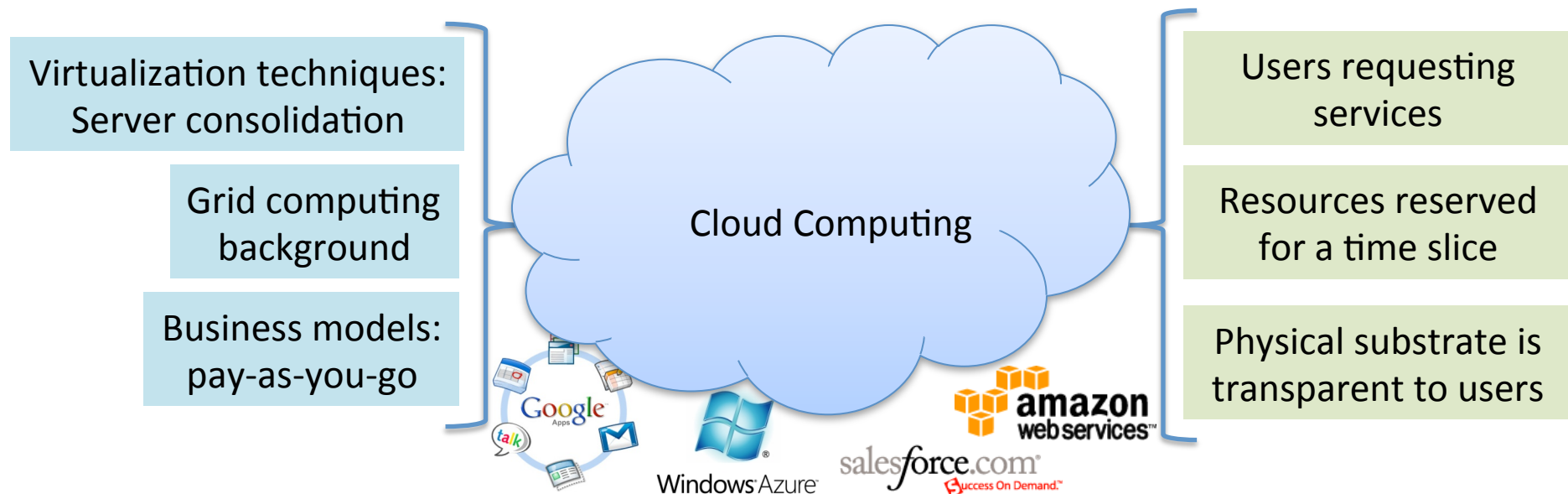


Results on Virtualization

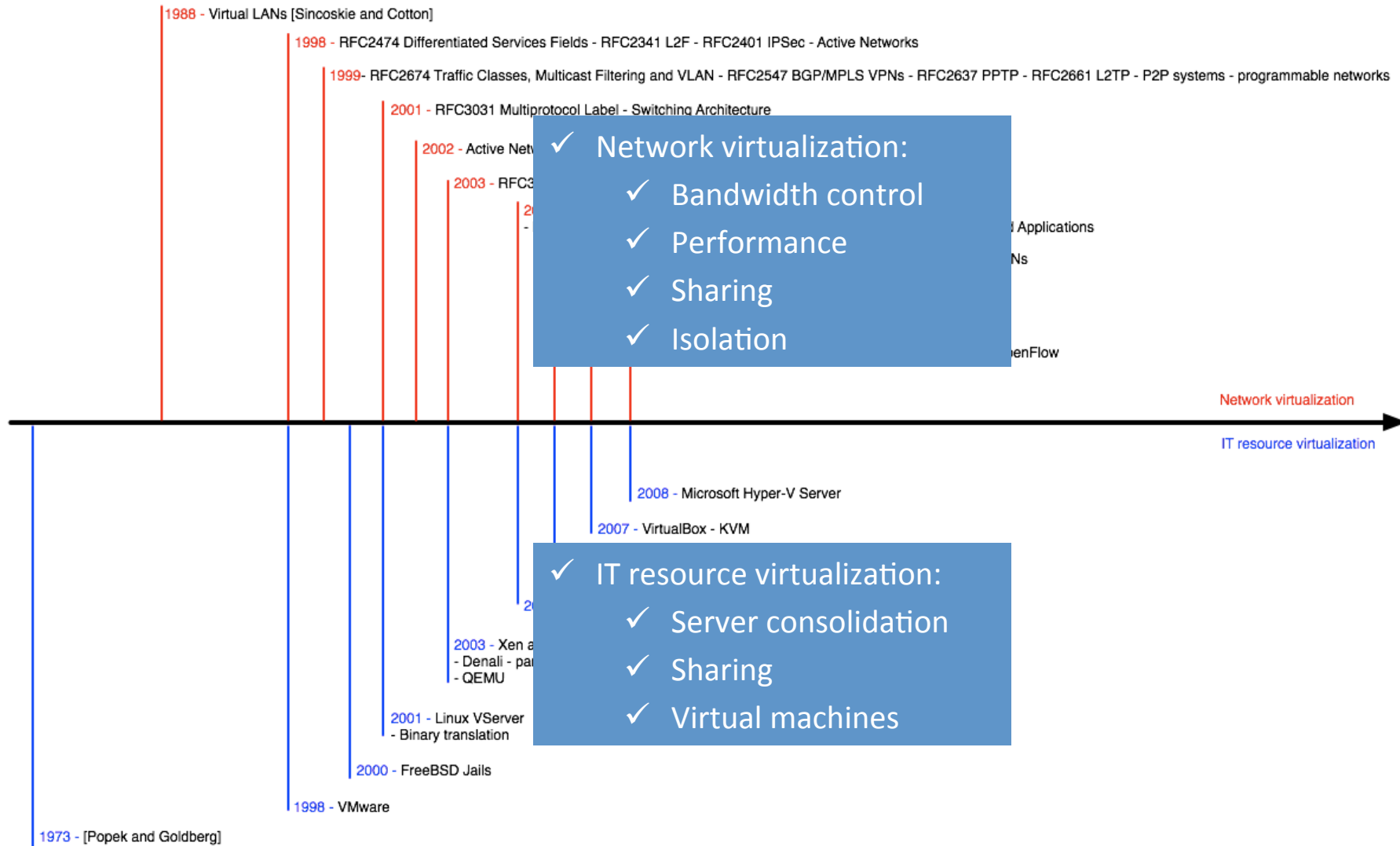


Context: Cloud Computing

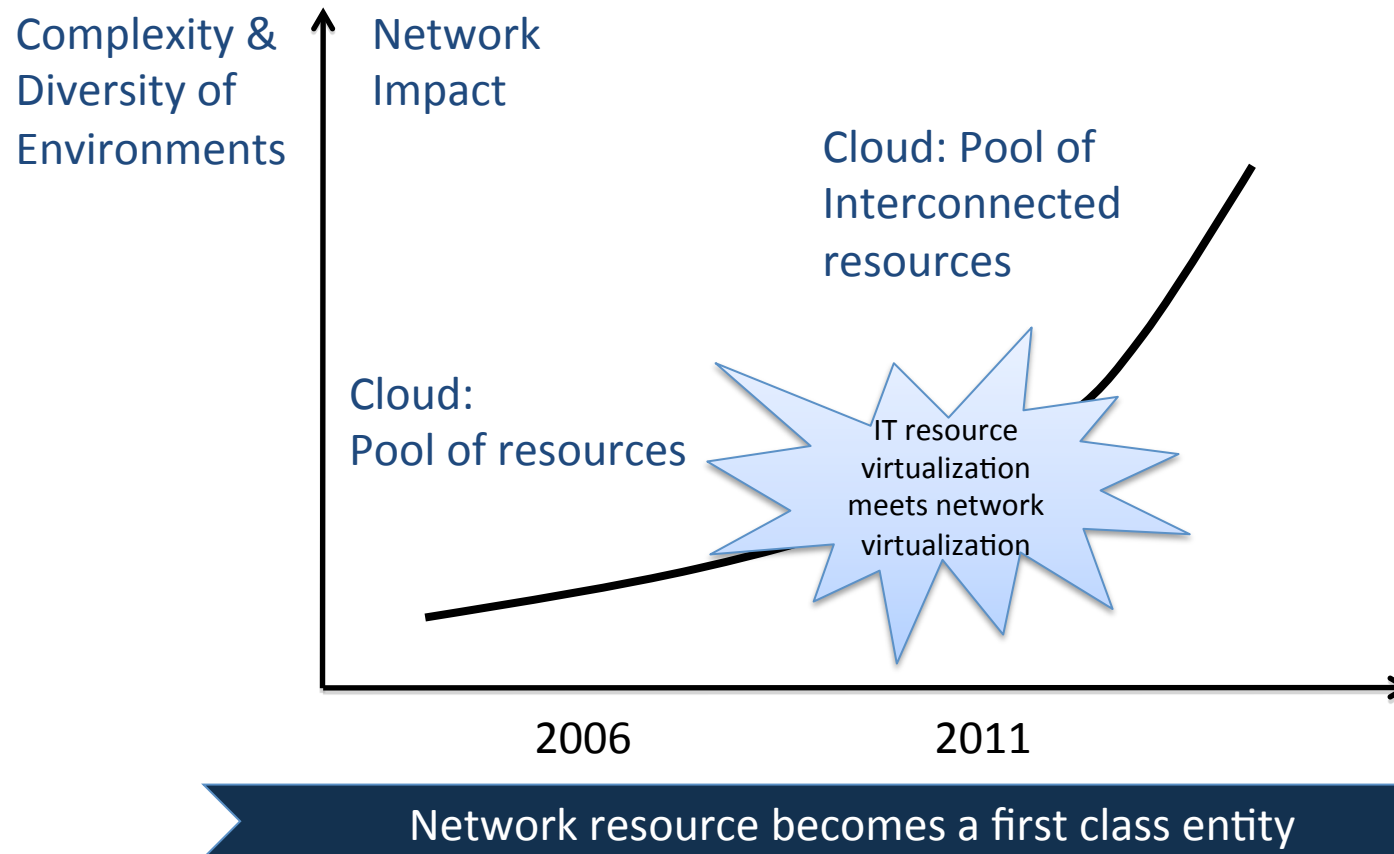
Cloud Computing: provides a new means for delivering