Future Internet Transport Layer - Heading towards a Post-TCP Era?

Michael Scharf
Institute of Communication Networks and Computer Engineering
University of Stuttgart
michael.scharf@ikr.uni-stuttgart.de

ECOC Workshop "Future Internet Design" - September 16, 2007

This work is partly funded by the German Research Foundation (DFG) through the Center of Excellence (SFB) 627 "Nexus".

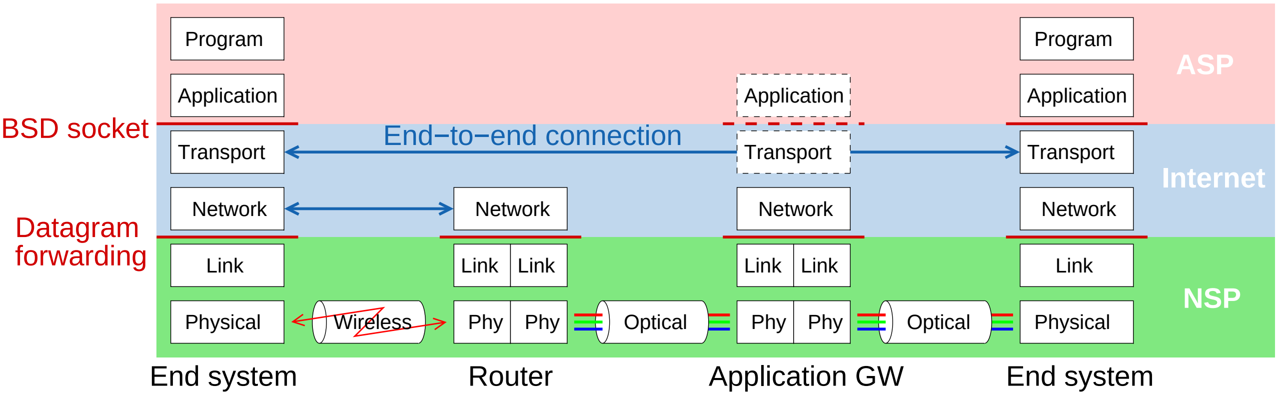
Outline

- **Status quo**
  - Internet
  - Internet transport layer
- **TCP/IP midlife crisis**
- **Transport "layer" future**
  - Evolution
  - Revolution
- **Conclusions and outlook**

Status Quo - Internet

Never standardized, still evolving...

Internet TCP/IP-"Architecture"



ASP ... Application Service Provier, NSP ... Network Service provider

- **Current Internet := layer 2.9 to layer 4.5**
  - Network layer: Addressing, routing, *peering*
  - Transport layer: End-to-end interconnection of *intelligent* hosts
- **Narrow APIs, no control-plane**

➥ **TCP/IP as "spanning" layer**

Status Quo - Internet

~~Internet~~ Arpanet Design Paradigms

Design objectives	Paradigms
1. Communication must continue despite loss of networks or gateways	1. Packet switching
2. Multiple types of communications services	2. Layering (with simple APIs)
3. Accomodate a variety of networks	3. A network of collaborating networks (IP spanning layer)
4. Distributed management of its resources	4. Intelligence in end systems (end−to−end principle)
5. Cost effective	
6. Host attachment with a low level of effort	
7. Resources must be accountable	

From D. Clark, "The Design Philosophy of the DARPA Internet Protocols", Proc. SIGCOMM 1988, Sept. 1988

