

Lambda-Grid developments

History - Present - Future

Cees de Laat

University of Amsterdam



Contents

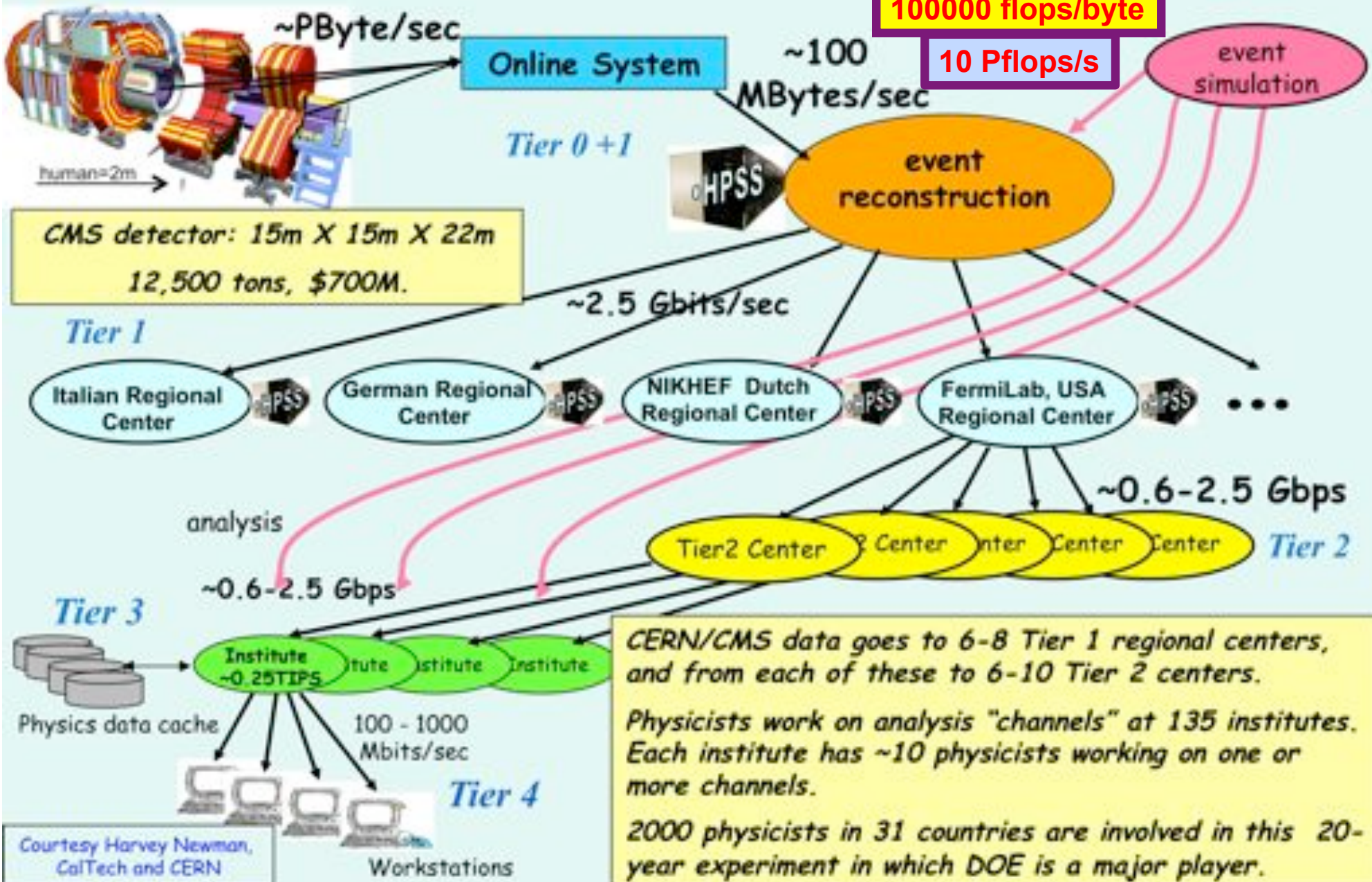
1. The need for hybrid networking
2. StarPlane; a grid controlled photonic network
3. Cross Domain Authorization using Tokens
4. RDF/Network Description Language
5. Tera-networking
6. Programmable networks





