Controlling the Internet in the era of Software – Defined and Virtualized Networks

Fernando Paganini

Universidad ORT Uruguay

CDS@20, Caltech 2014

Motivation

1. The Internet grew in its first 30 years with limited role of mathematical theory.

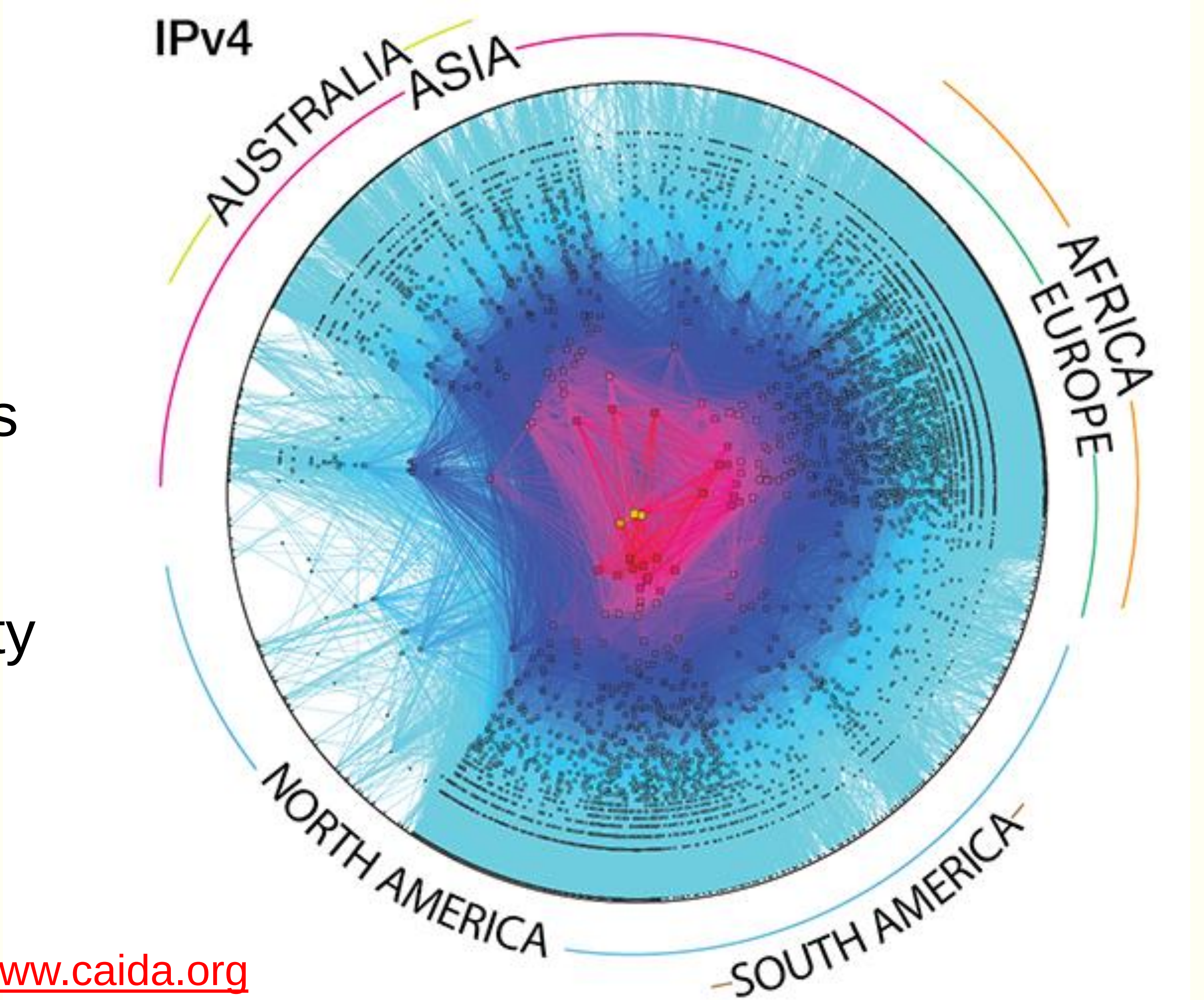
2. Theory based on economics+optimization+control developed since late 90s for protocols and resource allocation. Some (limited) practical impact.

3. The ground is shifting at the technical level with new paradigms: software-defined networks, virtualization.

4. Is there a role of CDS-type theory in this new era? What remains relevant? New challenges?

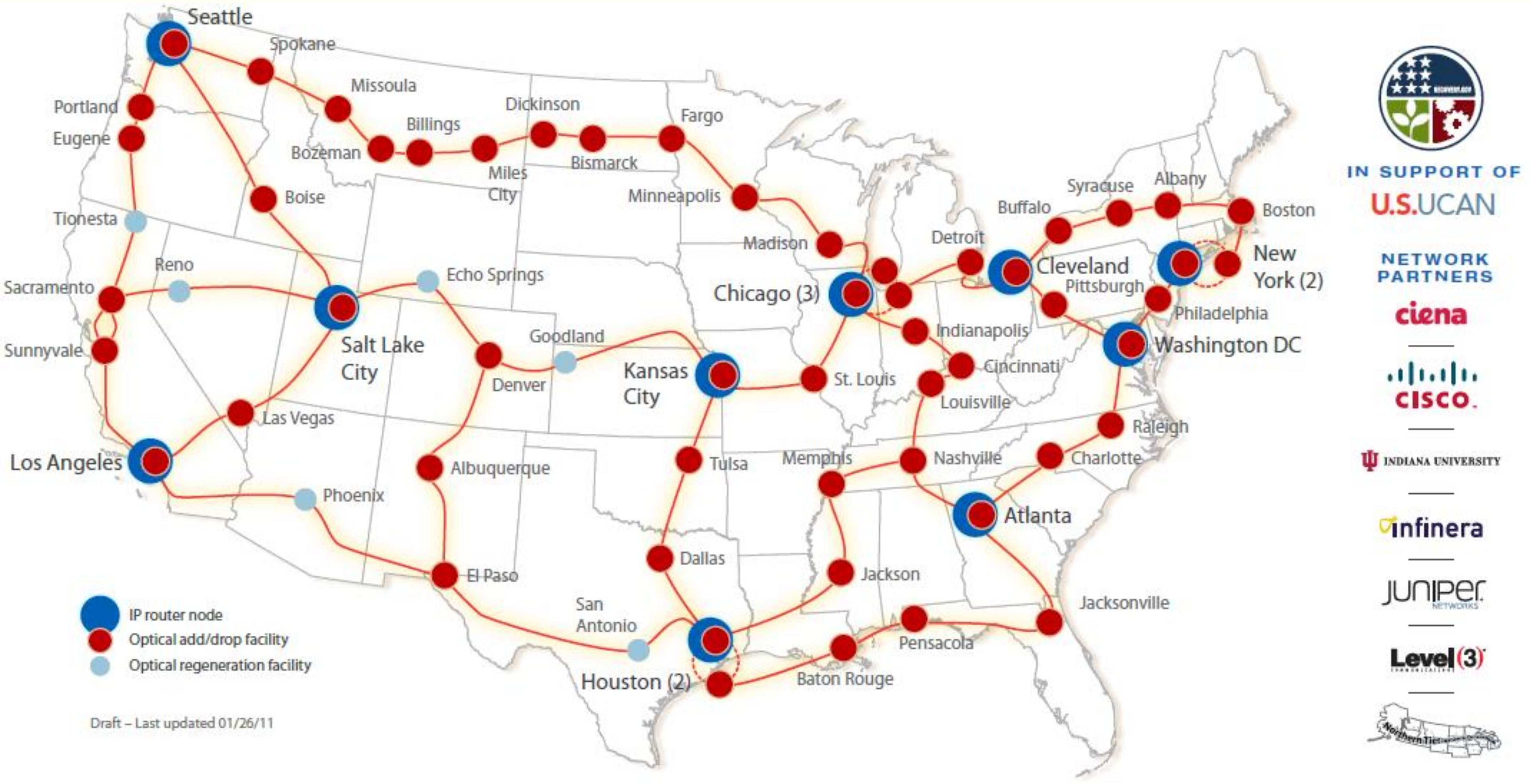
A network of worldwide scale...