IT services, based on **cost-efficient**, **scalable** and **on-demand** provisioning guided by the user's requirements. Exploiting the flexibility offered by **virtualization**, IT elements are exposed as a set of **services** (or even resources) that can be **reserved for a specific timeframe** [NIST – Mell and Grance, 2009]



Virtualization: two parallel paths



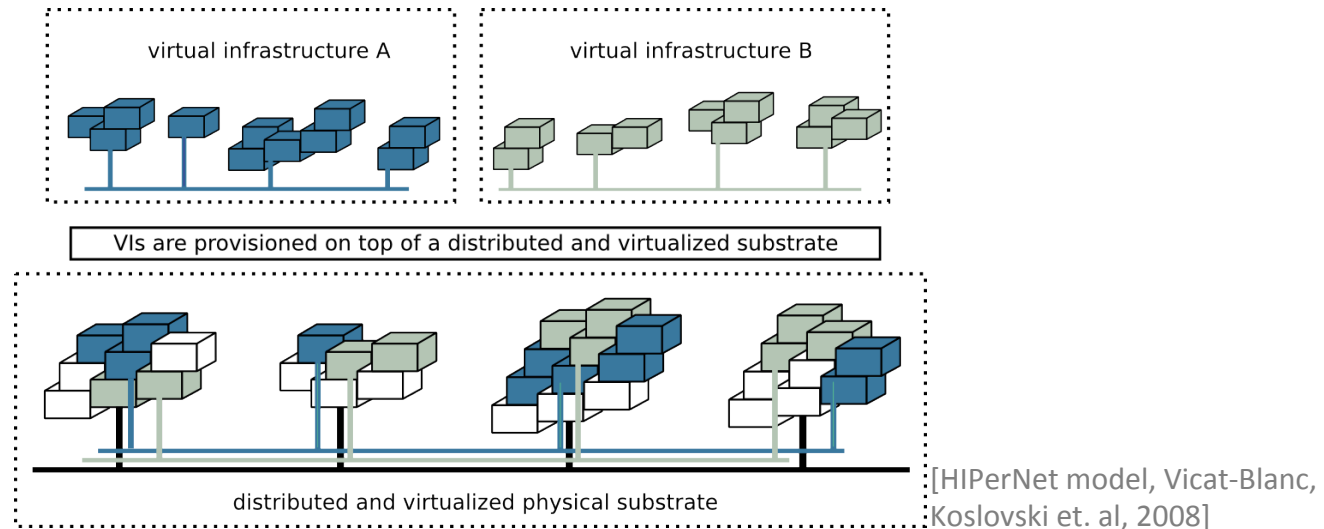
Clouds & Network



[VXDLforum, 2011] [Vicat-Blanc, 2011]