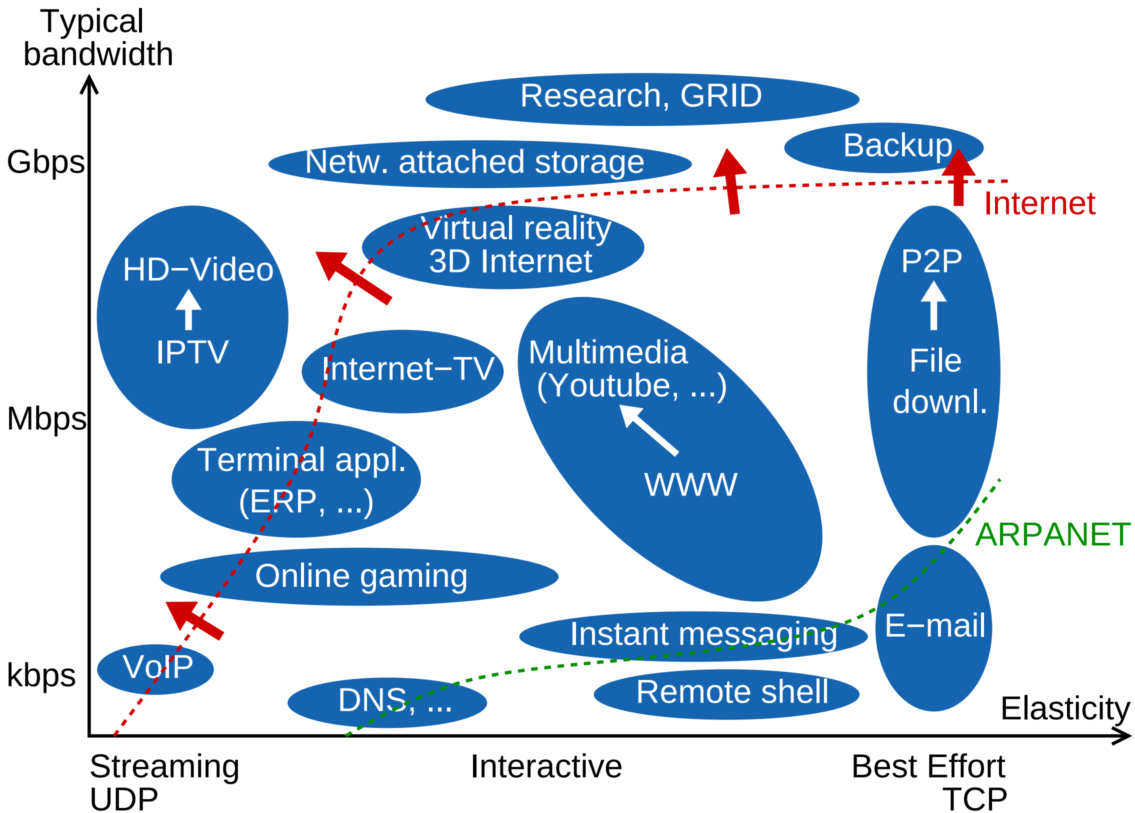
➥ **TCP/IP heavily influenced by these paradigms**

➥ **New paradigms? Other protocol architecture!**

Status Quo - Internet

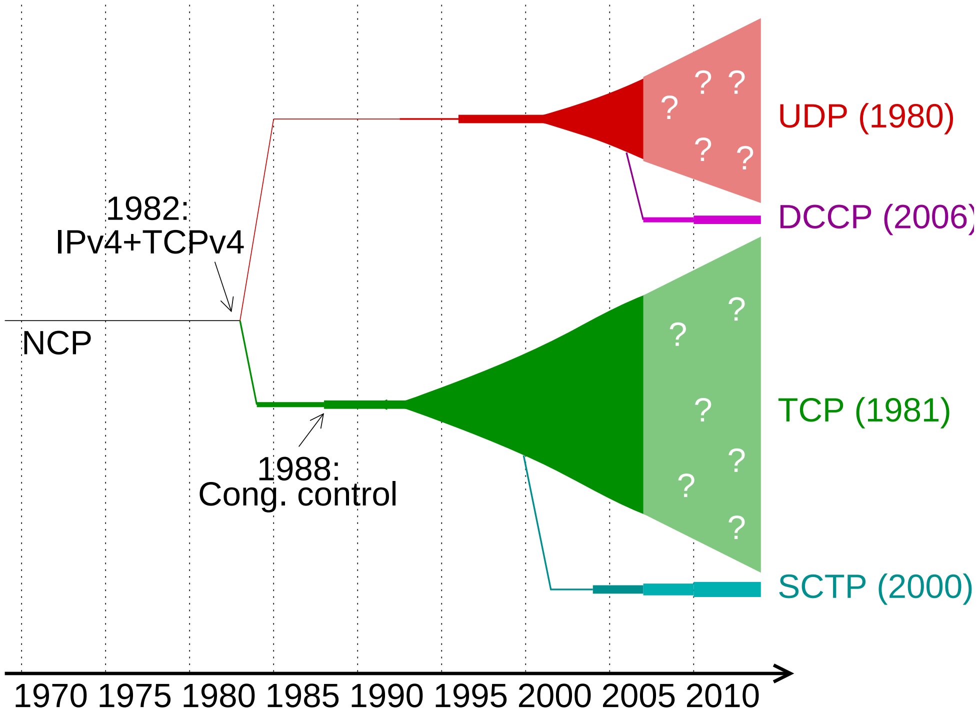
We network guys don't invent killer apps...