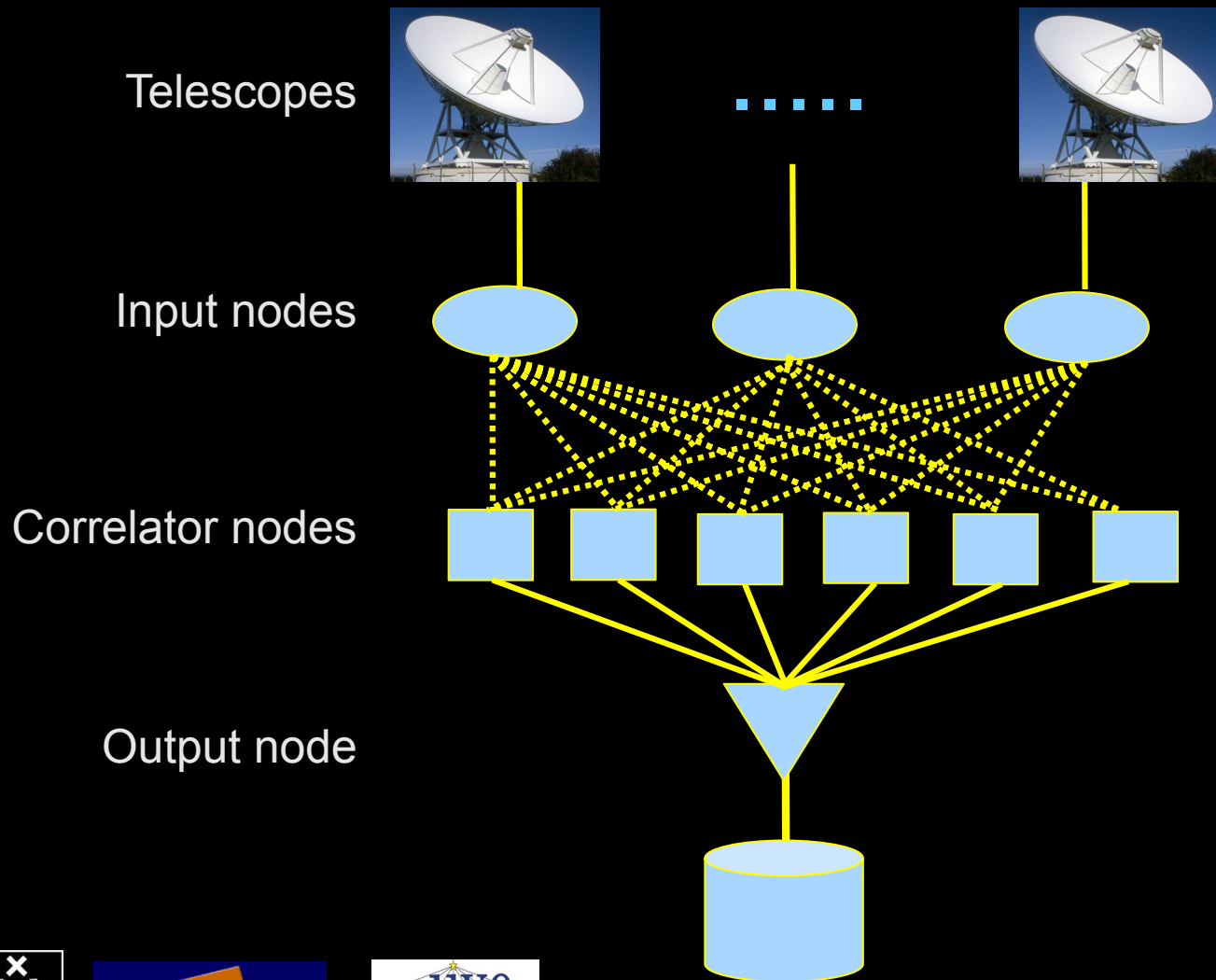
LHC Data Grid Hierarchy

CMS as example, Atlas is similar



The SCARIE project

SCARIE: a research project to create a Software Correlator for e-VLBI.
VLBI Correlation: signal processing technique to get high precision image from spatially distributed radio-telescope.



To equal the hardware correlator we need:

16 streams of 1Gbps

16 * 1Gbps of data

2 Tflops CPU power

2 TFlop / 16 Gbps =

1000 flops/byte

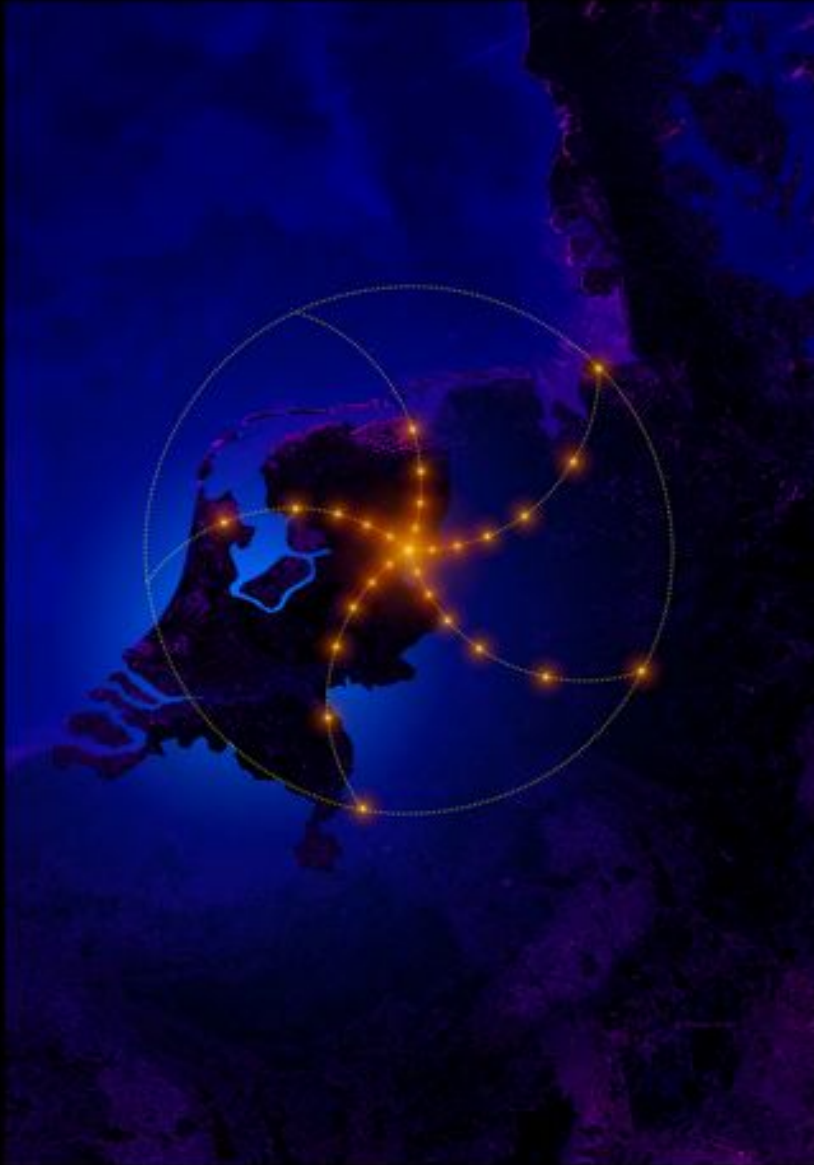
0.1 Pflops/s

THIS IS A DATA FLOW PROBLEM !!!



LOFAR as a Sensor Network

20 flops/byte



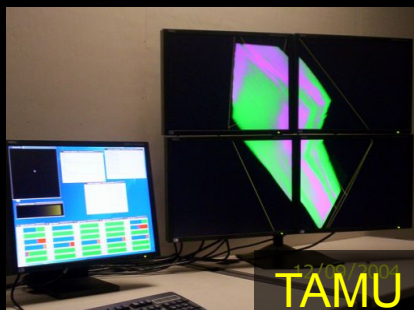
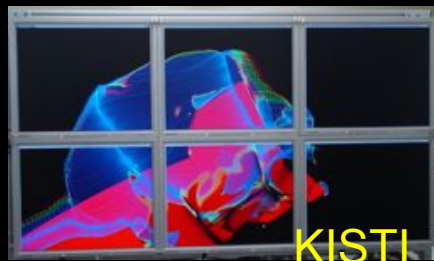
– LOFAR is a large distributed research infrastructure:

2 Tflops/s

- Astronomy:
 - >100 phased array stations
 - Combined in aperture synthesis array
 - 13,000 small “LF” antennas
 - 13,000 small “HF” tiles
- Geophysics:
 - 18 vibration sensors per station
 - Infrasound detector per station
- >20 Tbit/s generated digitally
- >40 Tflop/s supercomputer
- innovative software systems
 - new calibration approaches
 - full distributed control
 - VO and Grid integration
 - datamining and visualisation



US and International OptIPortal Sites



Real time, multiple 10 Gb/s



The "Dead Cat" demo

1 Mflops/byte

Real time issue



SC2004,
Pittsburgh,
Nov. 6 to 12, 2004
iGrid2005,
San Diego,
sept. 2005

Many thanks to:
AMC
SARA
GigaPort
UvA/AIR
Silicon Graphics,
Inc.
Zoölogisch Museum

M. Scarpa, R.G. Belleman, P.M.A. Sloot and C.T.A.M. de Laat, "Highly Interactive Distributed Visualization",
iGrid2005 special issue, Future Generation Computer Systems, volume 22 issue 8, pp. 896-900 (2006).