Each dot is AS, radius indicates connectivity

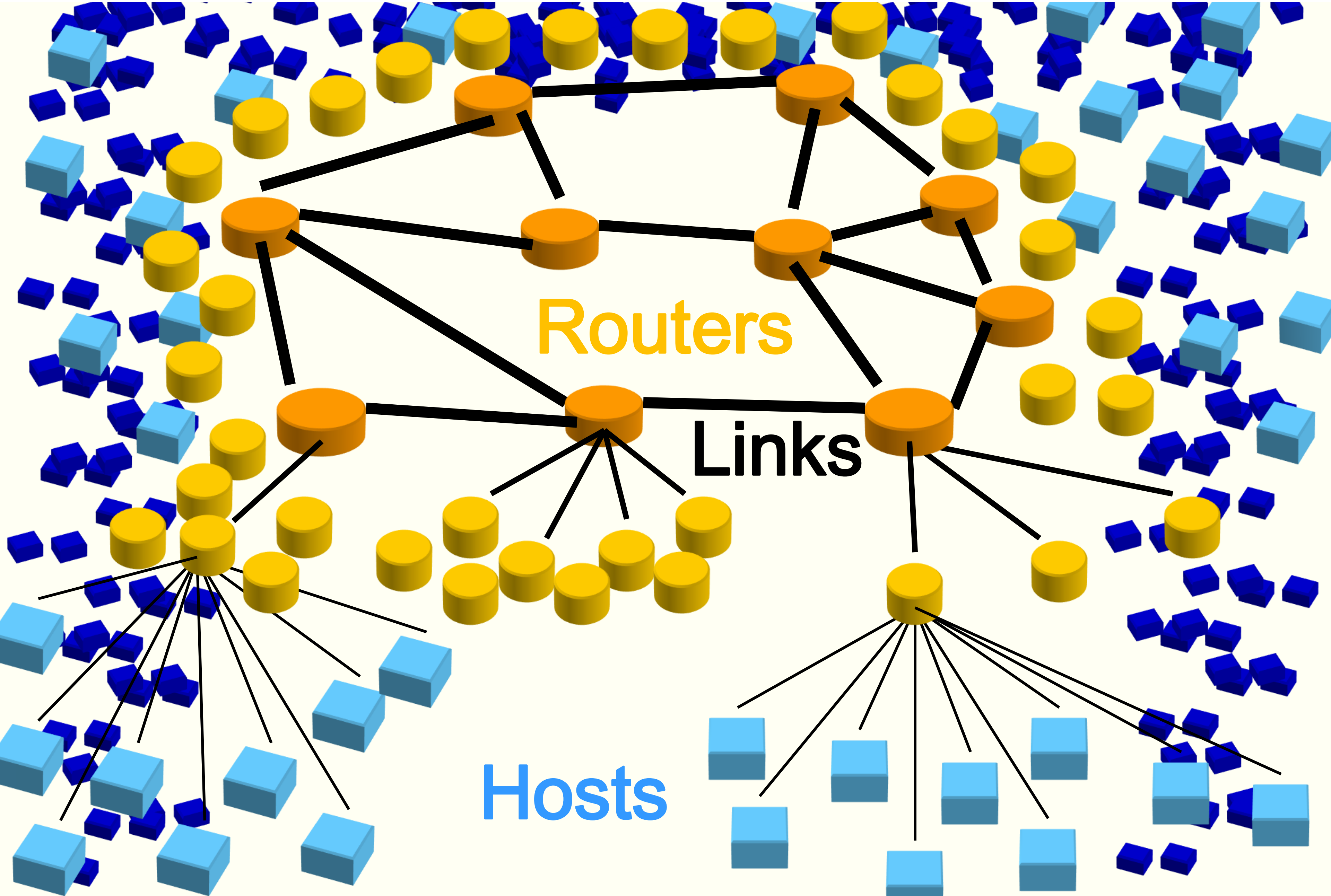


Source: http://www.caida.org

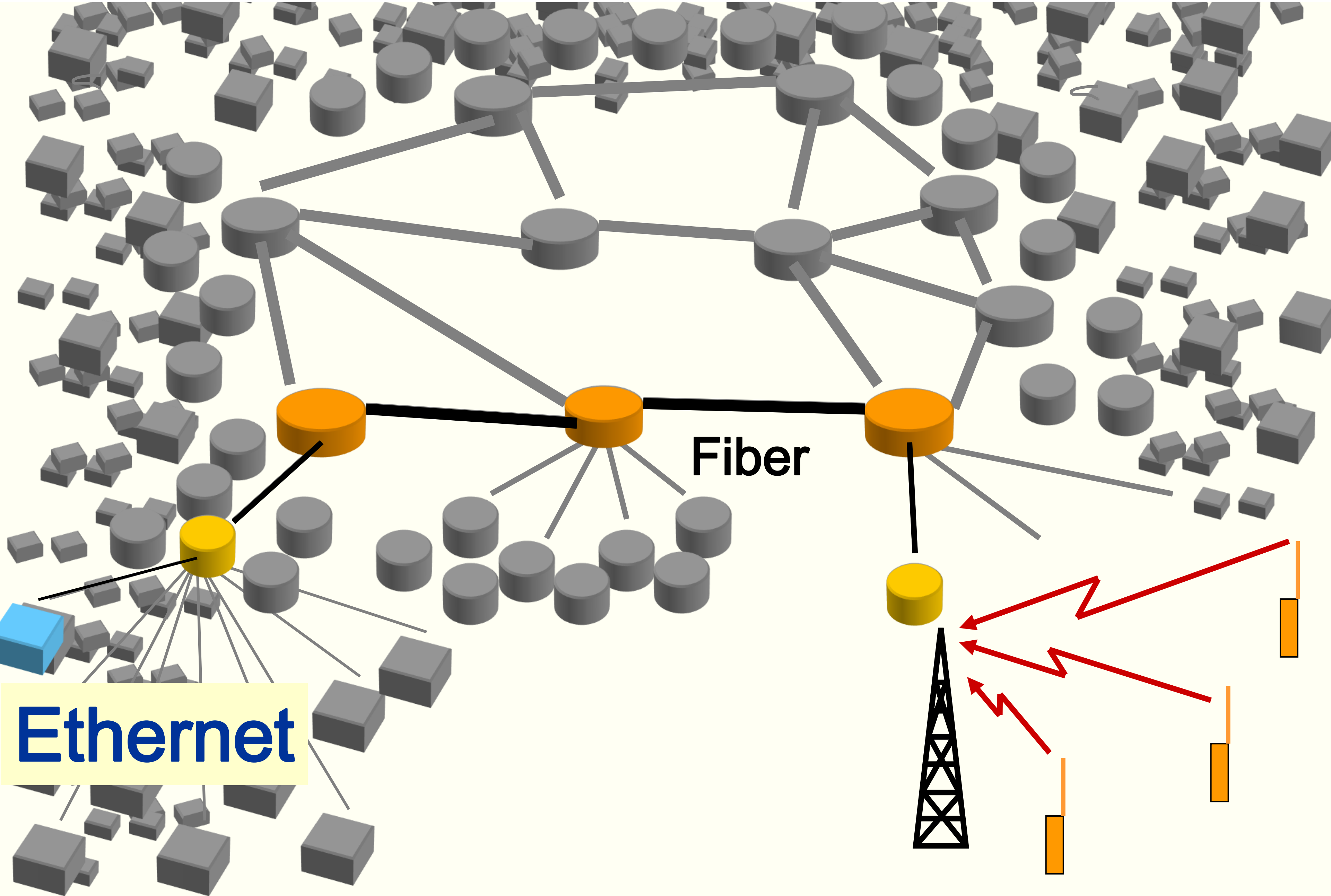
Core AS's are backbones



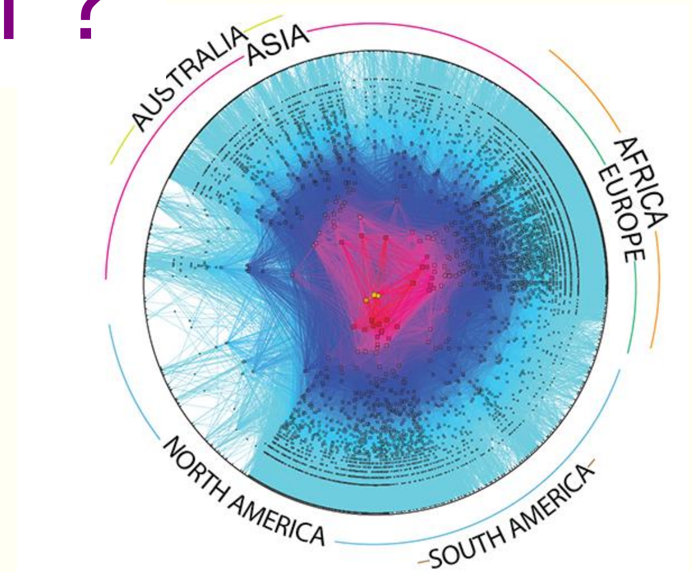
Outer AS's, access networks, …



Links have different technologies



