Network Support for Grid Computing

Recent Research Work and Plans at the University of Innsbruck

Michael Welzl http://www.welzl.at

DPS NSG Team http://dps.uibk.ac.at/nsg
Institute of Computer Science
University of Innsbruck

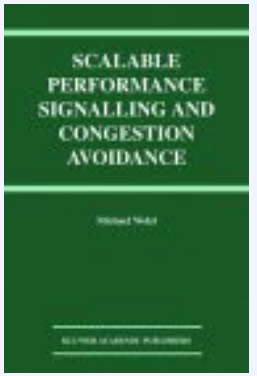
University of Trento
14 October, 2005

Outline

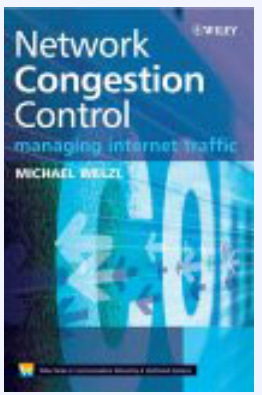
- Introduction; the NSG Team at the University of Innsbruck
- Problem scope
- Proposed solutions
  - Example 1: Network Measurement
  - Example 2: QoS / High Performance Communication
- Conclusion

Who am I?

- A real globetrotter :) Innsbruck ⇒ Linz ⇒ Innsbruck



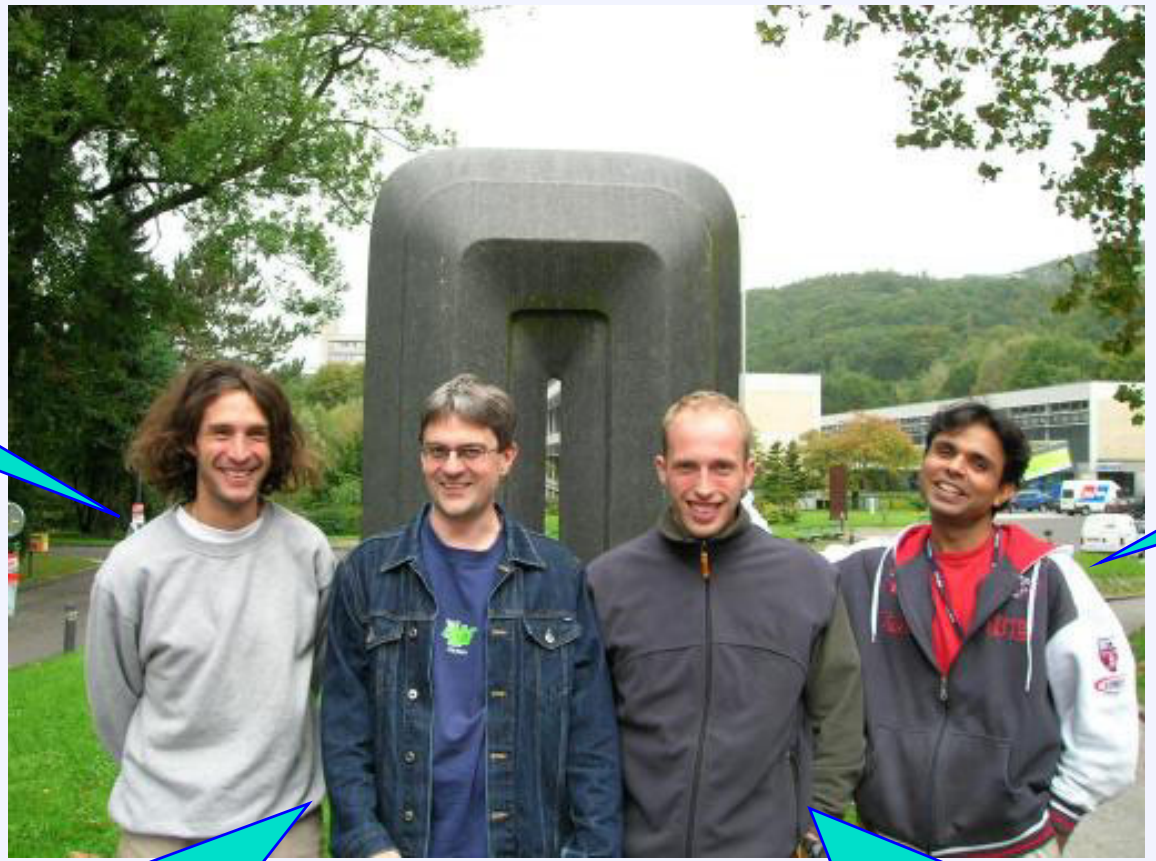
- Ph.D. in Darmstadt (Max Mühlhäuser + Jon Crowcroft)
  - Defense passed with distinction November 2002
  - Published as Kluwer (now Springer) book "Scalable Performance Signalling and Congestion Avoidance", August 2003
  - Received "Best Dissertation Award 2004" from German GI/ITG KuVS



- Network Congestion Control: Managing Internet Traffic
  - John Wiley & Sons, July 2005
  - The first introductory book on this topic

- Research notion: one-size-fits-all TCP + IP not optimal
  - Main interest: tailor network technology to work with
    - heterogeneous infrastructure (e.g. high-speed or noisy links, with mobility)
    - heterogeneous applications (e.g. streaming media, signaling, Grid)