Applications



Status Quo - Internet Transport Layer

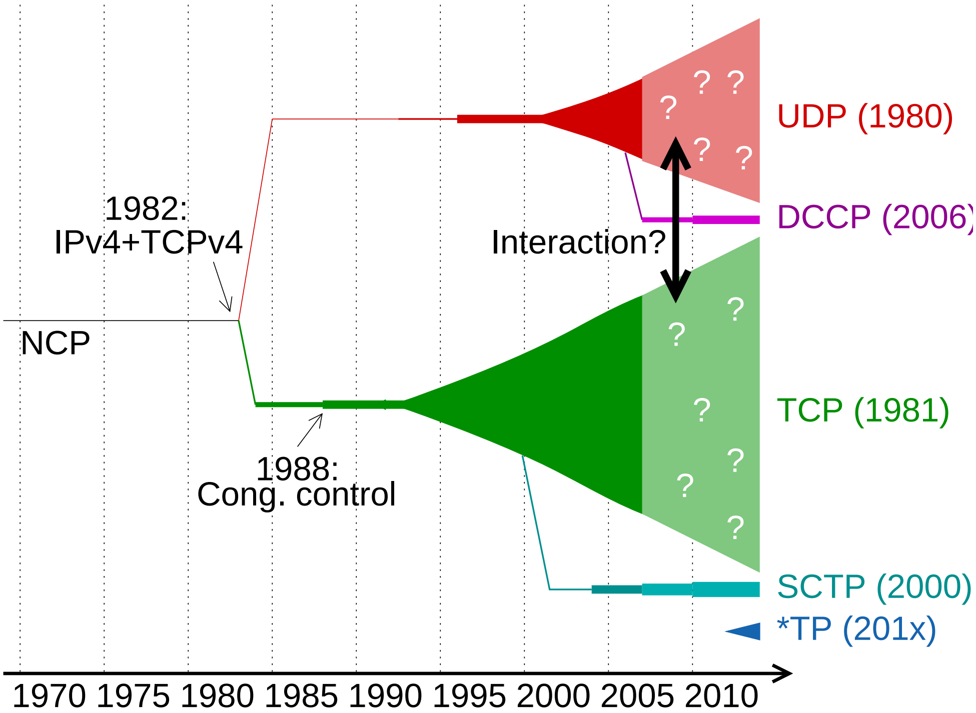
Looking Back ...



➥ **Only 4 significant transport protocols in more than 25 years ...**

Status Quo - Internet Transport Layer

Looking Back ...