Was it ever "under control"?



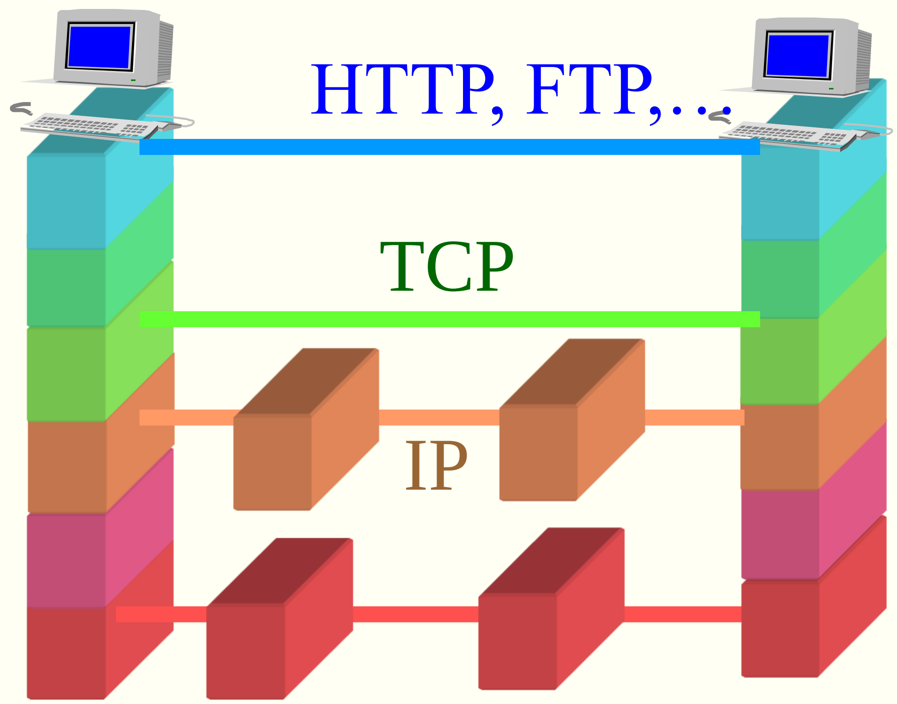
- No single function
- No common authority
- Changes all the time

Most control confined to within each AS.

But some global functionalities are required:

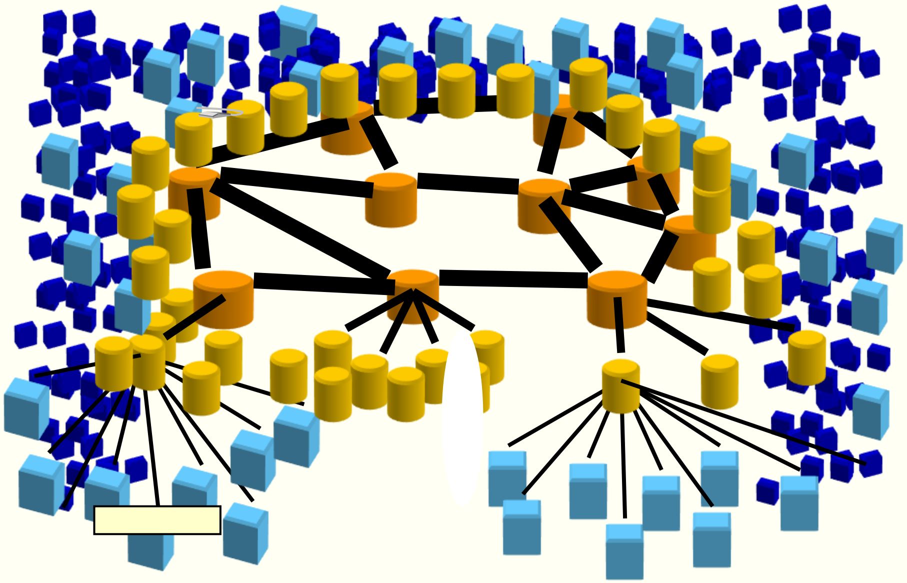
- Ensuring connectivity
- Regulating transport.