The NSG team



Werner Heiss
Tyrolean Science Fund

Murtaza Yousaf
Scholarship from Pakistani Government

Michael Welzl
Institute of Computer Science

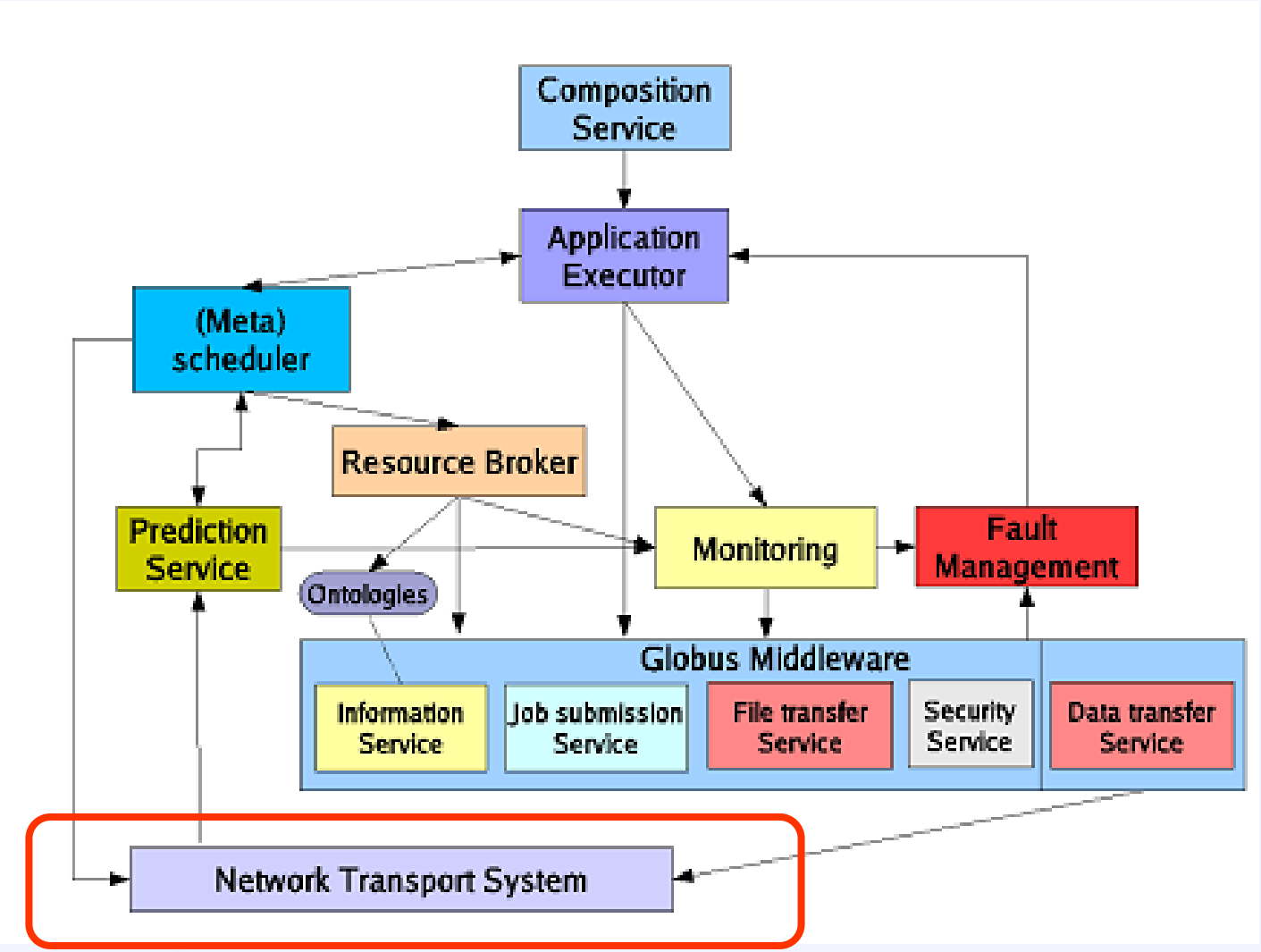
Sven Hessler
Austrian Science Fund (FWF)

Not shown - starting November 2005: **Kashif Munir**
Scholarship from Pakistani Government

NSG activities

- Research topics: Grid = main focus
  - Tailored network technology in support of Grid applications
    - Congestion Control
    - Quality of Service (QoS)
    - Transport Protocols
    - Network Measurement and Prediction
    - Middleware Communication
  - Also other aspects of networking (e.g. multimedia communication)
- Teaching: we cover the networking courses at UIBK
- Collaborations: Grid related results are...
  - contributed to standards via GHPN-RG of Global Grid Forum (GGF)
  - embedded in the Workflow system developed by the DPS Group at UIBK

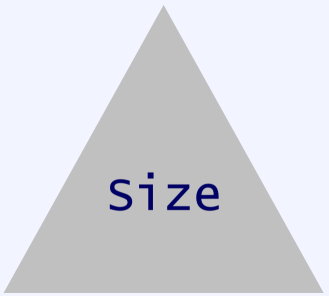
The DPS Grid Workflow Application Execution Environment



Problem scope

Scope

- Grid history: parallel processing at a growing scale
  - Parallel CPU architectures
  - Multiprocessor machines
  - Clusters
  - ("Massively Distributed") computers on the Internet



- Traditional goal: processing power
  - Grid people = parallel people; thus, goal has not changed much

Reasonable to focus on this.

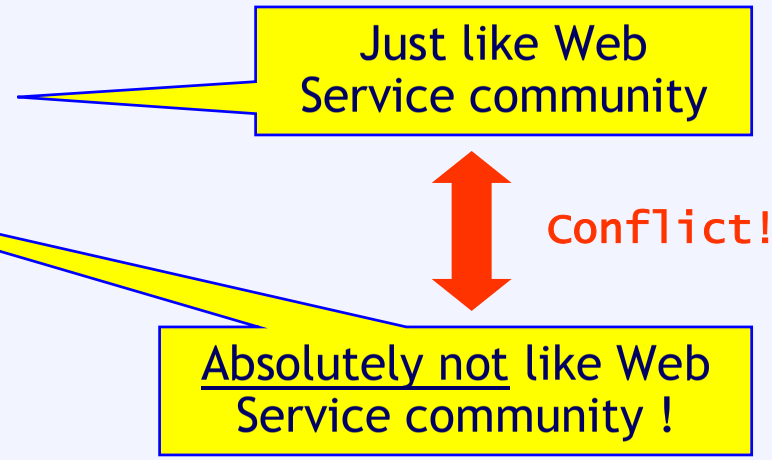
- Broader definition ("resource sharing")
  - reasonable - e.g., computers also have harddisks :-)
  - New research areas / buzzwords: Wireless Grid, DataGrid, Pervasive Grid, [*this space reserved for your favorite research area*] Grid
  - sometimes perhaps a little *too* broad, e.g., "P2P Working Group" is now part of the Global Grid Forum

Grid requirements

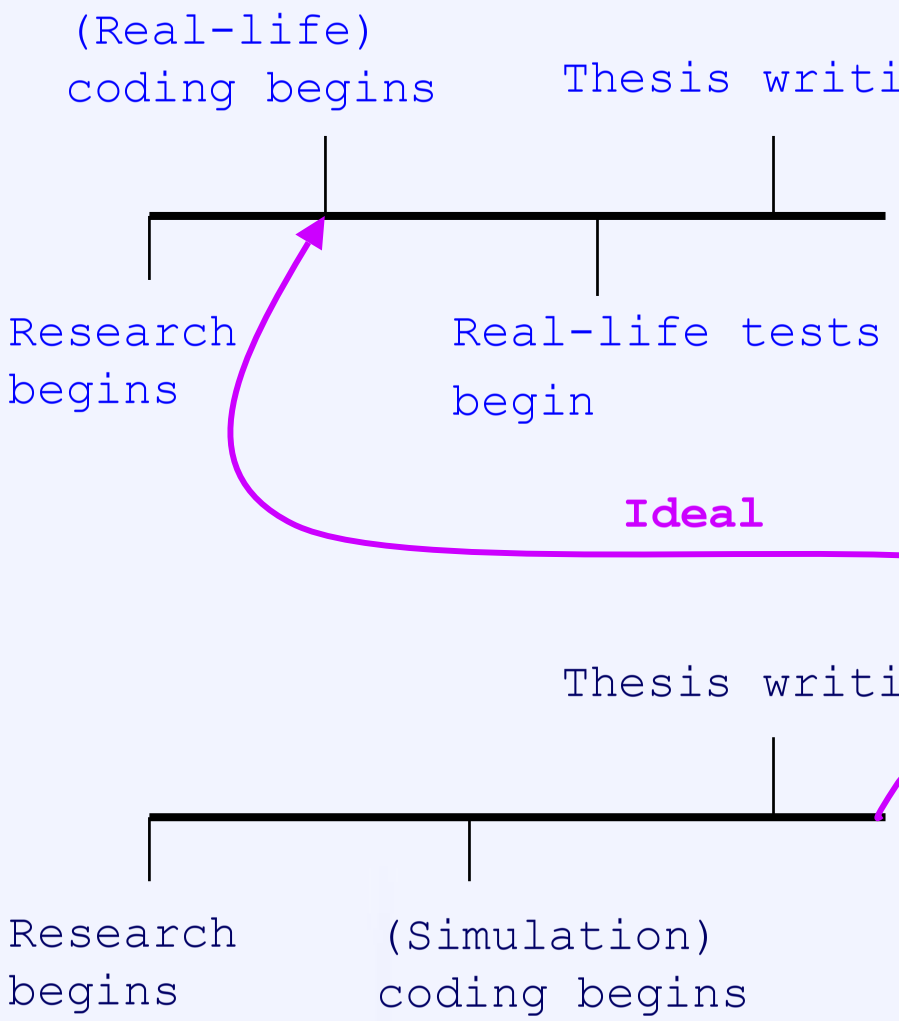
- Efficiency + ease of use !
  - Programmer should not worry about the Grid
  - Ideally, applications should automatically be distributed
- Underlying system has to deal with
  - Error management
  - Authentification, Authorization and Accounting (AAA)
  - Efficient Scheduling / Load Balancing
  - Resource finding and brokerage
  - Naming
  - Resource access and monitoring
- No problem: we do it all - in Middleware
- de facto standard: “Globus Toolkit“
  - installation of GT3 in our high performance system: 1 1/2 hours or so...
  - yes, it truly does it all :) 1000s of addons - GridFTP, MDS, NWS, GRAM, ..

