGREGATION using MPLS:

  => MPLS tunnels and

  => the IETF PCE (Path Computation Elements)

PCE Architecture



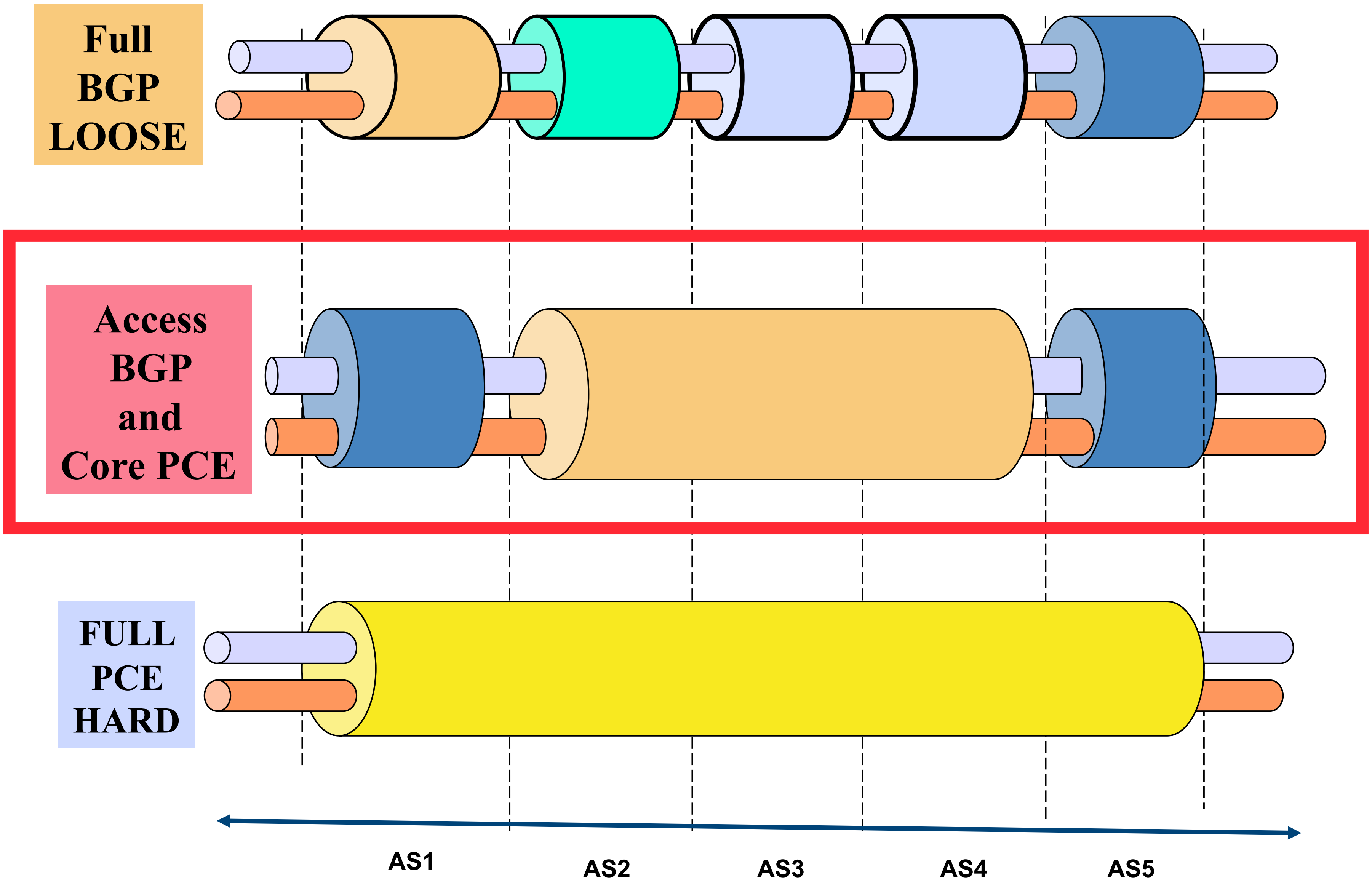


- **PCE defines**
  - **PCC (Path Computation Client) :**

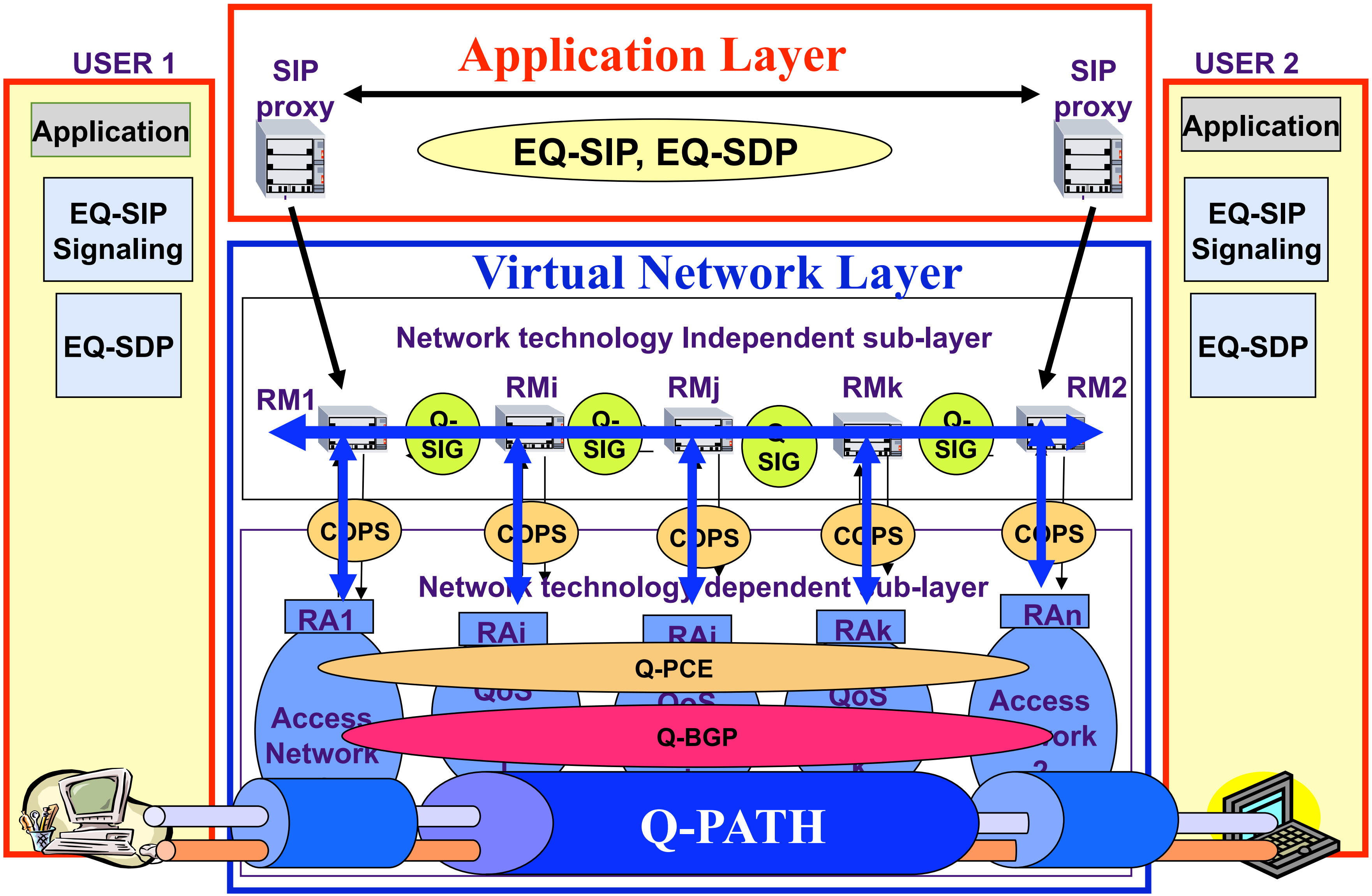
    requests Multi-Domains MPLS path computation, to be performed by a PCE
  - **PCEs (Path Computation Elements) :**

    the entities that can compute the path in function of the network graph & its constraints

Family of E2E Paths



ARCHITECTURE v6



PART 2

DATA Transfer

Rule 7 :

Integrate the DATA PLANE

Architecture v7

Two layers have to be defined:

a) Network Layer :

Network Classes of Services (CoSs)

b) Transport Layer :

sending DATA depending on each CoSs

by EQ-ETP, a New Multi-Services Multi-Services Transport Layer Protocol

EuQoS: 3 network Classes of Services

Classes de Service	Garantie	
RT	Bande passante Maximum	
NRT	Bande passante Minimum = g	
BE	Pas de valeur garantie	

EuQoS EQ-ETP

Application profile / Network Classes of Service	**Streams** *Error tolerant e.g. VoD*	**Non-Streams** *Error intolerant e.g. file transf*
RT	**ETP = UDP[RC]**	**ETP[EC]**
NRT	**ETP[gTFRC]**	**ETP[gTFRC+EC]**
BE	**ETP[TFRC+DT]**	**ETP[TFRC+DT+EC]**

CONCLUSION

– Generic Architecture to guarantee QoS
in Multi-domains & Multi-technologies

– Scalable

– Able to integrate all technologies

- First BGP and MPLS, and then, as
- NRN GEANT PIP (Premium IP) in the core

– In EuQoS Prototype implemented with
6 applications, 11 testbeds, 6 technologies