→ Protocols



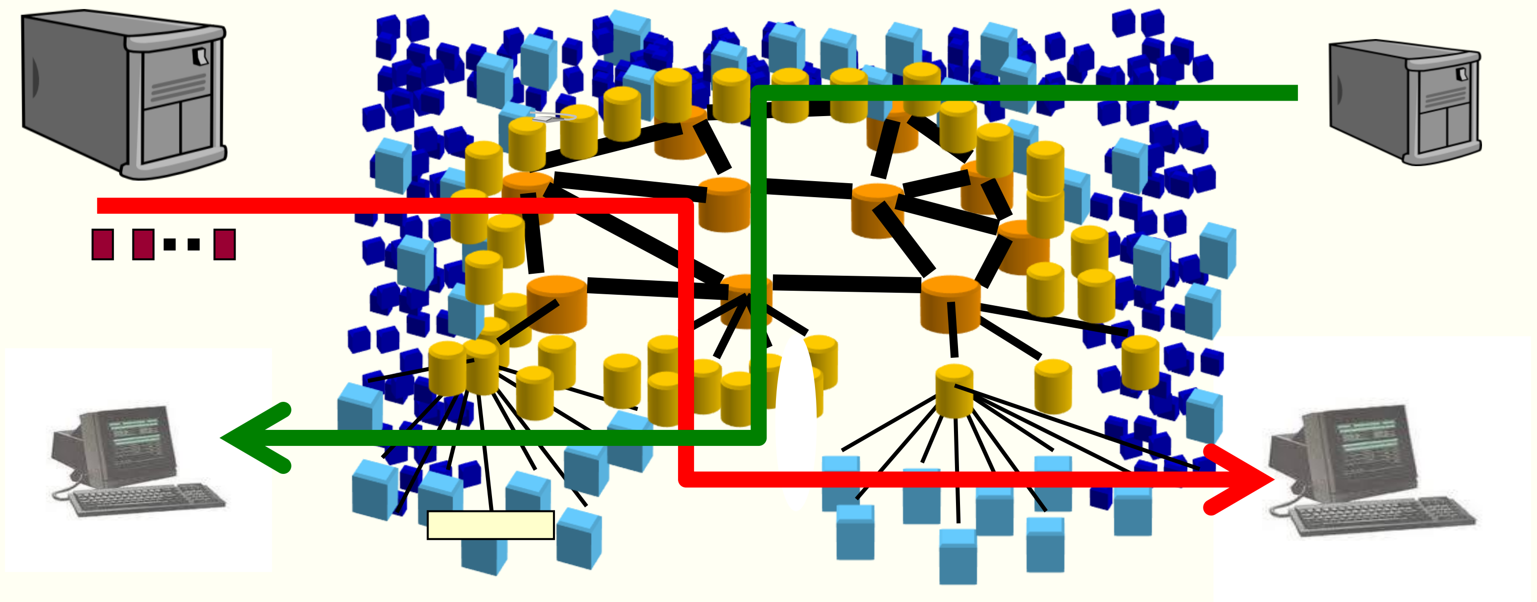
Two **objectives** and design **premises**

Ensuring global **connectivity** for changing networks: global addressing (IP), plus control of “logical state” under the **premise of decentralization.**



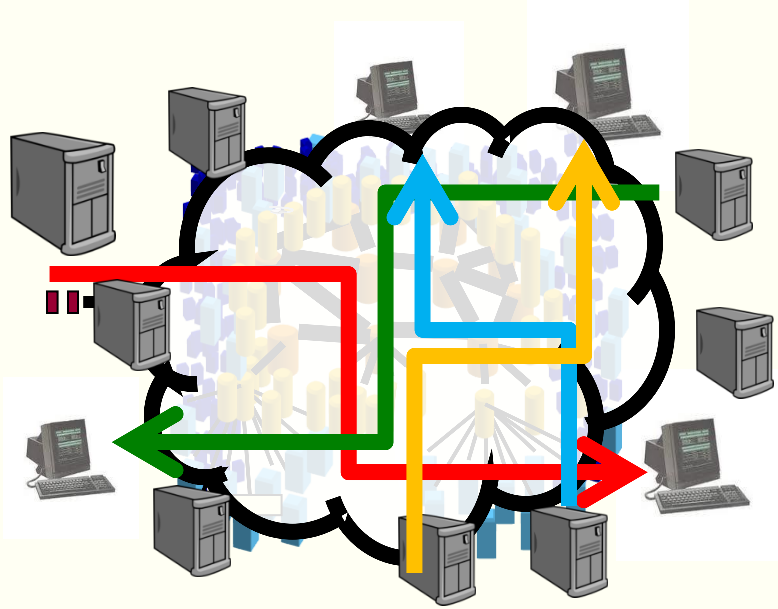
Two **objectives** and design **premises**

Ensuring global **connectivity** for changing networks: global addressing (IP), plus control of "logical state" under the **premise of decentralization.**



Regulating **transport** (TCP) under the **premise that bandwidth is scarce,** requires **feedback control.**

Congestion control



Regulate traffic sources to fit available capacity using congestion feedback.

However:

- ~10^9 hosts on the Internet!
- Decentralized decisions, **coupled** outcome.
- Not your standard feedback design!

Proposal (Kelly, late 90s):
To study **decentralized** allocation of **scarce bandwidth** resources: turn to **economic theory:**

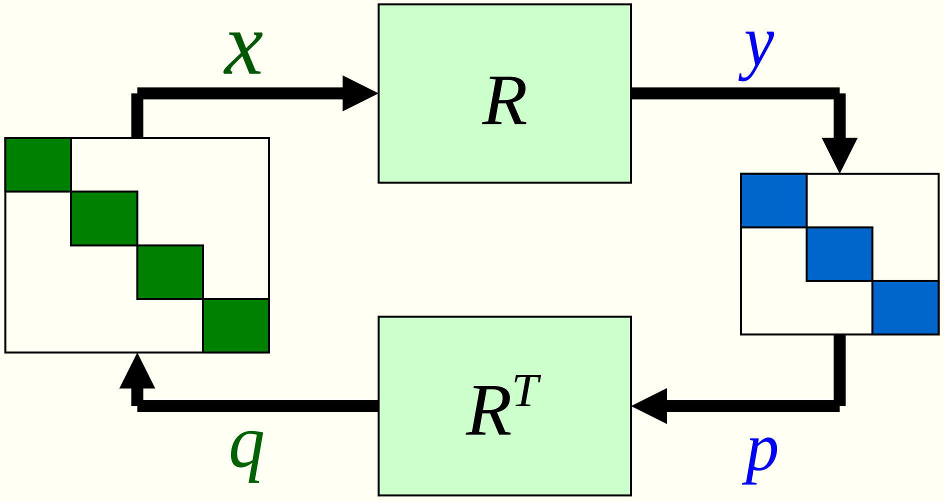
A utility $U_i(x_i)$ for each flow of rate x_i.

A price p_l for each scarce resource (e.g. link rate $y_l \leq c_l$).

Network Utility Maximization

$$\max_x \sum_i U_i(x_i), \text{ subject to } y_l = \sum R_{li} x_i \le c_l, \ \forall l.$$

- Convex program. Duality: prices = Lagrange multipliers.
- Optimization algorithms become dynamic control laws.
- Gradient (primal or dual) steps use decentralized info.