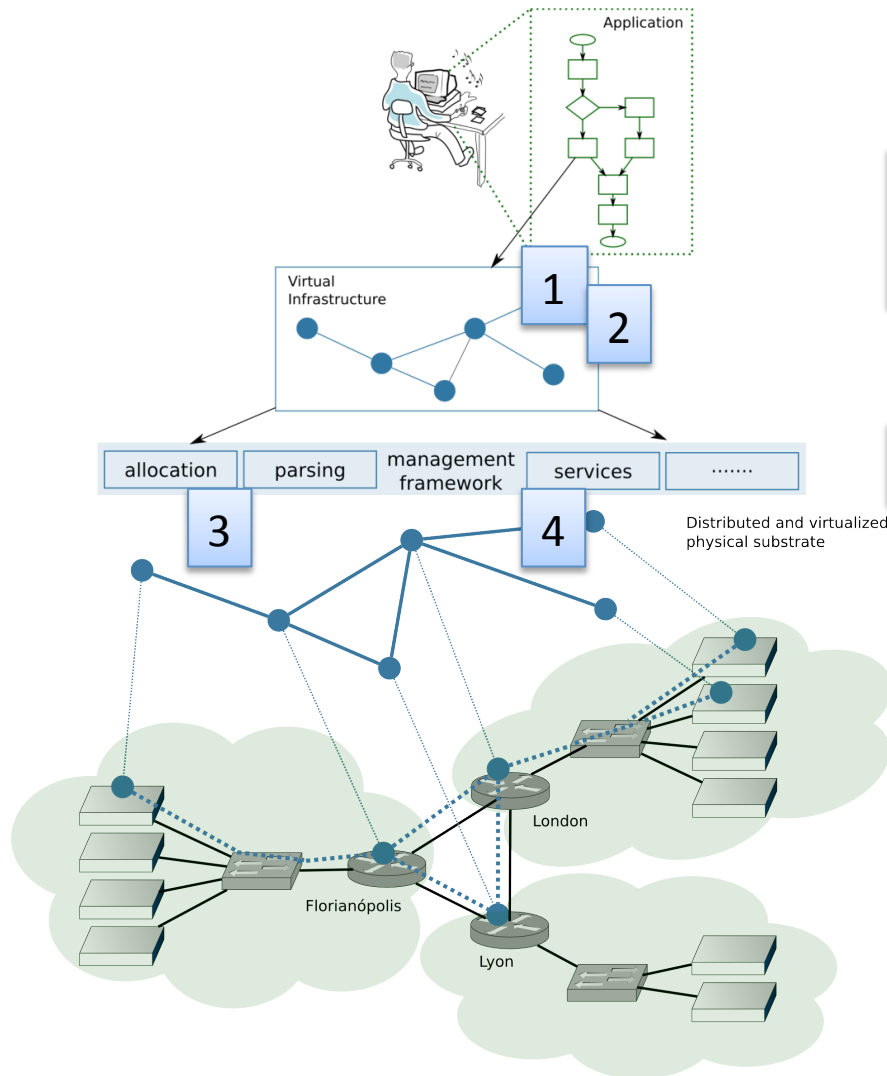
Virtual Infrastructure concept

- Virtual Infrastructures (VIs) extend the original IaaS to provide dynamic virtual networks of computing (also virtualized) resources [Anhalt, Koslovski, Vicat-Blanc, ACM International Journal of Management, 2010]



- Synonyms of VI (or similar definitions):
 - Slice from PlanetLab [L. Peterson: "PlanetLab: Catalyzing Network Innovation", 2007]
 - VPC from Amazon [<http://aws.amazon.com/>]
 - VPXI from HIPerNet (ANR HIPCAL) [Vicat-Blanc et al., 2005]
 - SRV from Carriocas [Audouin et al., 2009]
 - FNS from SAIL project [<http://www.sail-project.eu/>]
 - VI from GEYSERS project [<http://www.geysers.eu/>]