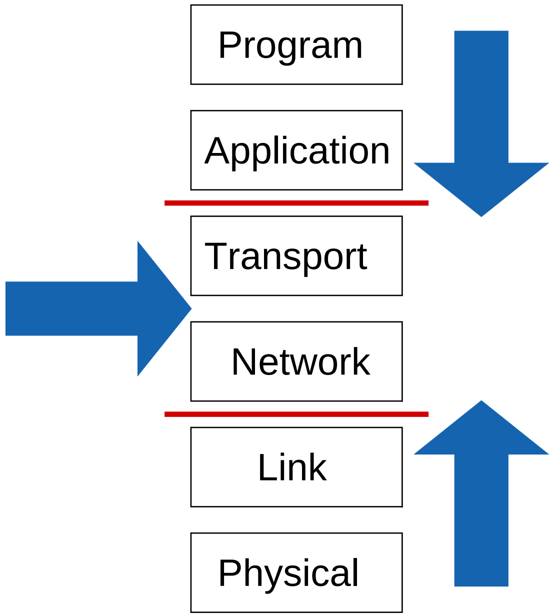
➡ **Only 4 significant transport protocols in more than 25 years ...**

Midlife Crisis

New (and Old) Requirements

From within:

- Security (spam, DoS)
- QoS+charging
- Multicast, *−cast
- Cross−layer stuff



From above:

- Overlays
- Internet of things
- Network−challenging apps (GRID, ...)

From below:

- Photonics (100G, ...)
- Sensor networks
- Interstellar networks

➥ **"Future Internet: Just more QoS and multicast?" (H. Schulzrinne)**

Midlife Crisis I: Congestion Control

Design Principles of V. Jacobson's Congestion Control

- **Sender-side control of data rate by congestion window**
- **Greedy probing of available bandwidth on path (window increase)**
- **Implicit congestion feedback by packet loss (window decrease)**

Characteristics

Just putting more bandwidth might not help ...

- **Key impact on application performance**
- **Best effort, elastic applications only, no QoS**
- **Vague notion of fairness (unfair to connections with larger RTT)**

➥ **Never perfect, but usually *good enough***

However ...

- **Network characteristics changed a lot since 1988**
- **More and more network-demanding applications**
- **Fairness may become an issue (e. g., network neutrality debate)**

Example: High-Speed WANs

- **Problem: Large bandwidth-delay products**
  - Standard TCP congestion control not well suited for large window sizes

    Example: 10 Gbit/s TCP throughput with MTU=1500-byte, RTT=100 ms requires
    - Average congestion window of 83,333 segments
    - At most one drop/mark every 5 bill. packets (one drop every 1 2/3 hours)
  - Long convergence times and significant unfairness
- **Solution approaches**
  1. Increase MTU
  2. UDP-based protocols (with some application-level congestion control)
  3. Modification of TCP congestion control in sender
     - More aggressive window increase functions
     - Delay-based congestion control
     - ➥ Multitude of proposals (Highspeed TCP, Scalable TCP, HTCP, FAST TCP, BIC TCP, CUBIC TCP, Compound TCP, ...)

➥ **Next challenge: 100 Gbit/s links?**

Midlife Crisis I: Congestion Control

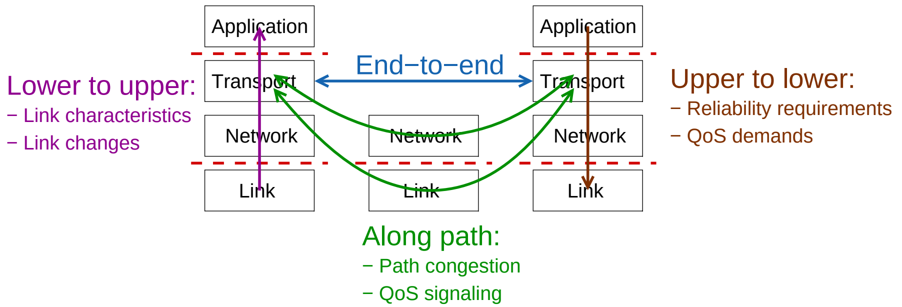
Open Issues

- **Extrem variety of networks**
  From sensor networks to high-speed optical networks
- **Large range of application requirements**
  Many non-elastic applications (e. g., multimedia, pseudo-wires)
- **TCP-unfriendly path characteristics**
  Variable link capacities, corruption packet loss, packet reordering, ...
- **Multi-domain operation**
  Misbehaving senders, receivers, and applications
- **Fairness**
- **...**