CDS is in the game!

- Lyapunov stability
- Nyquist to include delays [Vinnicombe ’00-’02] [Low-Doyle-P´, ´01-´02],…

Impact: interpret current protocol behavior, propose alternatives: e.g. (Fast TCP, Low et al).

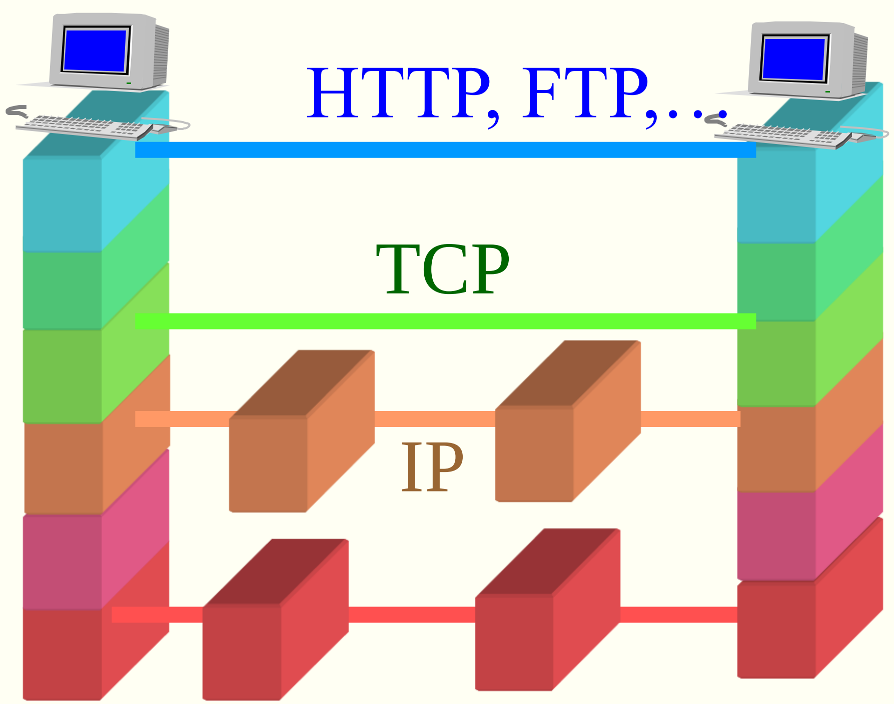
Cross-layer optimization

- Dynamic control in other layers: routing, medium access, admission of connections, etc.
- Can we jointly design them through a global NUM?
- Ideally, optimization decomposition should dictate layering [Chiang-Low-Calderbank-Doyle '07].
- Highlights from my group's work ('06-'10) :

Joint control of TCP and number of connections

Joint congestion control and multipath routing

Cross-layer control of MAC+ transport layer for wireless networks

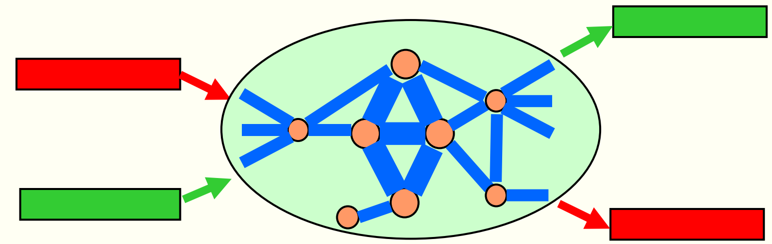


Optimization in Traffic Engineering

TE: distribute traffic inside an AS

Given: network and demands x^k

of rate per $k=(s,d)$ pair.



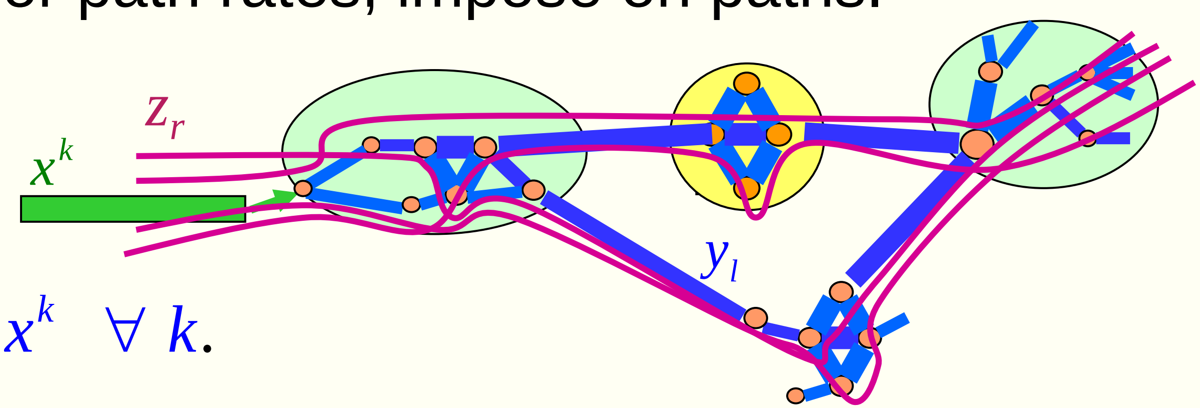
Multicommodity optimization: min $\sum_l \phi_l(y_l)$, s.t. $\quad y_l = \sum_k y_l^k, \quad y_l^k \geq 0,$

$\sum_{l \text{ in}} y_l^k = \sum_{l \text{ out}} y_l^k$ at interior node; $\sum_{l \text{ out}} y_l^k = x^k$ at source node of flow k.

- Requires multipath routing, not standard in IP. Workaround:
  - define end-to-end forwarding paths, implement via MPLS.
  - optimize offline for path rates, impose on paths.

min $\sum_l \phi_l(y_l)$, s.t.