Contributions:



- 1) Virtual Infrastructure Description Language (VXDL)
- 2) Mechanisms to translate the application's requirements into VXDL
- 3) A heuristic for allocating resources to VIs
- 4) A mechanism for provisioning reliable Vis (Reliability as a service)
- 5) Virtual switching fabric for CQ switches
- 6) Malleable requests bandwidth sharing approach
- 7) Routing games over time

VXDL: a language for describing VIs

- ✦ A language is the enabler for user and infrastructure provider interaction (SLAs)

- ✦ **VXDL**

- Model IT and network resources -> VI
- General description, resources, network, and timeline
- Adopted by several communities (academic and business)

- ✦ **VXDL forum**

- An open forum for discussions and improvements of VXDL
- www.vxdlforum.org

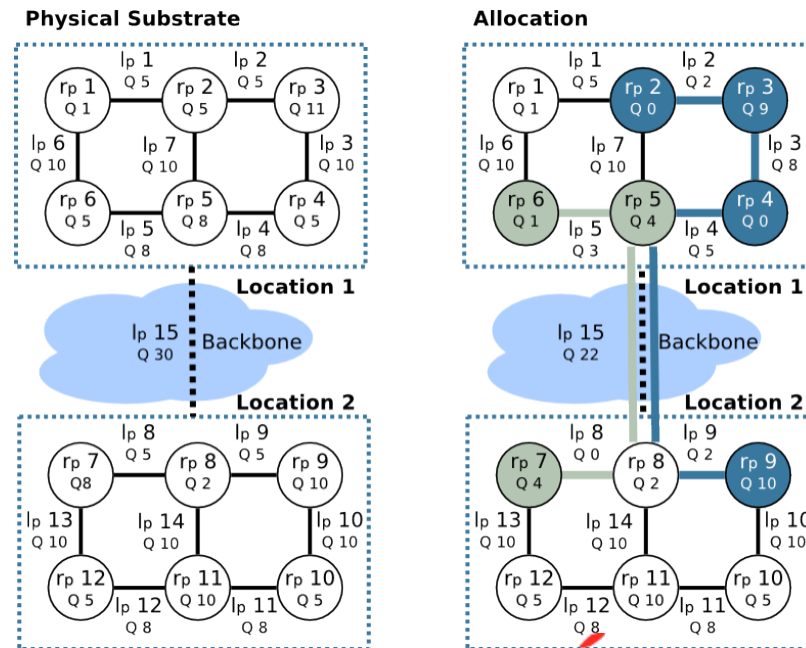
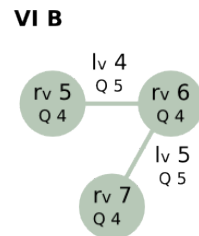
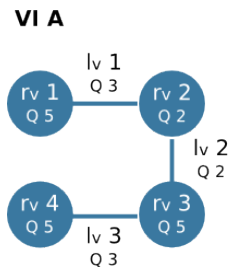


DOCOMO USA Labs



Allocation of VIs

- ✦ **Sharing/Performances:** Users with *different objectives and expectations* can rent *dynamic* VIs
- ✦ **Decoupling:** VIs can be allocated anywhere on top of a *virtualized substrate*
- ✦ **NP-hard problem** [Chowdhury and Boutaba, 2009]

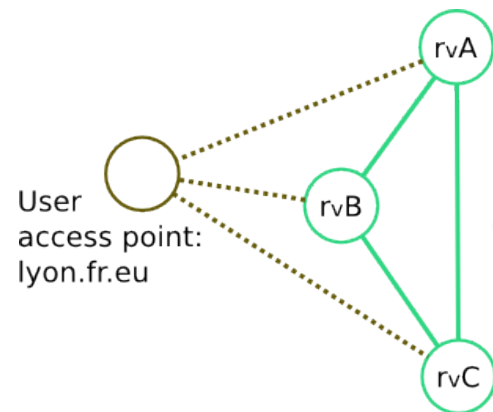


Insight: locating & allocating

Locating VI considering users and InPs perspectives: [Koslovski et. al, IEEE/IFIP International Symposium on Integrated Network Management, 2011], [Koslovski, Huu, et al., Int. Conf. on Cloud Computing 2009], [Koslovski, Huu, et al., Journal of Grid Computing (JoGC), 2010]