➥ **Any chance for *one* Internet congestion control?**

Further details in: Michael Welzl, Dimitri Papadimitriou, Michael Scharf, "Open Research Issues in Internet Congestion Control", IETF internet draft, work in progress, July 2007, draft-irtf-iccrg-welzl-congestion-control-open-research-00.txt

Cross-Layer Information Exchange



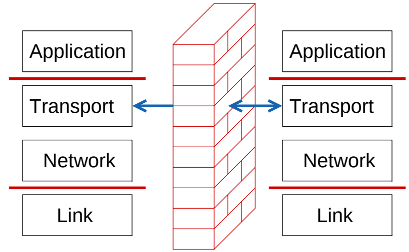
➥ **Significant potential for optimization**

Challenges

- **Host local: Standardized interfaces (device - OS - application)**
- **Remote along path**
  - Protocol extensions or new protocols (in-band or out-of-band)
  - Interaction with routing, IP tunnels, ...
  - Security issues (AAA, DoS prevention, ...)

➥ **Internet control plane?**

The Raise of Application Gateways



Application GW

- **Intermediaries break up end-to-end semantics**
- **Many functions**
  - Security, information hiding (NAT/firewalls, session border controllers, ...)
  - Performance optimization (Web caches, WAN accelorators, ...)
  - Content processing (compression, virus checking, transcoding, ...)
  - Facilitate rendezvous and/or forwarding (SIP proxies, SMTP relays, ...)
- **Problems**
  - May become single point of failure
  - May hinder/limit communication and/or new protocol extensions
  - May require certain trust relationships

A bug, or a feature?

➥ **End-to-end vs. "balkanization" of the Internet?**

Recent TCP Research+Standardization

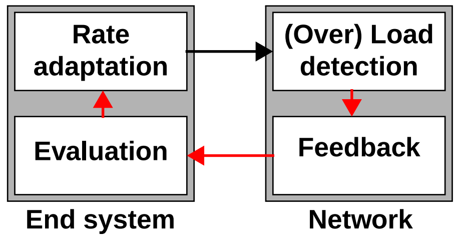
Incremental → Evolution → Revolution

- TCP tweaking (always)
- Highspeed−TCP (2003)
- AQM (1998)
- ECN (2001)
- Quick−Start (2007)
- re−ECN (?)
- XCP (?)
- (BISDN 2.0 − IMS/TISPAN) (200x)

Incremental → Evolution → Revolution

- **Major focus: Congestion control**
- **Main objectives**
  - Optimize performance for high-speed and wireless networks
  - Increase fairness

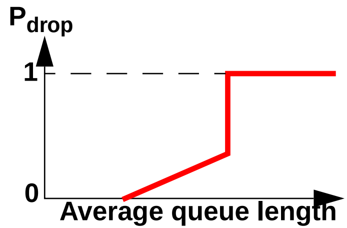
➥ **Question: Role of network, i. e., routers?**



Transport Layer Evolution: Examples

Active Queue Management (AQM) - RFC 2309 (1998)

- **Replace drop-tail buffers in routers**
  - Avoids synchronization effects
  - Improves fairness
- **Reality check**
  - Enabled in some routers
  - No single optimal parameter set



➥ **Simple idea, but still too complex for the real world?**

Explicit Congestion Notification (ECN) - RFC 3168 (2001)

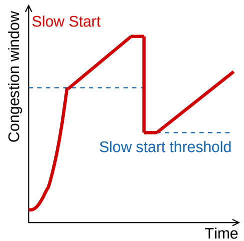
- **Congested router sets bits in IP header instead of dropping packets**
- **Reality check**
  - Support by major operating systems, but not enabled by default
  - Deployment problems with buggy routers and middleboxes