Problem: How Grid folks see the Internet

- Abstraction - simply use what is available
  - still: performance = main goal
- Existing transport system
  (TCP/IP + Routing + ..) works well
- QoS makes things better, the Grid needs it!
  - we now have a chance for that, thanks to IPv6
- Quote from a paper review:
  "In fact, any solution that requires changing the TCP/IP protocol stack is practically unapplicable to real-world scenarios, (..)."
- How to change this view: GGF GHPN-RG
  - documents such as "net issues with grids", "overview of transport protocols"
  - also, some EU projects, workshops, ..



A time-to-market issue



Typical Grid project

Result: thesis + running code; tests in collaboration with different research areas

Typical Network project

Result: thesis + simulation code; perhaps early real-life prototype (if students did well)

Grid-network peculiarities

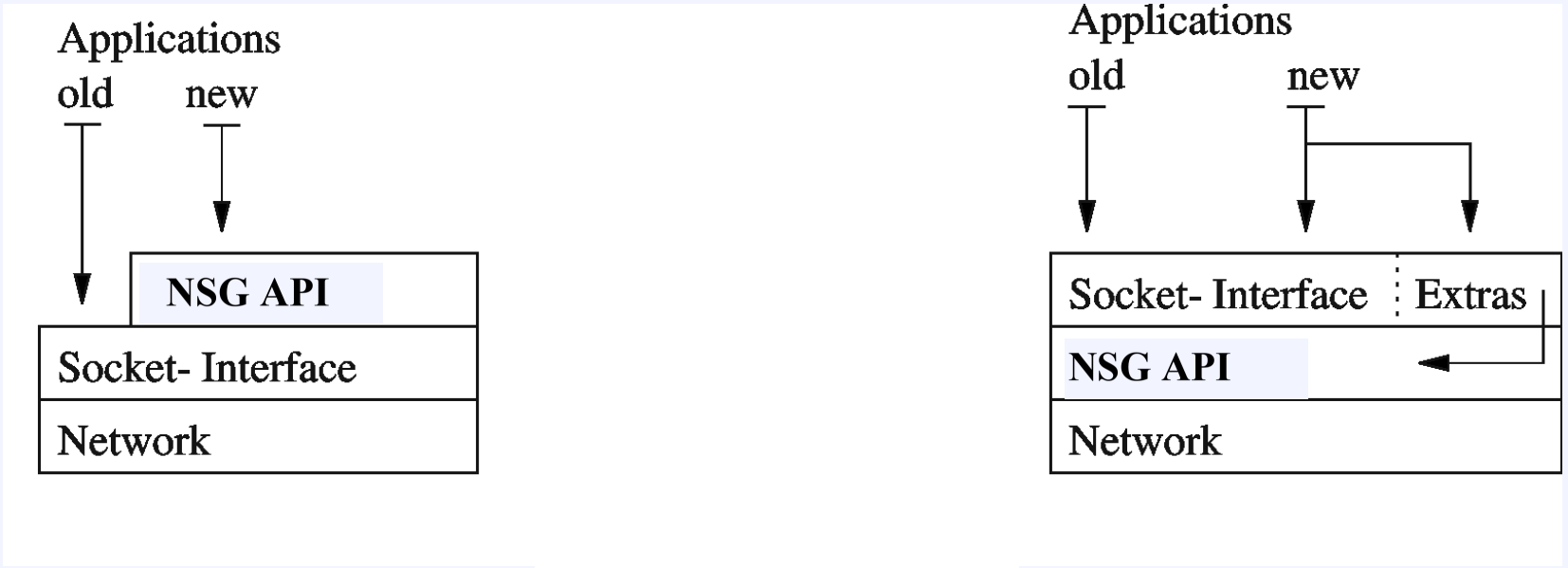
- Special behavior
  - Predictable traffic pattern - this is totally new to the Internet!
    - Web: users create traffic
    - FTP download: starts ... ends
    - Streaming video: either CBR or depends on content! (head movement, ..)
    - Could be exploited by congestion control mechanisms
  - Distinction: Bulk data transfer (e.g. GridFTP) vs. control messages (e.g. SOAP)
  - File transfers are often "pushed" and not "pulled"

- Special requirements
  - Predictions
  - Latency bounds, bandwidth guarantees ("advance reservation") => QoS

- Distributed system, active for a certain duration
  - Can use distributed overlay network strategies (done in P2P systems!)
    - Multicast
    - P2P paradigm: "do work for others to enhance the total system" (for your own good) - e.g. transcoding, act as a PEP, ..
  - Can exploit highly sophisticated network measurements!
    - some take a long time, some require a distributed infrastructure

Some issues: application interface...

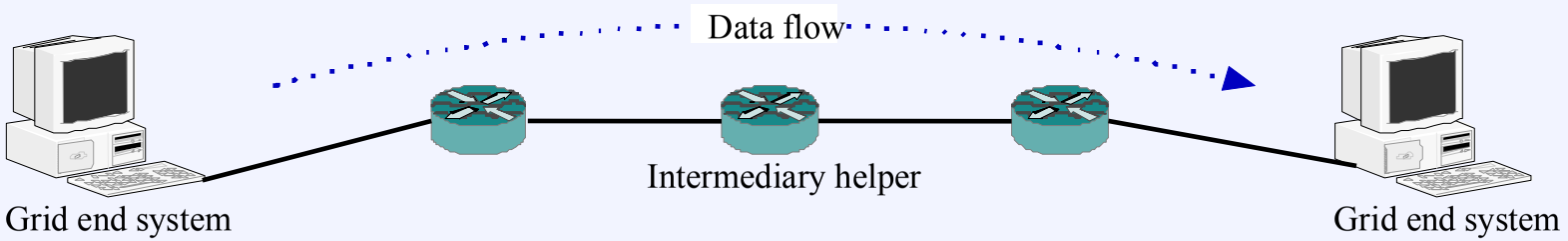
- How to specify properties and requirements
  - Should be simple and flexible - use QoS specification languages?
  - Should applications be aware of this?
    ⇒ Trade-off between service granularity and transparency!