IJKDIJK

300000 * 60 kb/s * 2 sensors (microphones) to cover all Dutch dikes



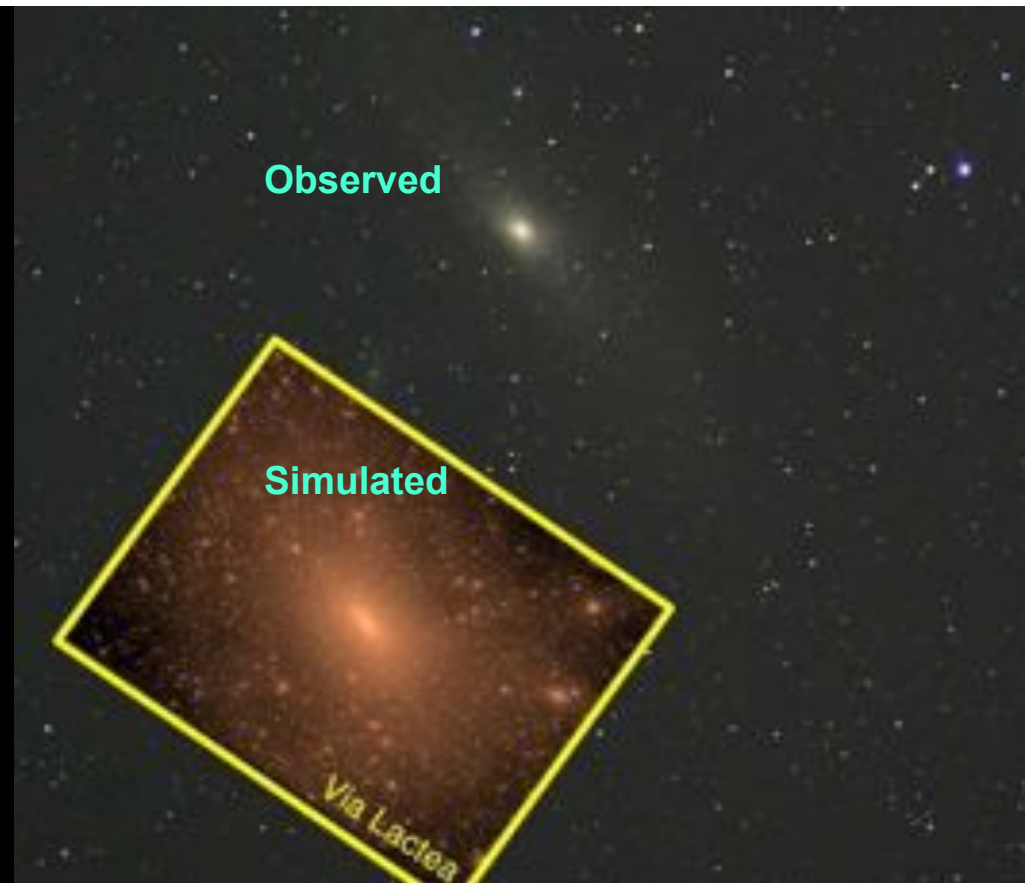
Sensor grid: instrument the dikes

First controlled breach occurred on sept 27th '08:



CosmoGrid

- Motivation:
previous simulations found >100 times more substructure than is observed!
- Simulate large structure formation in the Universe
 - Dark Energy (cosmological constant)
 - Dark Matter (particles)
- Method: Cosmological N -body code
- Computation: Intercontinental SuperComputer Grid