➥ **Chicken-egg deployment problem - and no pressing need?**

Transport Layer Evolution: Examples

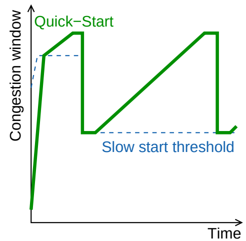
Quick-Start TCP Extension - RFC 4782 (2007)

Slow-Start:



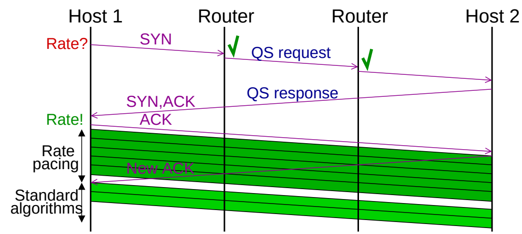
- One pillar of TCP congestion control
- Exponential window growth

Quick-Start:



- Recent experimental TCP extension
- (Almost) immediately use large window

- **Speeds up interactive WAN applications**
  - After connection setup or idle periods
  - For large bandwidth-delay products
- **Reality check**
  - Requires support in *all* routers
  - Some open (research) issues

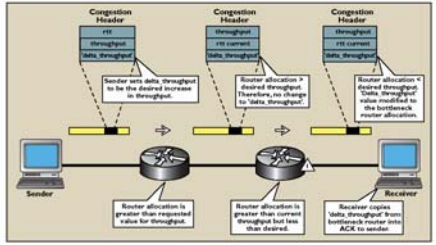


➥ **Short-term deployment in puplic Internet unrealistic**

Transport Layer Evolution: Examples

eXplicit Control Protocol (XCP) - RFC 5xxx

- **Explicit congestion feedback from routers**
  - Some congestion state in packets
  - Feedback on rate increment/decrement
  - Routers do some per-packet calculations (but no per-flow state)



Source: A. Falk et. al., "Transport Protocols for High Performance"

- **Potential advantages**
  - High link utilization for high-speed WANs
  - Fair bandwidth allocation
- **Reality check**
  - Complexity: Requires per-packet computations in routers
  - Unsolved issues with short-lived flows
  - Only a congestion control framework - no transport protocol (so far)

➥ **Revolutionary research ...**

~~Internet~~ Arpanet Design Paradigms

Design objectives

1. Communication must continue despite loss of networks or gateways
2. Multiple types of communications services
3. Accomodate a variety of networks
4. Distributed management of its resources
5. Cost effective
6. Host attachment with a low level of effort
7. Resources must be accountable

Paradigms

1. Packet switching
2. Layering (with simple APIs)
3. A network of collaborating networks (IP spanning layer)
4. Intelligence in end systems (end−to−end principle)

From D. Clark, "The Design Philosophy of the DARPA Internet Protocols", Proc. SIGCOMM 1988, Sept. 1988

Transport "Layer" Revolution

~~Internet~~ Arpanet Design Paradigms

Design objectives	Paradigms
1. Communication must continue despite loss of networks or gateways	1. Packet switching
2. Multiple types of communications services	2. Layering (with simple APIs)
3. Accomodate a variety of networks	3. A network of collaborating networks (IP spanning layer)
4. Distributed management of its resources	4. Intelligence in end systems (end−to−end principle)
5. Cost effective	
6. Host attachment with a low level of effort	
7. Resources must be accountable	

From D. Clark, "The Design Philosophy of the DARPA Internet Protocols", Proc. SIGCOMM 1988, Sept. 1988

That’s what transport layer is about ...

But, any internetwork architecture has to provide

communication between various applications on end-systems

... not necessarily a "layer" with service primitives we have today

Transport "Layer" Revolution

Actually, a network layer problem!