$y_l = \sum_r R_{rl} z_r, \quad \sum_{r \in k} z_r = x^k \quad \forall\, k.$

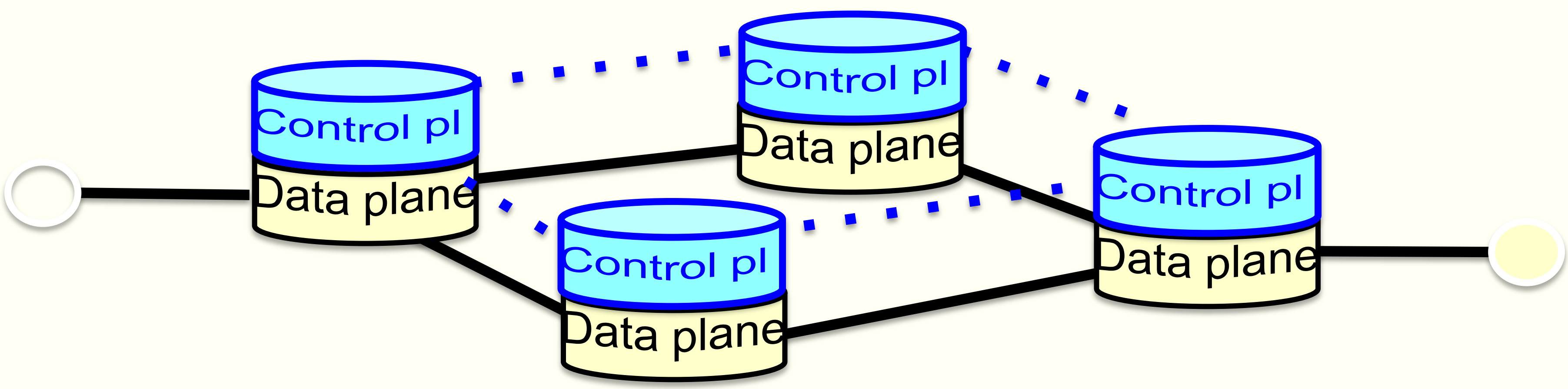


Dynamic Traffic Engineering

- If demands vary, use congestion feedback to control either:
  - Input rates in predefined end-to-end paths.
  - Routing splits at nodes, per destination.
- Integrates with congestion control
  [Kelly et al. ’98, Han et al ’03, Voice ’07, P’ Mallada ‘09].
- No strict convexity. To avoid oscillations, modify gradient laws.
- Practical impact? So far, essentially zero
  (but see recent efforts by Kevin Tang’s group at Cornell).
- Reasons: legacy constraints on IP routing, also the **scarcity premise** rarely holds inside AS!
  - ISPs deliberately choke input traffic, overprovision core. Simplifies management, resilience.
  - Good for charging for access!

So what's new? Reviewing our Premises

- The mandate on decentralization means that the control plane (which figures out how packets are routed) should have the topology of the data plane (which actually forwards the packets).
- In particular: control algorithms should involve message passing among neighboring routers.



Software – Defined Network

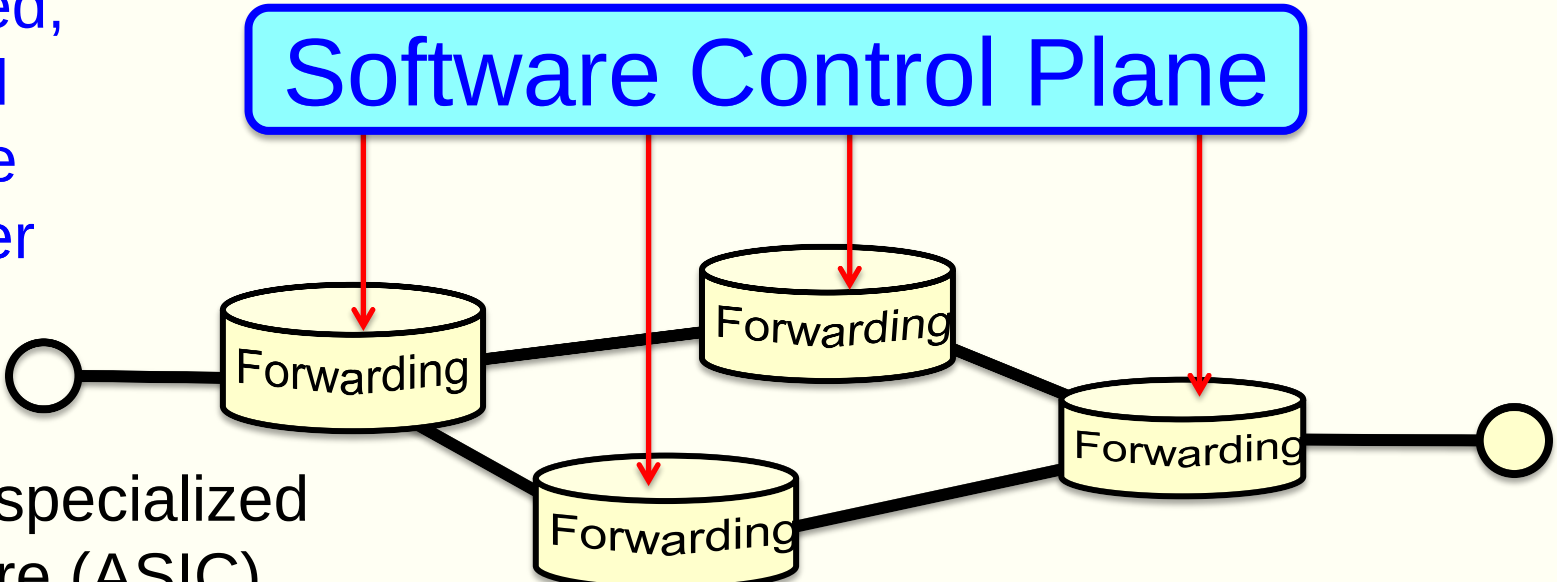
[as defined by McKeown, IET Appleton Lecture ‘14]

A network in which the control plane is physically separate from the forwarding plane.

and

A single control plane controls several forwarding devices.

Runs on centralized, general purpose computer



Runs on specialized hardware (ASIC)

Implications

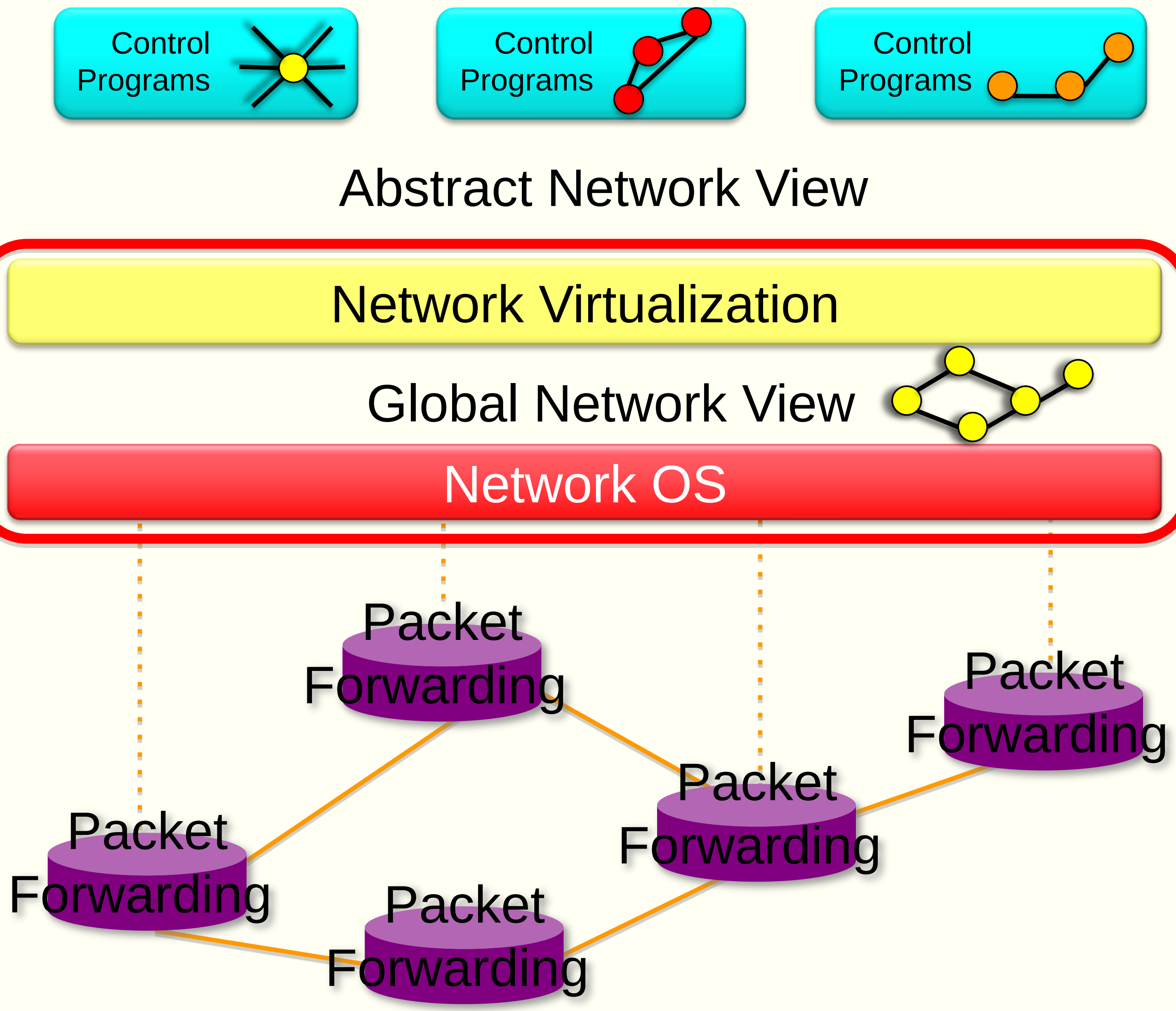
- A computer with global network info makes traffic control decisions, imposes them on “dumb” forwarding devices.
- Open source protocols (OpenFlow) allow interaction between planes. Business implications.
- Makes sense at AS scale (where TE is done).
- Our decentralization premise has been removed! Proponents educate us on the advantages of centralized control! Of course we knew this…
- Can now do centralized multicommodity optimization. Best interior point method in lieu of gradient descent.
- Less need for cute microeconomics, or for dynamics in solving the resource allocation.
- Control *over* network issues may appear.