The hardware setup

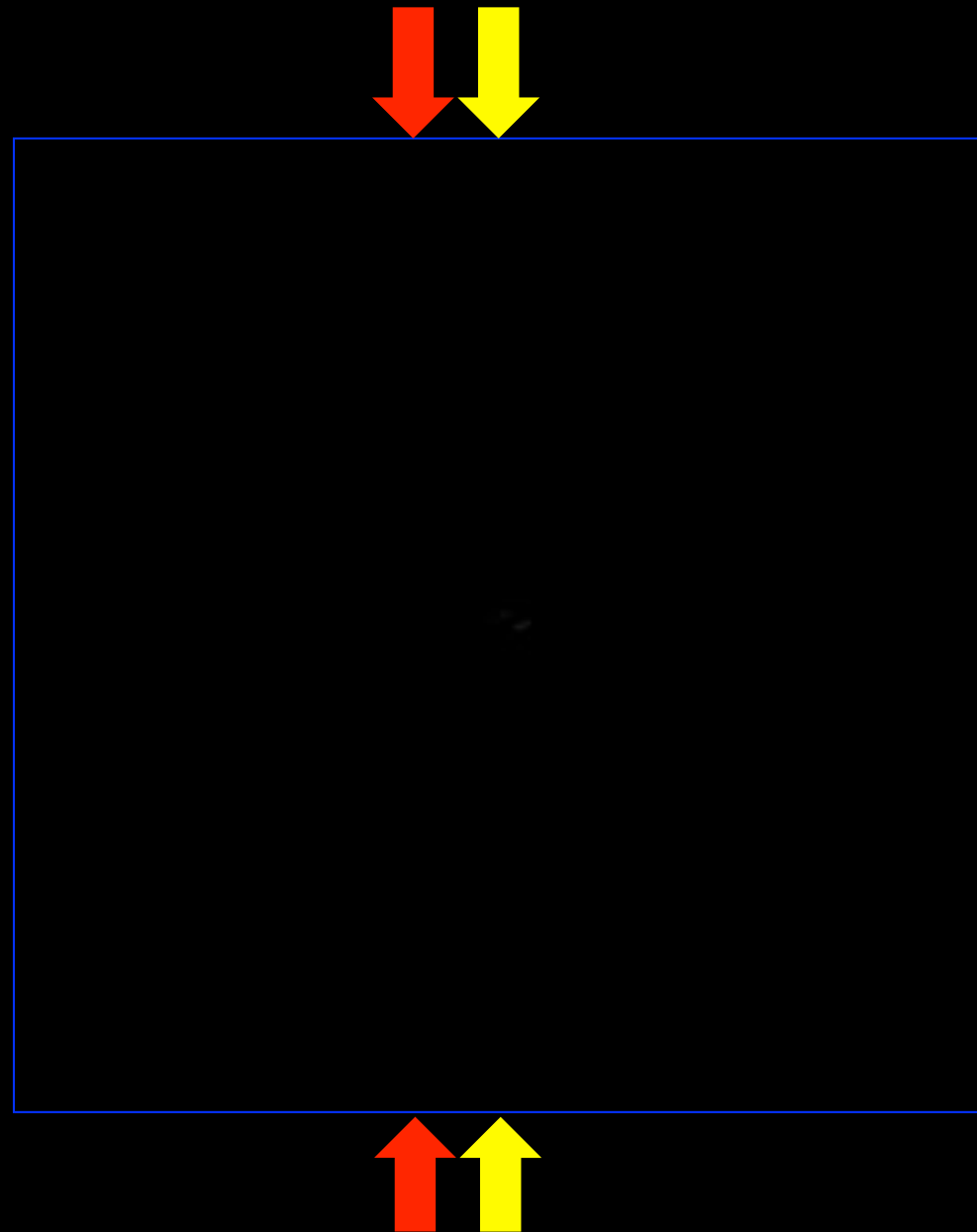
10 Mflops/byte

1 Eflops/s

- 2 supercomputers :
 - 1 in Amsterdam (60Tflops Power6 @ SARA)
 - 1 in Tokyo (30Tflops Cray XD0-4 @ CFCA)
- Both computers are connected via an intercontinental optical 10 Gbit/s network



Auto-balancing Supers



CosmoGrid

Supercomputing Grid across Continents and Oceans

And yes, it works!

Application

We originally developed MPWide to manage the long-distance message passing in the CosmoGrid¹ project. This is a large-scale cosmological project whose primary goal is to perform a dark matter simulation using supercomputers on two continents.

In this simulation, we use the cosmological Λ Cold Dark Matter model² to simulate the dark matter particles using a parallel tree/particle-mesh N-body integrator, TreePM³. This requires relatively little communication between different sites after each timestep. This integrator calculates the dynamical evolution of 2048³ (8.5 billion) particles. More information about the parameters used and the scientific rationale can be found in ⁴.

The integrator can be run as a single MPI application, or as two separately launched MPI applications on different supercomputers.

¹ Portegies Zwart et al., 2009, IEEE Computer (submitted)