Design Space for Congestion Control

Implicit network feedback Loss-based, delay-based, bandwidth estimation techniques	or	**Explicit network feedback** In-band signaling Out-of-band signaling
No state in routers ("end-to-end")	or	**Some state/processing in routers (per packet/per RTT/...)**
TCP friendly	or	**More/less aggressive/fair/... than TCP**
Best effort	or	**Support of precedence/QoS**

Design Space for Transport Functions

see SCTP and DCCP

Reliable transfer	or	**Also partial reliable/unreliable transfer**
Single stream	or	**Multiple streams**
Single path	or	**Multiple (disjoint) paths**
Unicast only	or	***cast support**

(adapted from: S. Shalunov et. al., "Design Space for a Bulk Transport Tool")

Revolution - Food For Thought (1)

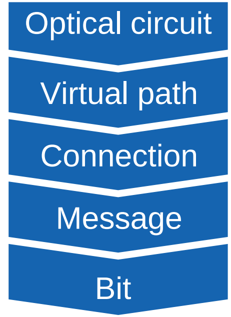
About Layering

Program	Connection−oriented (e. g., WS−*)
Application	Connection−less (HTTP)
Transport	Connection−oriented (TCP)
Network	Connection−less (IP)
Link	Connection−oriented (MPLS, ATM, UMTS, ...) Connection−less (Ethernet, ...)
Physical	Connection−oriented???

- **Reduce "self-similarity" in stack?**
- **Get rid of static layering?**
- **Transport layer functions in user space, instead of kernel space?**
  - Higher performance (locking, caching)
  - Less complexity, more flexibility
- **What about better support for tunneling?**

Revolution - Food For Thought (2)

About Granularity



- **What is the minimum granularity of data exchange? What timescales?**
- **What are the communication primitives (in particular, for non-bulk-data transport)?**
- **Role of photonics (e. g., dynamic establishment transparent optical paths)?**

About Performance

- **Where (and how) to handle resource sharing?**
- **Traffic engineering per aggregate/flow/connection/message/...? Centralized or distributed?**
- **What levels of reliability, availability, resilience?**
- **What about self-optimization?**

Revolution - Food For Thought (3)

About Addressing

- Anything more intelligent than port numbers?
- Incorporate security and access control features?
- Handle heterogenity in naming and addressing?
- Connection-centric vs. data-centric?
- Support discovery and rendevous services?

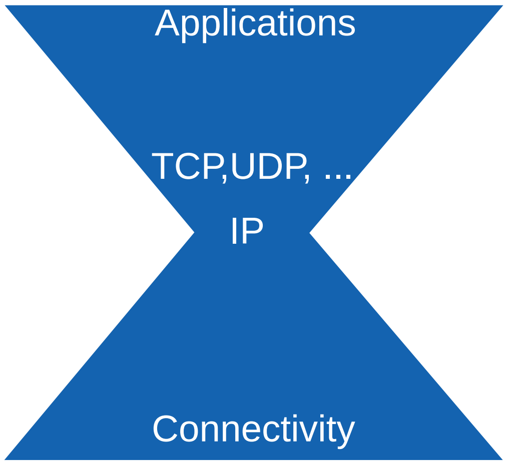
About Interfaces

- What degree of transparency?
- What interaction between data/control/management plane?
- What about peering interfaces? Role of business issues?

Conclusions and Outlook

Heading towards a Post-TCP Era?

- **Transport layer functions are a key challenge for any "Future Internet"**
- **Evolution vs. revolution?**
  - Short-term: TCP’s shortcomings become more and more evident
  - Mid-term: Role and functions of routers to be rediscussed
  - Long-term: We will design Post-TCP once we know the "Future Internet"
- **However ...**



Heading towards a Post-TCP Era?

- **Transport layer functions are a key challenge for any "Future Internet"**
- **Evolution vs. revolution?**
  - Short-term: TCP’s shortcomings become more and more evident
  - Mid-term: Role and functions of routers to be rediscussed
  - Long-term: We will design Post-TCP once we know the "Future Internet"
- **However ...**