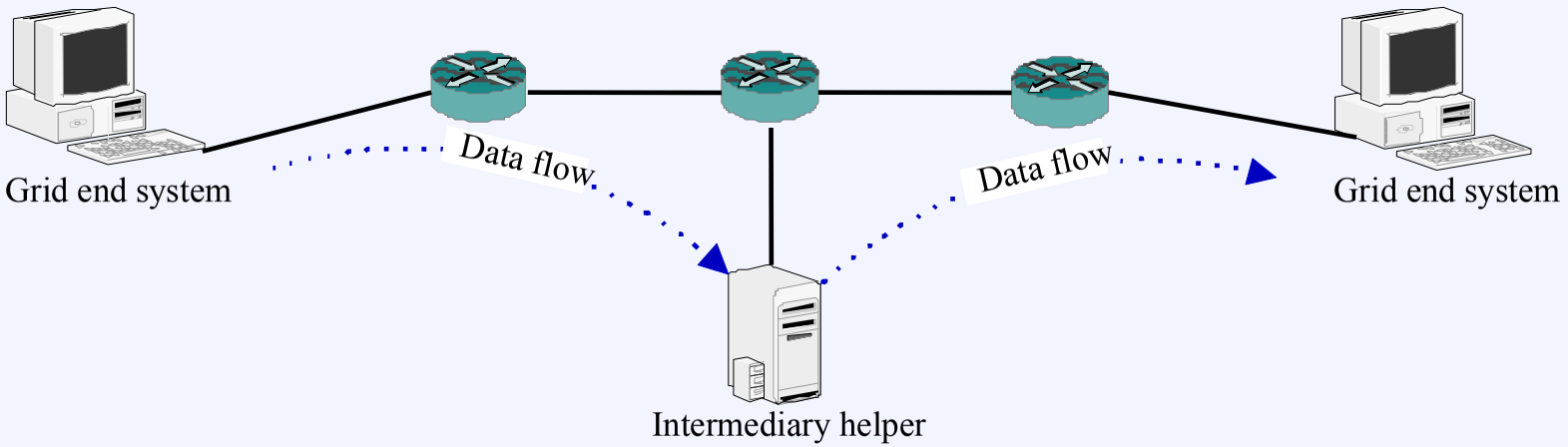
Traditional method

Our approach

... and peer awareness



(a) Traditional PEP



(b) NSG PEP

Proposed solutions

Example 1: Network Measurement

Measuring the network

- When you measure, you measure the past
  - predictions / estimations with a ?? % chance of success
- When you measure, you change the system
  - e.g., high-rate-UDP vs. TCP: non-intrusiveness really important
- Measurements yield no guarantees
  - Internet traffic = result of user behavior!
- Research often carried out in controllable, isolated environments
  - Here, measurements are different from measurements in the 'net
  - Field trials are a necessary extra when you know that something works

NWS: The Network Weather Service

- Distributed system consisting of
  - Name Server (boring)
  - Sensor - actual measurement instance, regularly stores values in.....
  - Persistent State
  - Forecaster (calculations based on data in Persistent State)

- Interesting parts:

  - Sensor
    Measured resources: availableCpu, bandwidthTcp, connectTimeTcp, currentCpu, freeDisk, freeMemory, latencyTcp

  - Forecaster
    Apply different models for prediction, compare with actual measurement data, choose best match

Duration of a long TCP transfer

RTT of a small message

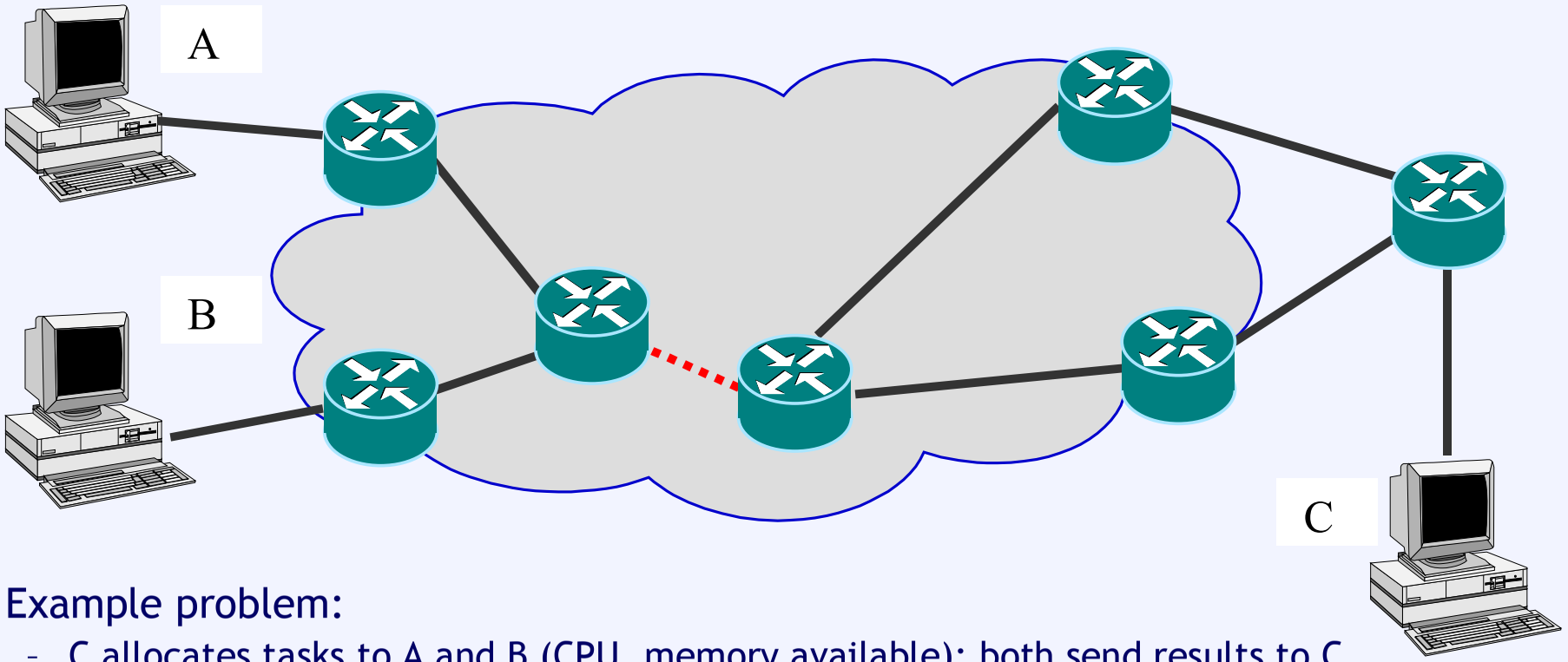
NWS critique

- Architecture (splitting between sensors, forecaster etc.) seems reasonable; open source ⇒ consider integrating new work in NWS

- Sensor
  - active measurements even though non-intrusiveness was an important design goal - does not passively monitor TCP (i.e. ignores available data!)
  - strange methodology:
    (Large message throughput) *"Empirically, we have observed that a message size of 64K bytes (..) yields meaningful results"*
  - ignores packet size (= measurement granularity!) and path characteristics
  - trivial method - much more sophisticated methods available (e.g. packet pair - later!)
  - point-to-point measurements: distributed infrastructure not taken into account

- Forecaster
  - relies on these weird measurements, where we don't know much about the distribution (but we do know some things about other measurement methods!)
  - uses quite trivial models (but they may in fact suffice...)