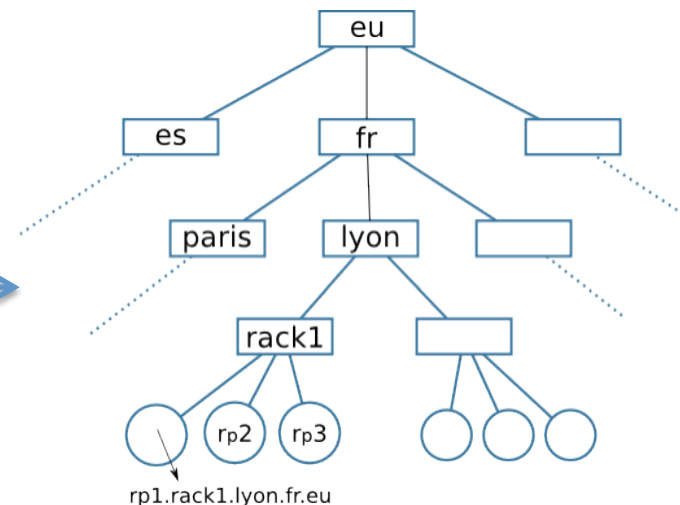
- **InP**: substrate fragmentation
- **Users**: allocation quality (= distance between the user and its resources, plus between the resources themselves)

Virtual Infrastructure request



must be allocated

Virtualized substrate



Components hierarchically distributed and interconnected

Problem formulation

- Physical substrate **fragmentation**: minimize the ratio between activated and available resources

$$\text{minimize: } \frac{\#F_R + \#F_L}{\#R^p + \#L^p}$$

- Allocation quality**: minimize the average distance (in hops) among all resources and a reference point

$$\text{minimize: } \sum_{r_i \in R^v} \frac{D_R(A_R(\mathcal{M}_R(r_i)), a^u)}{\#R^v}$$

- Minimize the **allocation cost** considering IT and network resources

$$\text{minimize: } \sum_{G_i^v \in S^v} C_{VI}(G_i^v, T) \quad C_{VI}(G^v, T) = \alpha C_R(R^v, T) + \beta C_L(L^v, T)$$

$$C_R(R^v, T) = \underbrace{\int_0^T \left(\sum_{r_i \in R^v} C_r(r_i, t) \right) dt}_{\text{IT resources}}$$

$$C_L(L^v, T) = \underbrace{\int_0^T \left(\sum_{l_i \in L^v} C_l(l_i, t) \times \overbrace{\text{len}(\mathcal{M}_L(l_i))}^{\text{Path length}} \right) dt}_{\text{Links}}$$