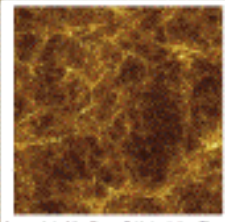
² Guth, 1981: Physical Review D

³ Yoshikawa and Fukushige, 2005, PASJ

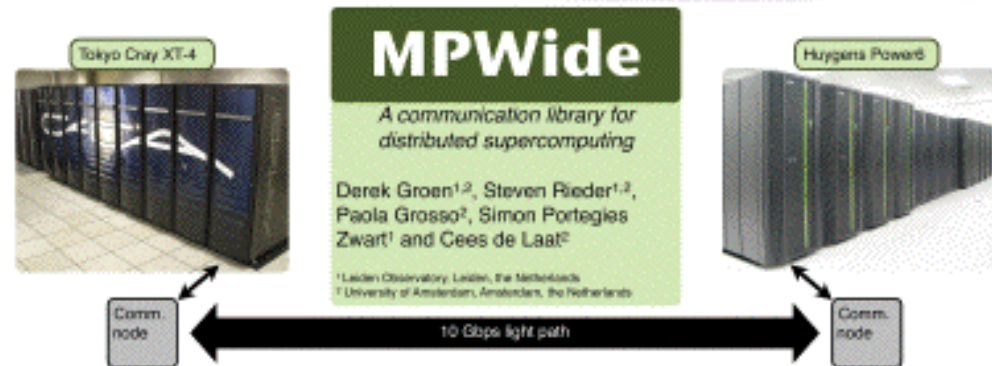
Motivation

We use MPWide to manage the wide area communications in the CosmoGrid project, where cosmological N-body simulations run on grids of supercomputers connected by high performance optical networks. To take full advantage of the network light paths in CosmoGrid, we need a message passing library that supports the ability to use customized communication settings (e.g. custom number of streams, window sizes) for individual network links among the sites. The supercomputers see use vary both in hardware architectures and software setup.

Many supercomputers have a recommended MPI implementation which has been optimized for the network architecture of that particular machine. Installing and optimizing a homogeneous MPI implementation on multiple supercomputer platforms is a task that may be politically difficult to initiate, and requires considerable effort and man hours to complete. This has led us to develop MPWide, a light-weight communication library which connects two applications, each of them running with the locally recommended MPI implementation.



A snapshot of the CosmoGrid simulation. The bright dense areas form a cosmic web structure.



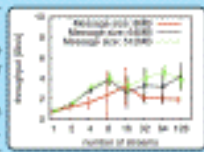
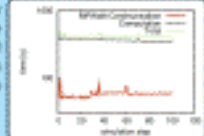
Benchmarks

We measured the performance of MPWide between two nodes on different supercomputers, one located in The Netherlands, the other in Finland. These supercomputers are connected with a 10 Gbps interface. The round trip time for this network is 37.6 ms.

Each test consists of 100 two-way message exchanges, where we record the average throughput and the standard error. We performed the tests over a shared network with frequent background traffic.

Our tests show increased performance when using more streams, especially for larger message sizes.

We also tested MPWide in a production environment, during a CosmoGrid run. In this run, we used the Huygens supercomputer in Amsterdam and the Cray supercomputer in Tokyo. In this run, the calculation time dominated the overall performance, with the communication time constituting about one eighth of the total execution time.

Related work and future

The MPI implementation most closely related to our work is the PACX-MPI[†] implementation. Like MPWide, this implementation connects different machines, while making use of the vendor MPI library on the system. The main difference between PACX-MPI and MPWide lies in the fact that MPWide supports a de-centralized startup, where PACX-MPI does not. For CosmoGrid, support for this is required, as it is not possible to start the simulation on all supercomputers from one site.

Other implementations of MPI, like Open MPI and MPICH-G2, differ further from MPWide, and do not support manual specification of the network topology, required by CosmoGrid.

In the near future, we will expand the CosmoGrid simulation to run on four supercomputer sites, and we will implement support for this in MPWide.