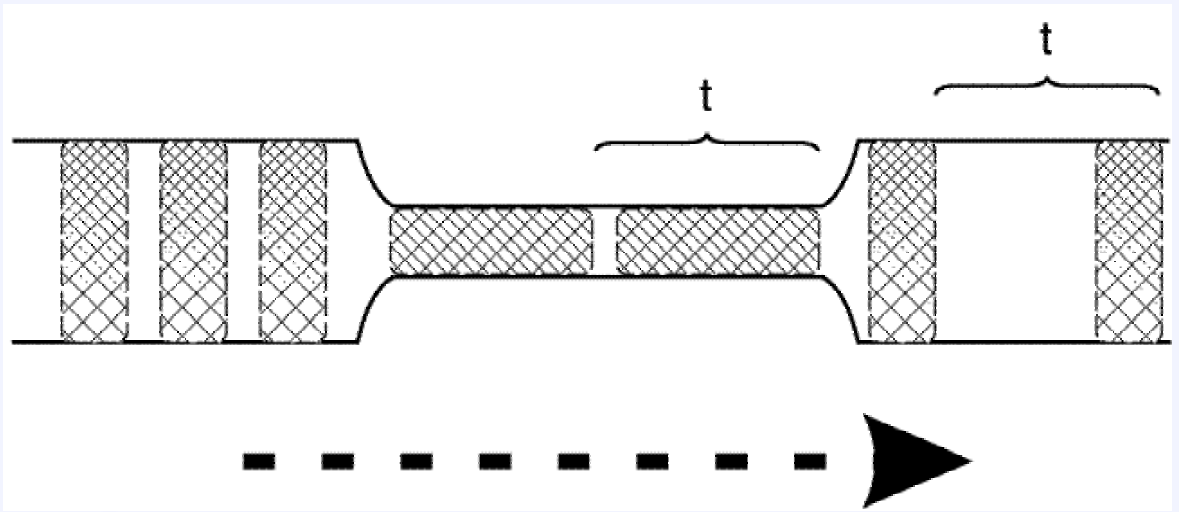
Exploiting the Distributed Infrastructure



- Example problem:
  - C allocates tasks to A and B (CPU, memory available); both send results to C
  - B hinders A - task of B should have been kept at C!
- Path changes are rare - thus, possible to detect potential problem in advance
  - generate test messages from A, B to C - identify signature from B in A's traffic
- Another issue in this scenario: how valid is a prediction that A obtains if the measurement / prediction system does not know about the shared bottleneck?

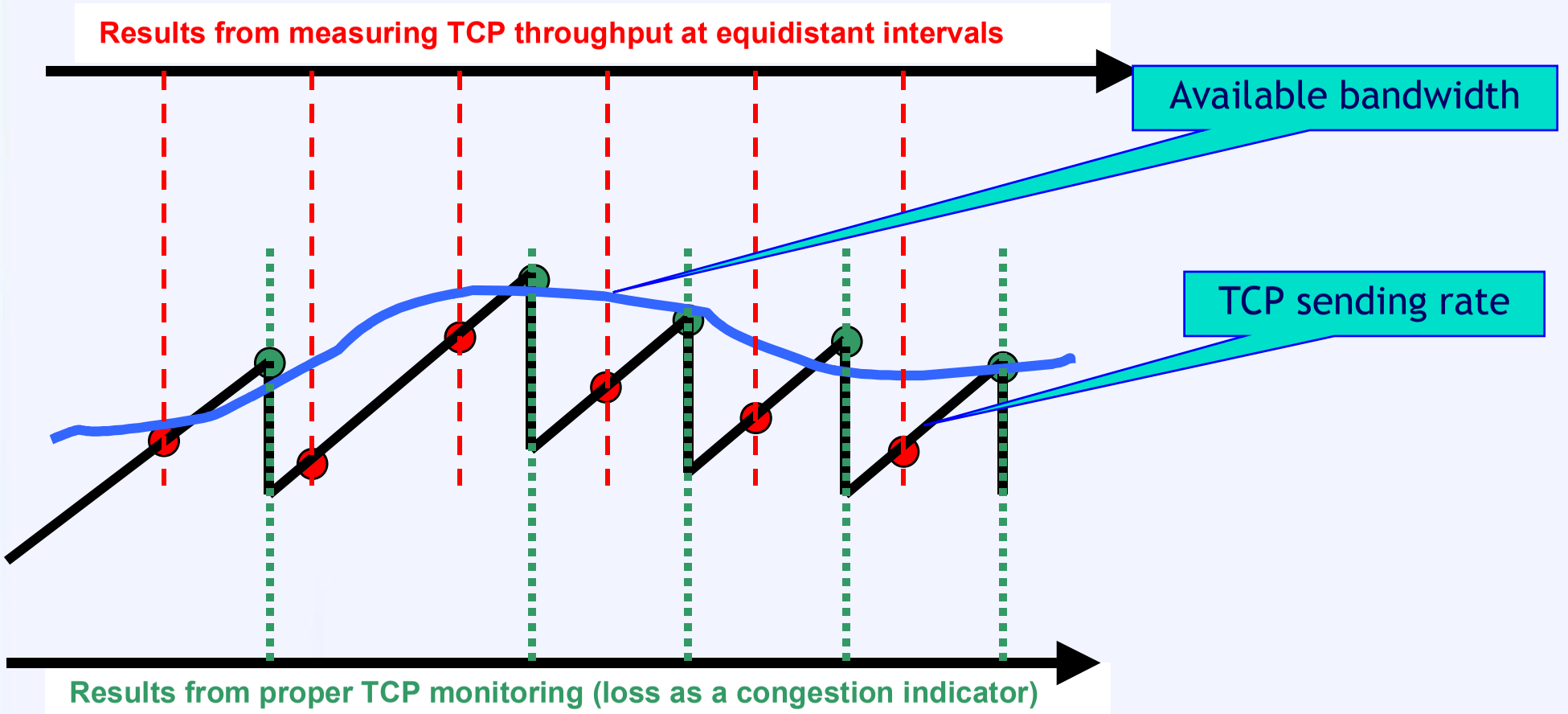
Exploiting longevity

- Time scale of traffic fluctuations < time scale of path changes
  ⇒ knowledge of link capacities may be more useful than traffic estimate
- Underlying technique: packet pair
  - send two packets p1 and p2 in a row; high probability that p2 is enqueued exactly behind p1 at bottleneck
  - at receiver: calculate bottleneck bandwidth via time between p1 and p2
  - minimize error via multiple probes
  - TCP with "Delayed ACK" receiver automatically sends packet pairs
    ⇒ passive TCP receiver monitoring is quite good!



Traffic prediction by monitoring TCP

- TCP propagates bottleneck self-similarity to end systems! ("samples bandwidth")
- Automatic prediction? Complex, but possible, I think - e.g.:
  Yantai Shu, Zhigang Jin, Jidong Wang, Oliver W. W. Yang: Prediction-Based Admission Control Using FARIMA Models. ICC (3) 2000: 1325-1329



Example 2: QoS / High Performance Communication

QoS (reservation of network connections),
high performance communication for the Grid

QoS: the state-of-the-art :-(

Papers from SIGCOMM'03 RIPQOS Workshop: "Why do we care, what have we learned?"