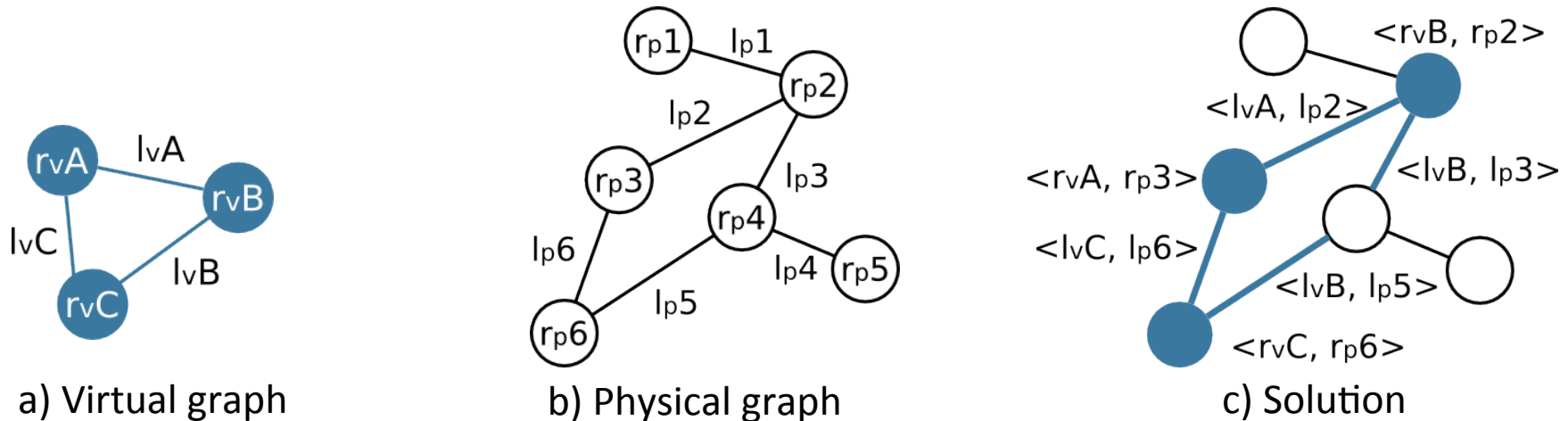
Allocation

⌘ Heuristic for VI allocation:

- Subgraph isomorphism detection [West, 2000] [Cordella et al., 2004] [Lischka and Karl, 2009]
- Subject to constraints: user's requirements and physical substrate capacity

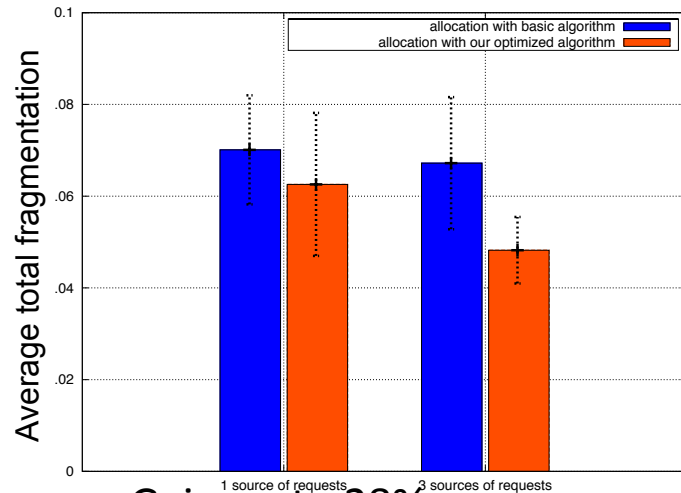
=> Location-aware algorithm [Koslovski et. al, IEEE/IFIP International Symposium on Integrated Network Management, 2011]

⌘ Subgraph isomorphism detection



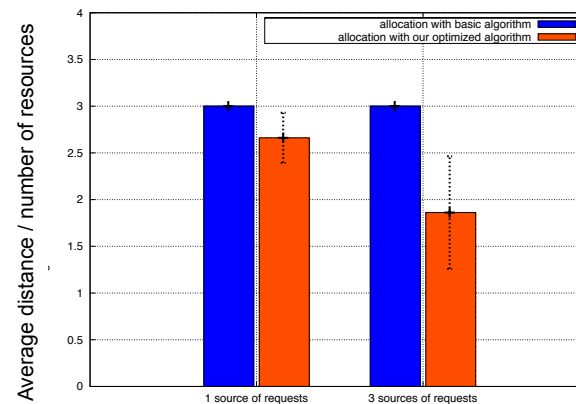
Experimental results

- Medium-size substrate **fragmentation**



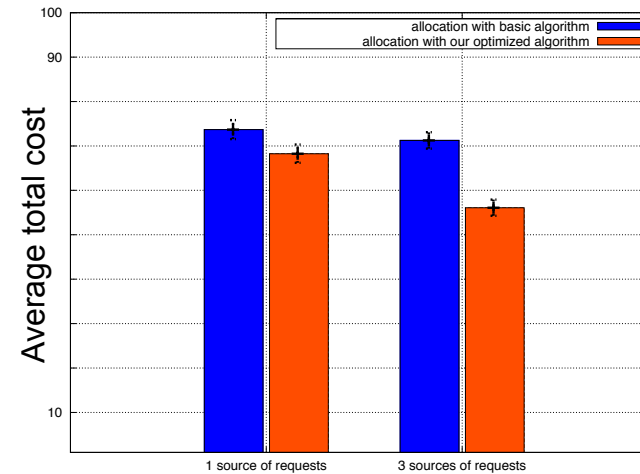
- Gain up to 28%

- Medium-size substrate **allocation quality**



- Average distance decreases -> allocation quality gain up to 39%

- Medium-size substrate **cost**



- Gain up to 21%
- Short virtual paths -> less physical resources