[†] <http://www.fhnw.de/organization/ia/cm/ia/research/pacx-mpi/>

7.6 Gb/s

Real time issue

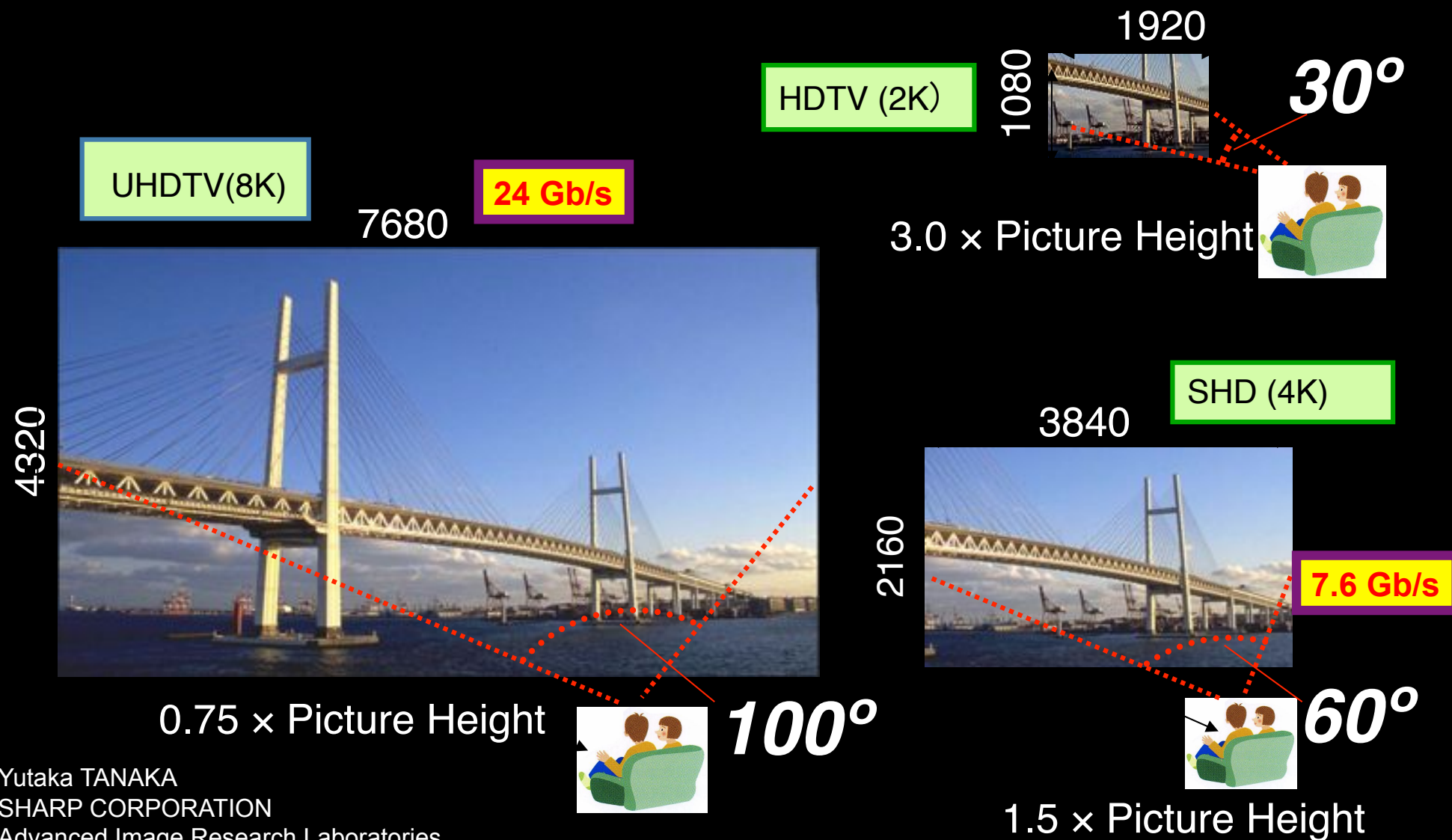


CineGrid @ Holland Festival 2007



Why is more resolution is better?

1. More Resolution Allows Closer Viewing of Larger Image
2. Closer Viewing of Larger Image Increases Viewing Angle
3. Increased Viewing Angle Produces Stronger Emotional Response



CineGrid portal



CineGrid distribution center Amsterdam

[Home](#) | [About](#) | [Browse Content](#) | [cinegrid.org](#) | [cinegrid.nl](#)

Amsterdam Node Status:

node41:
Disk space used: 8 GiB
Disk space available: 10 GiB

Search node:

Search

Browse by tag:

amsterdam animation
[antonacci](#) blender boat
bridge burny cgi delta holland
hollandfestival
leidsestraat
muziekgebouw
nieuwmarkt opera prague ship
train tram trams waag

Via Distributed via Amsterdam

CineGrid Amsterdam

Welcome to the Amsterdam CineGrid distribution node. Below are the latest additions of super-high-quality video to our node.

For more information about CineGrid and our efforts look at the about section.

Latest Additions



Wypke

Wypke

Available formats:

4k dot (4.0 KB)

Duration: 1 hour and 8 minutes

Created: 1 week, 2 days ago

Author: Wypke

Categories:



Prague Train

Steam locomotive in Prague

Available formats:

4k dot (3.9 KB)

Duration: 27 hours and 46 minutes

Created: 1 week, 2 days ago

Author: CineGrid

Categories: delta prague train



VLC: Big Buck Bunny

(C) copyright Blender Foundation | <http://www.bigbuckbunny.org>

Available formats:

1080p MPEG4 (1.1 GB)