- QoS`s Downfall: At the bottom, or not at all! Jon Crowcroft, Steven Hand, Richard Mortier,Timothy Roscoe, Andrew Warfield
- Failure to Thrive: QoS and the Culture of Operational Networking Gregory Bell
- Beyond Technology: The Missing Pieces for QoS Success Carlos Macian, Lars Burgstahler, Wolfgang Payer, Sascha Junghans, Christian Hauser, Juergen Jaehnert
- Deployment Experience with Differentiated Services Bruce Davie
- Quality of Service and Denial of Service Stanislav Shalunov, Benjamin Teitelbaum
- Networked games --- a QoS-sensitive application for QoS-insensitive users? Tristan Henderson, Saleem Bhatti
- What QoS Research Hasn`t Understood About Risk Ben Teitelbaum, Stanislav Shalunov
- Internet Service Differentiation using Transport Options:the case for policy-aware congestion control Panos Gevros

Key reasons for QoS failure

- Required participation of end users and all intermediate ISPs
  - "normal" Internet users want Internet-wide QoS, or no QoS at all
  - In a Grid, a "virtual team" wants QoS between its nodes
  - Members of the team share the same ISPs - flow of $$$ is possible
- Technical inability to provision individual (per-flow) QoS
  - "normal" Internet users
    - unlimited number of flows come and go at any time
    - heterogeneous traffic mix
  - Grid users
    - number of members in a "virtual team" may be limited
    - clear distinction between bulk data transfer and SOAP messages
    - appearance of flows controlled by machines, not humans
- $\Rightarrow$ QoS could work for the Grid !

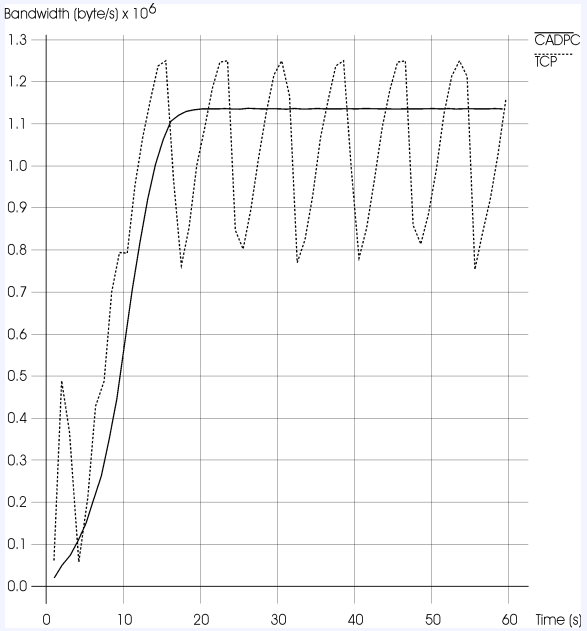
High Performance Communication

- Often, large files are transmitted in Grids, and large capacity links are bought. Thus, two goals:
  - efficient capacity usage: desirable to achieve 1 gbit/s across 1 gbit/s link
  - fairness: if 10 flows share a link, all 10 flows should get their share = efficiency: e.g., GridFTP should not block SOAP messages
- Standard since 1980's: Transmission Control Protocol (TCP)
  - roughly: additively increase rate until bottleneck queue grows, packet drop occurs (congestion caused!), then halve rate $\Rightarrow$ sawtooth
  - works poorly in today's environments: high speed links, "long fat pipes", noisy (wireless) links, ..
  - gradual (small + downward compatible) improvements standardized
- Many alternatives proposed, often in Grid context - but hard to deploy because of TCP-friendliness

QoS + congestion control = solution!

- Idea: use traditional coarse-grain QoS mechanism (DiffServ) to differentiate between high-performance bulk data transfer and everything else (= SOAP etc. over TCP)

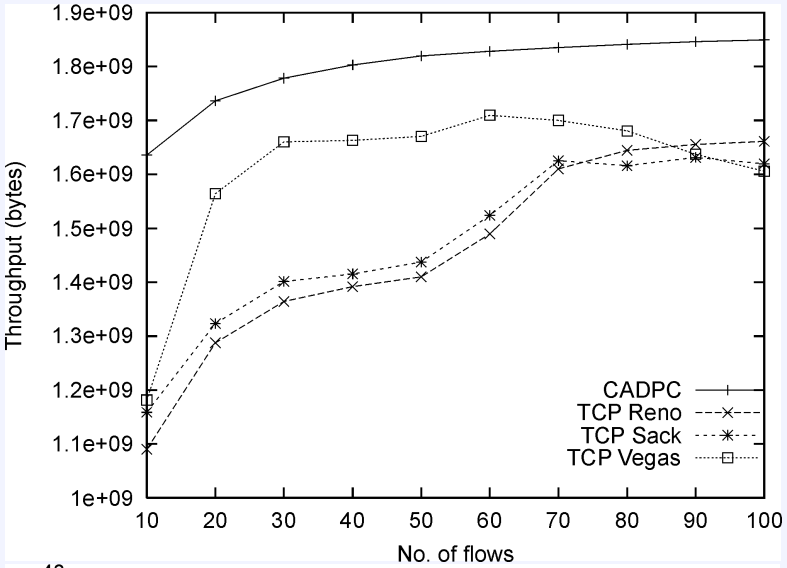
- Isolated long-living data transfer = requirements for CADPC/PTP
  - This is the best congestion control mechanism
  - because I developed it for my Ph.D. thesis :-)