One step further: network virtualization

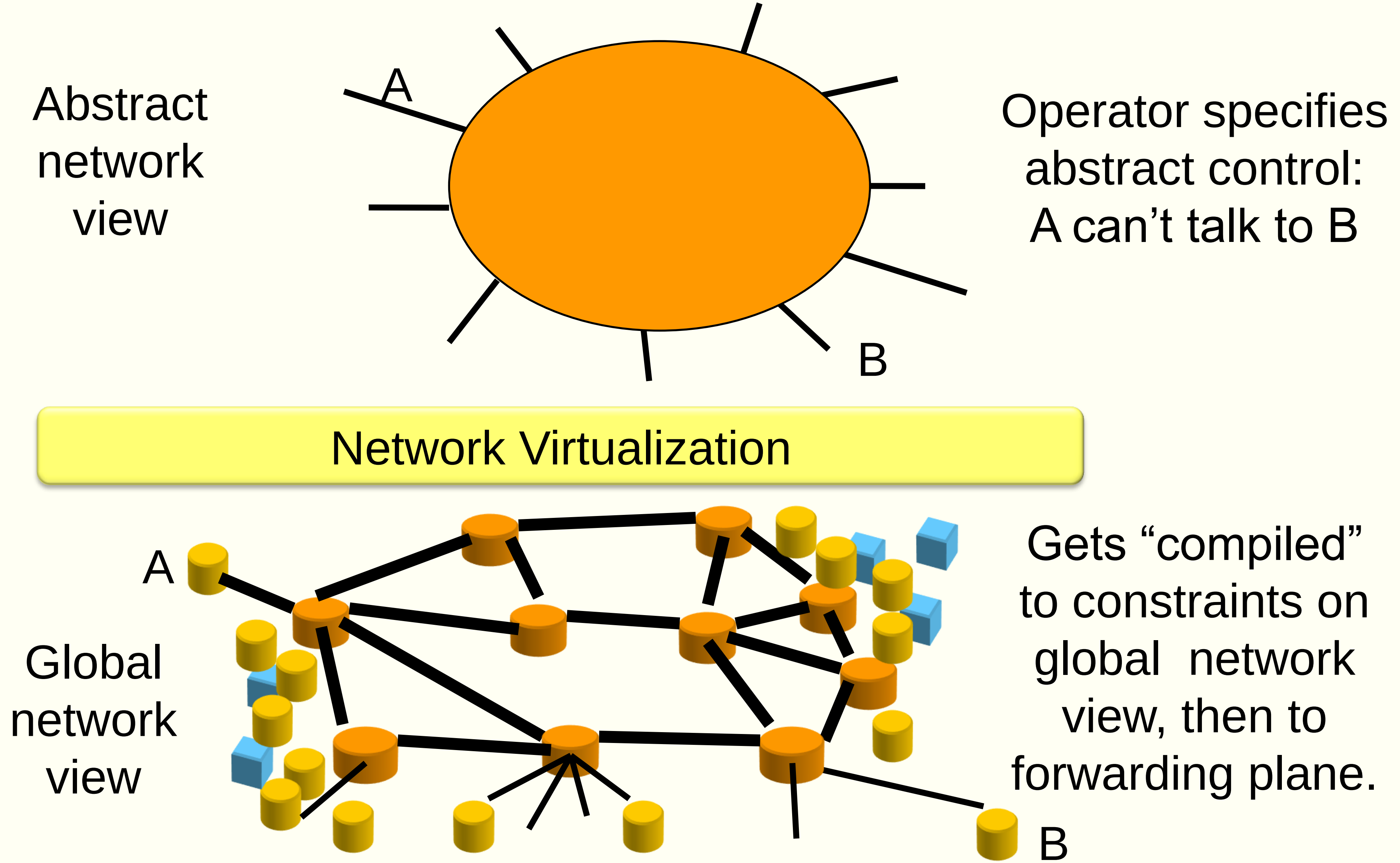
- Once a global network view sits in a centralized server, this view can be further abstracted.
- Abstraction: offer a simpler view of the network, avoiding details of the topology, for use by an application that demands services on it.
- Virtualization is a popular device in computers: e.g. install a virtual machine that emulates Windows while running on a Linux OS.
- Analogy:

  forwarding plane is hardware, control plane is OS, virtualization allows this network to emulate another.

Network virtualization

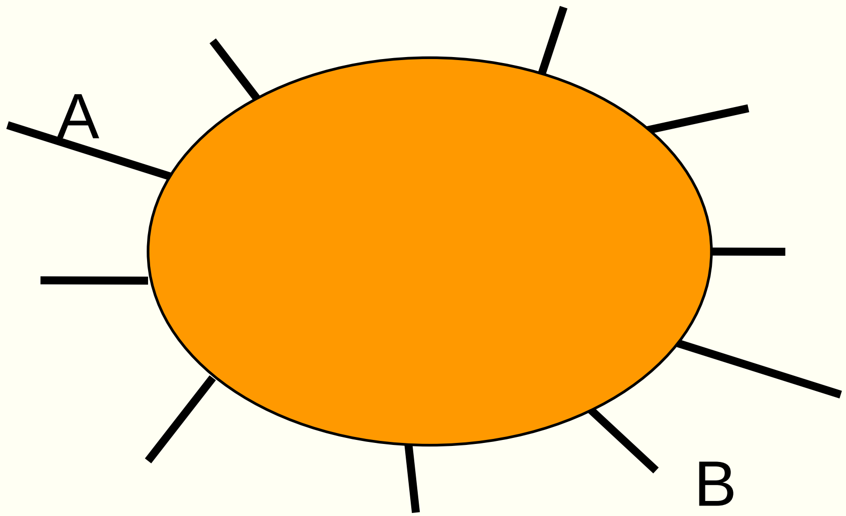


Example: abstracting connectivity
(from Scott Shenker, 2012)



Can we abstract transport?