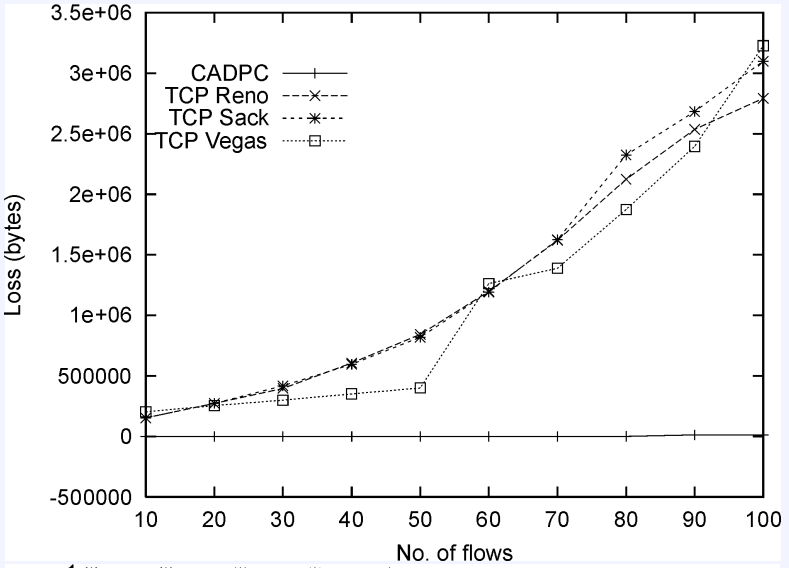


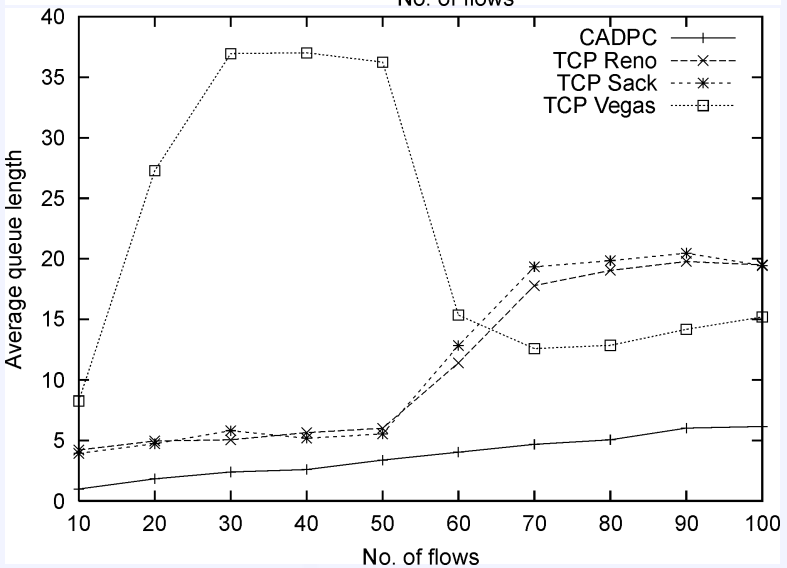
- Some properties:
  - low loss, high throughput
  - predictable and stable rate, only depends on capacity and number of flows

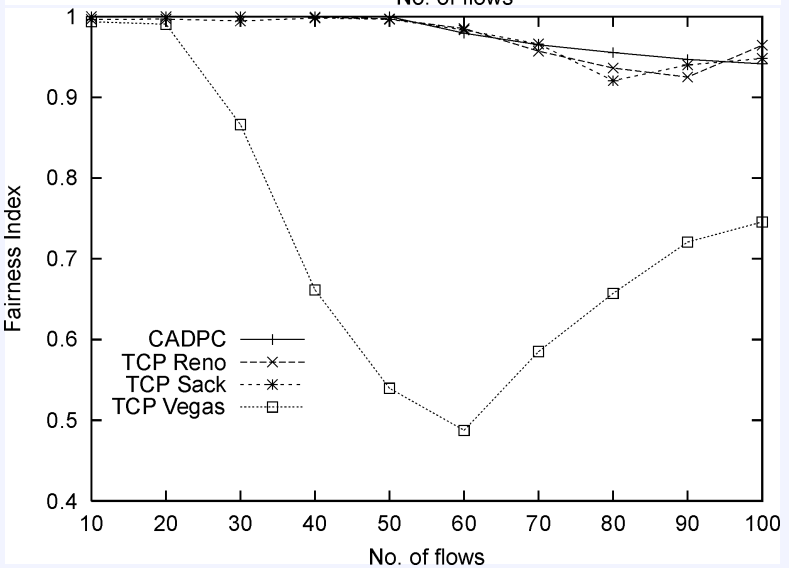
- Disadvantage: requires router support
  - may be realistic in a Grid!

CADPC vs. 3 TCP(+ECN) flavors

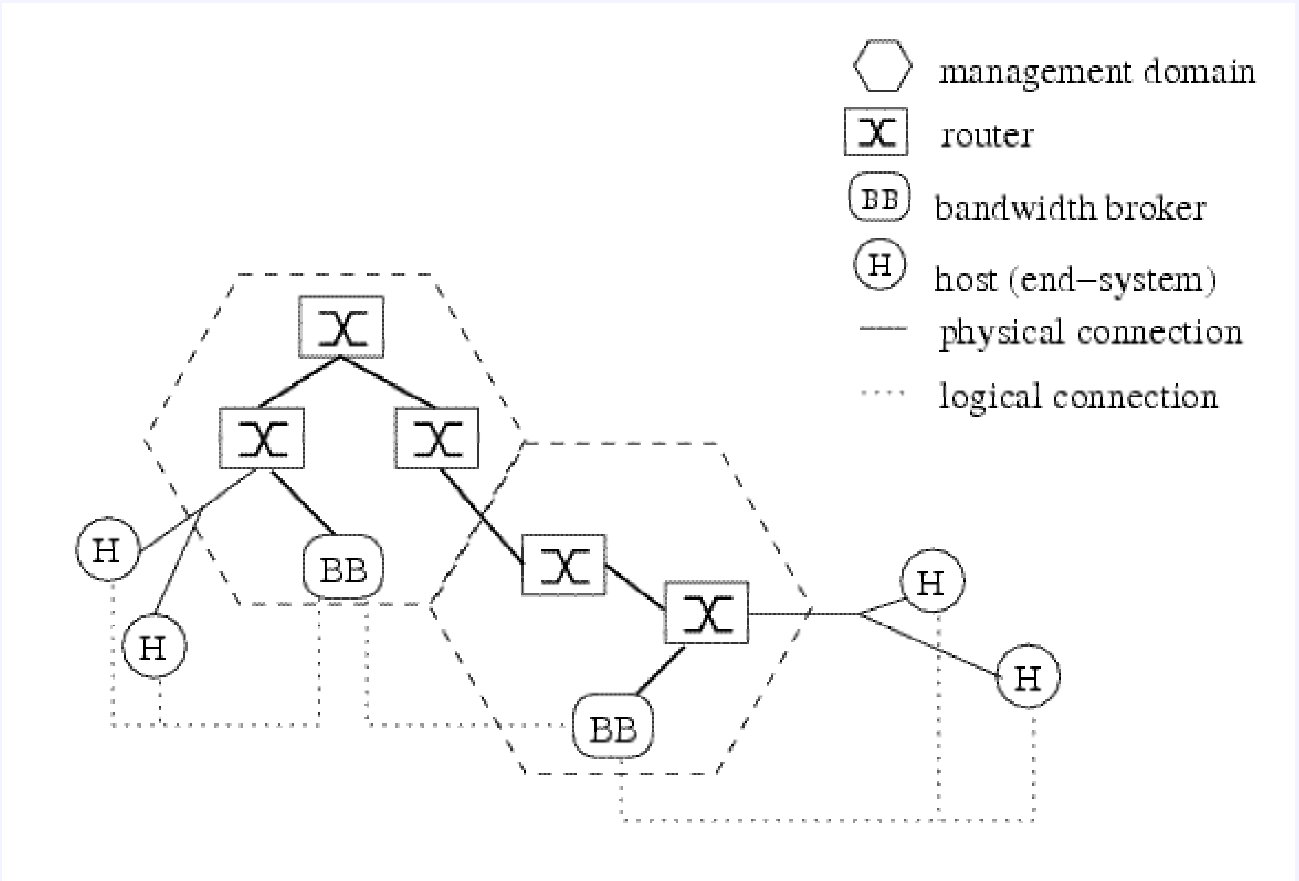








NSG Grid QoS architecture



- Mandate CADPC/PTP usage for bulk data transfer
- Resource reservation via admission control
  - Bandwidth broker decides what enters the network
  - Flow differentiation: simply allow a flow to act like *n* flows!