Abstract network view

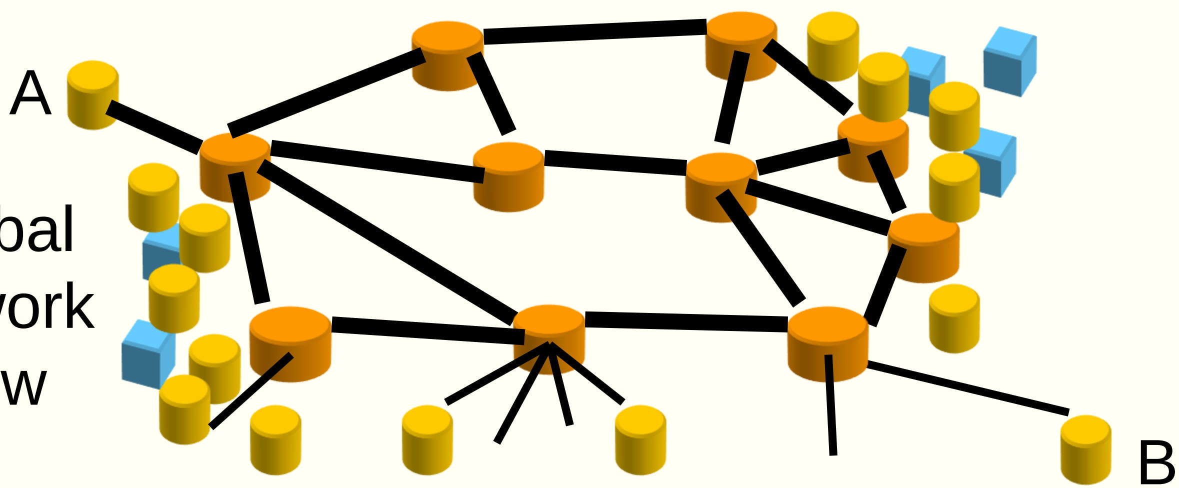


How much bandwidth can we offer A → B?

Network Virtualization

Answer coupled nontrivially with other flows (polytope in rate space).

Global network view



Hard to abstract, unnless core is highly overprovisioned.

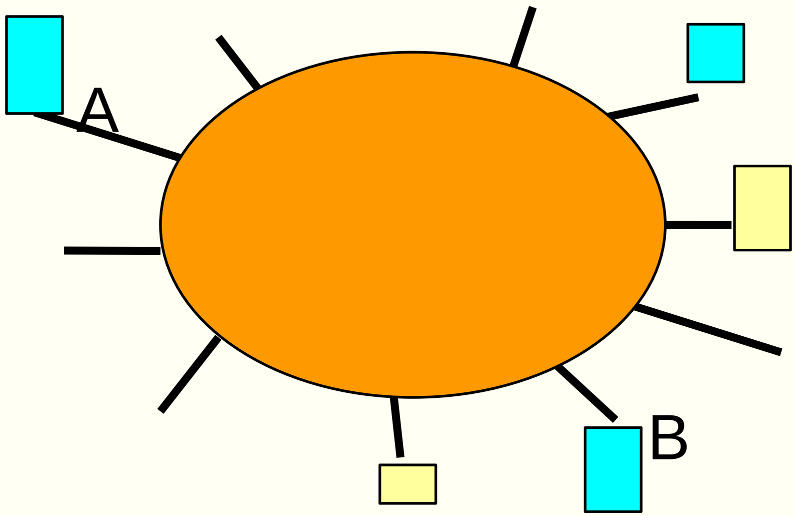
Discussion

- Abstraction is pervasive in software engineering, enabler for layered innovation.

- But performance considerations often don’t make it through the “hourglass”. Perfomance degrades unless hardware keeps running faster.

- Similar things can happen with virtualized networks. Unless the forwarding core is overprovisioned to an even greater degree than in current ISPs..

In an abstracted network, role of CDS?



- Idealized connectivity, bottlenecks in access.
- Focus moves to the *Content* layer outside.
- Centrally planned CDN, or unstructured peer-to-peer.

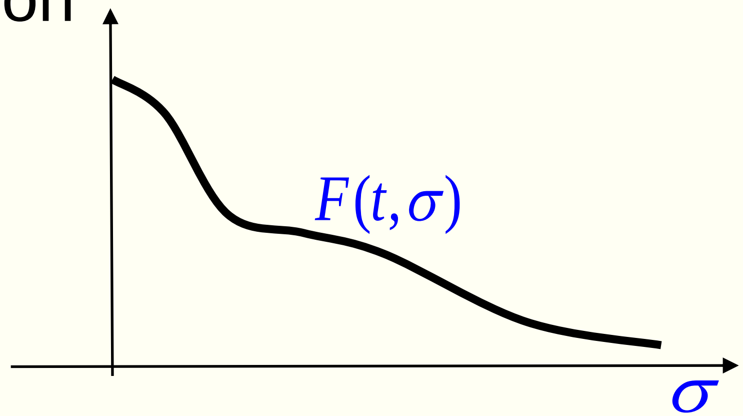
Content dissemination dynamics in p2p

- Peers arrive at a network to download file, exchange pieces with other following reciprocity rules. Depart some time after completion. Issues:
  - Population dinamics.
  - Download progress dynamics.

Highlights from our work on p2p dynamics

- PDE model for population as a function of time and download progress [Ferragut - P' CDC11, CISS12, ITC12]

$$\frac{\partial F(t,\sigma)}{\partial t} = \lambda + r(F, y, \sigma)\frac{\partial F(t,\sigma)}{\partial \sigma}$$



Results: equilibrium, stability, noise response, transient times.

- Multi-class versions for networks with heterogeneous access: Global stability invoking tools of monotone dynamical systems. [P'-Ferragut, CDC 13 -14].

- Reciprocity and proportional fairness: Sinkhorn iteration, Gibbs sampling [Zubeldía-Ferragut-P', Allerton13]

- M / G processor sharing queues and stochastic fluid limits in P2P networks [Ferragut, Stochastic Networks 14]

Some final thoughts

- Holy grail for us CDS-types: theory that is both mathematically deep *and* practically relevant.
- These objectives are often at odds: Practicioners favor simple (non-mathematical) building blocks and interfaces.
- The most popular "hourglasses" are those with no math in the 'waist'. Fast innovation, but loses quantitative view.
- In network control, we had success in coming up with "mathematical" layering interfaces based on duality and prices. Impact ran into legacy constraints.
- New developments (SDN, virtualization) open up the field again! But beware of too simple interfaces!
- Regardless: there are interesting opportunities with a content-centric view of the network.