Conclusion

Conclusion

- Grid applications show special requirements and properties from a network perspective
  - and it is reasonable to develop tailored network technology for them.
- There is another class of such applications...
- Multimedia.
- For multimedia applications, an immense number of network enhancements (even IETF standards) exist.
- For the Grid, there is nothing.
- This is a research gap; let's fill it together!

Thank you!

Questions?

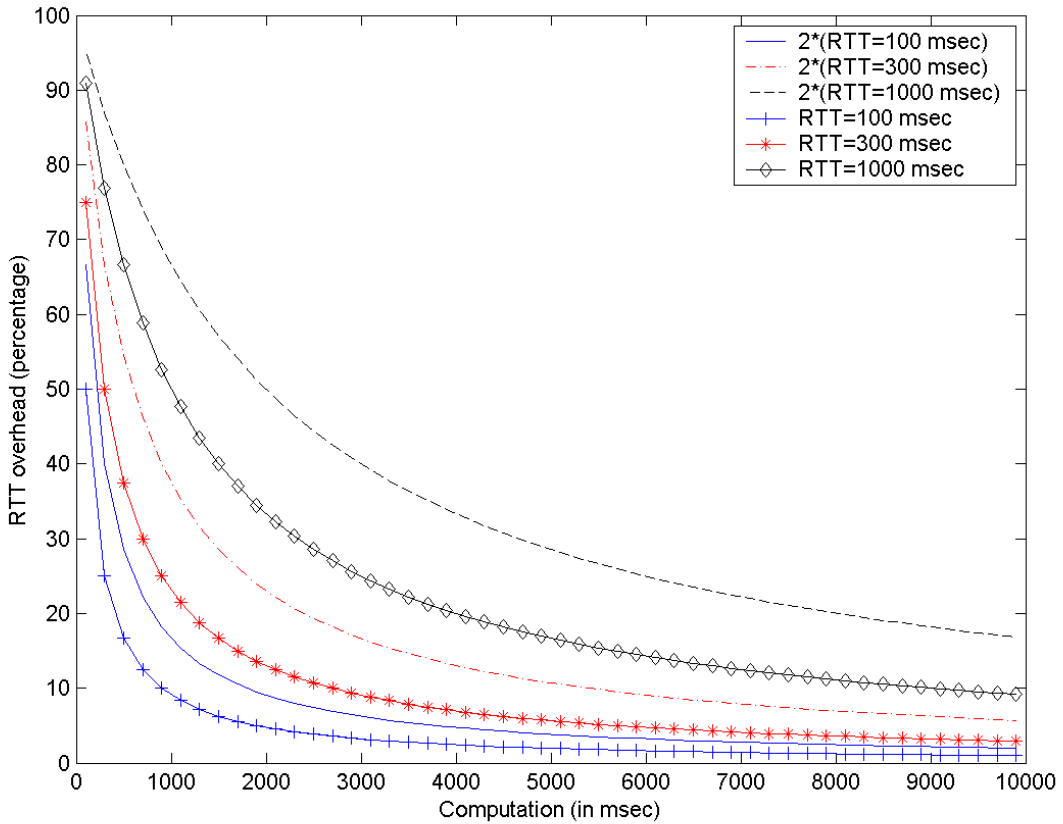
So they want efficiency...

- It's a large stack: Globus/SOAP/HTTP/TCP/IP

- "Hello World" Grid Service call (including service creation) with GT3 (no security features etc.)
  - 60 packets exchanged, at least 6 RTTs (mainly TCP connection handling)
  - each additional call: another 14 packets (at least 2 RTTs)
  - MPI is better (keeps connections open), but is hardly used outside clusters

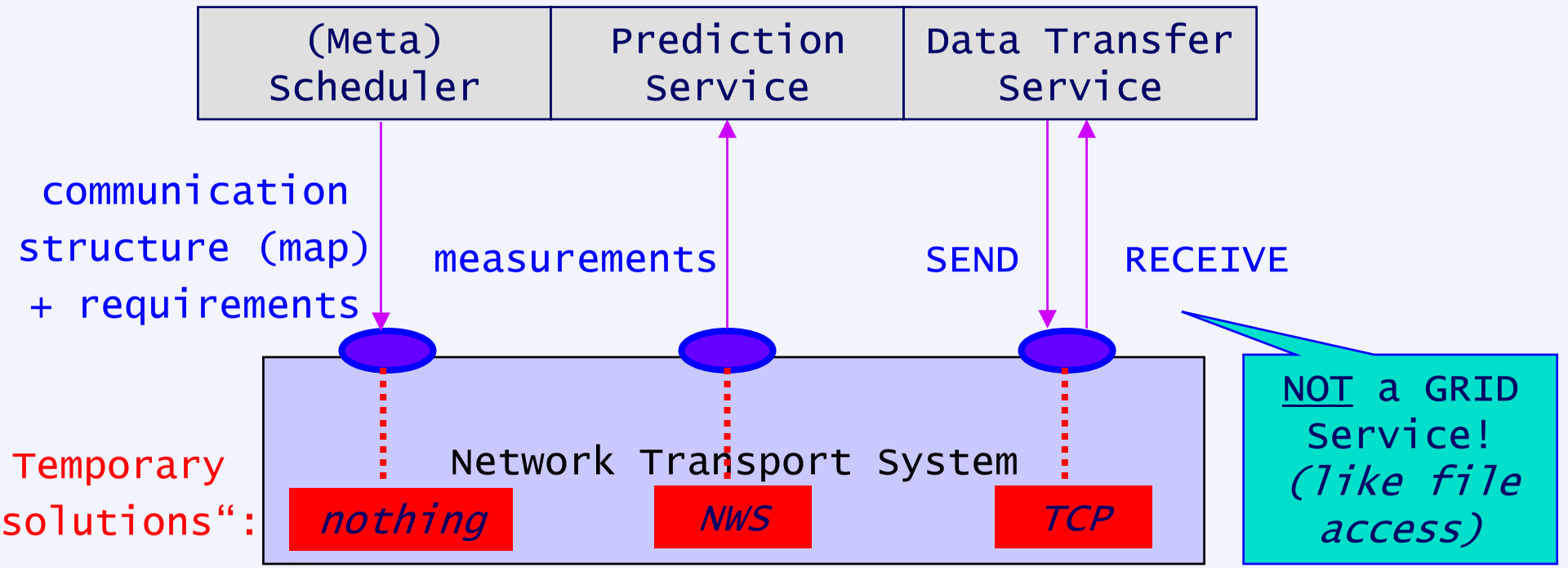
- Data transmission: 2 "clean" methods in GT3
  1. Parameters of a Grid Service call: SOAP/HTTP encoding :(
  2. GridFTP: common choice for "bulk data transfer"
     - like FTP++ ... multiple TCP connections, remote FTP invocation
     - but yes, it moves files only! Thus, data go mem-file-net-file-mem !

Room for enhancements!


2*(RTT=100 msec)
2*(RTT=300 msec)
2*(RTT=1000 msec)
RTT=100 msec
RTT=300 msec
RTT=1000 msec
RTT overhead (percentage)
Computation (in msec)
0
10
20
30
40
50
60
70
80
90
100
0
1000
2000
3000
4000
5000
6000
7000
8000
9000
10000

How to use the Network Transport System

- Specify communication structure (map) + requirements
- Obtain good measurements
- Utilize efficient communication service *(no guarantees for now!)*



Adaptation Layer: architecture