Duration: 1 hour and 0 minutes

Created: 1 month, 1 week ago

Author: Blender Foundation

Categories: animation blender bunny
cgi

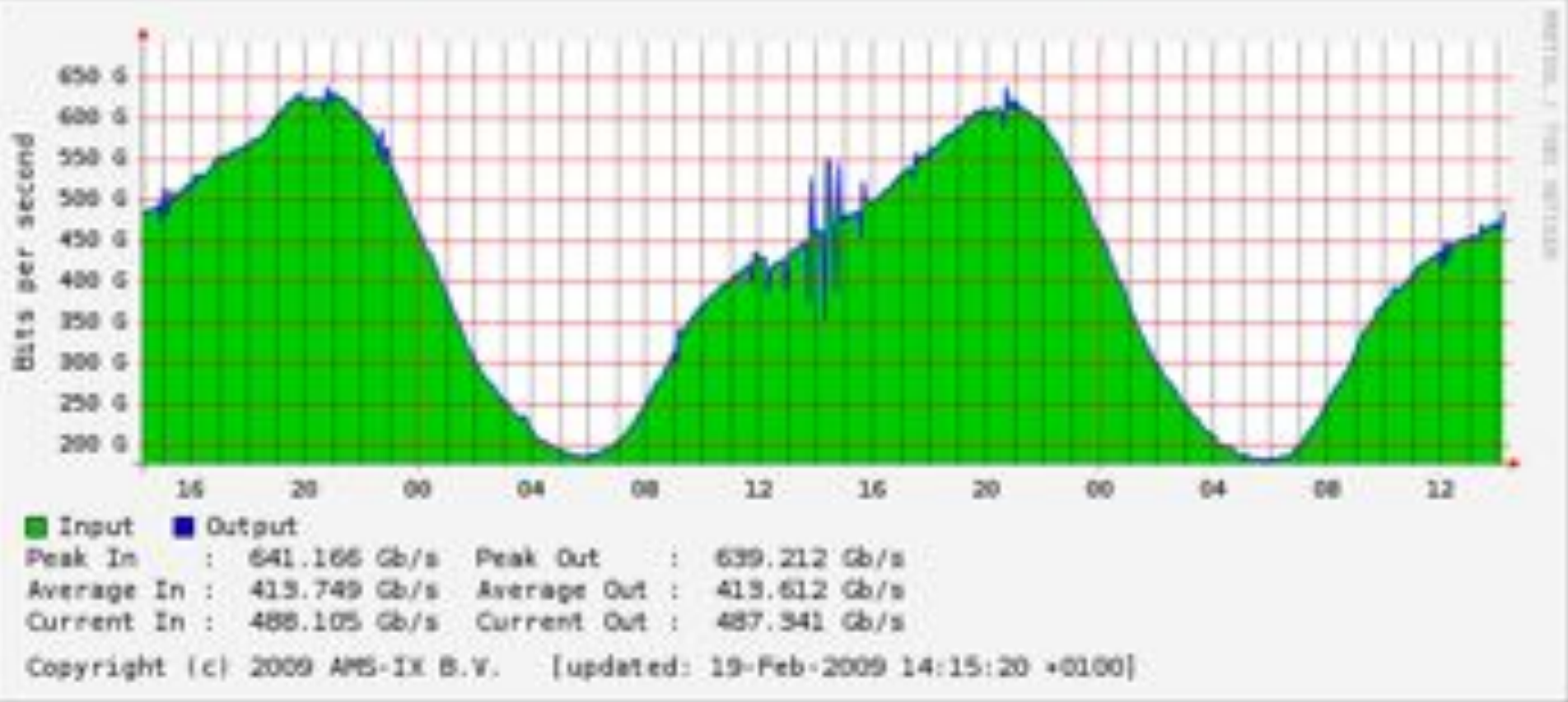
u
s
e
r
s

A. Lightweight users, browsing, mailing, home use

Need full Internet routing, one to all

B. Business/grid applications, multicast, streaming, VO's, mostly LAN

Need VPN services and full Internet routing, several to several + unlink to all



B

C

ADSL (12 Mbit/s)

BW GigE

Ref: Cees de Laat, Erik Radius, Steven Wallace, "The Rationale of the Current Optical Networking Initiatives"
iGrid2002 special issue, Future Generation Computer Systems, volume 19 issue 6 (2003)



Towards Hybrid Networking!

- Costs of photonic equipment 10% of switching 10 % of full routing
 - for same throughput!
 - Photonic vs Optical (optical used for SONET, etc, 10-50 k\$/port)
 - DWDM lasers for long reach expensive, 10-50 k\$
- Bottom line: look for a hybrid architecture which serves all classes in a cost effective way
 - map A -> L3 , B -> L2 , C -> L1 and L2
- Give each packet in the network the service it needs, but no more !

L1 \approx 2-3 k\$/port



L2 \approx 5-8 k\$/port

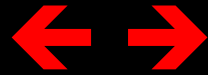


L3 \approx 75+ k\$/port



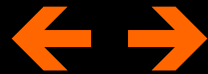
Hybrid computing

Routers



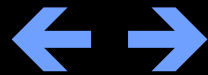
Supercomputers

Ethernet switches



Grid & Cloud

Photonic transport



GPU's

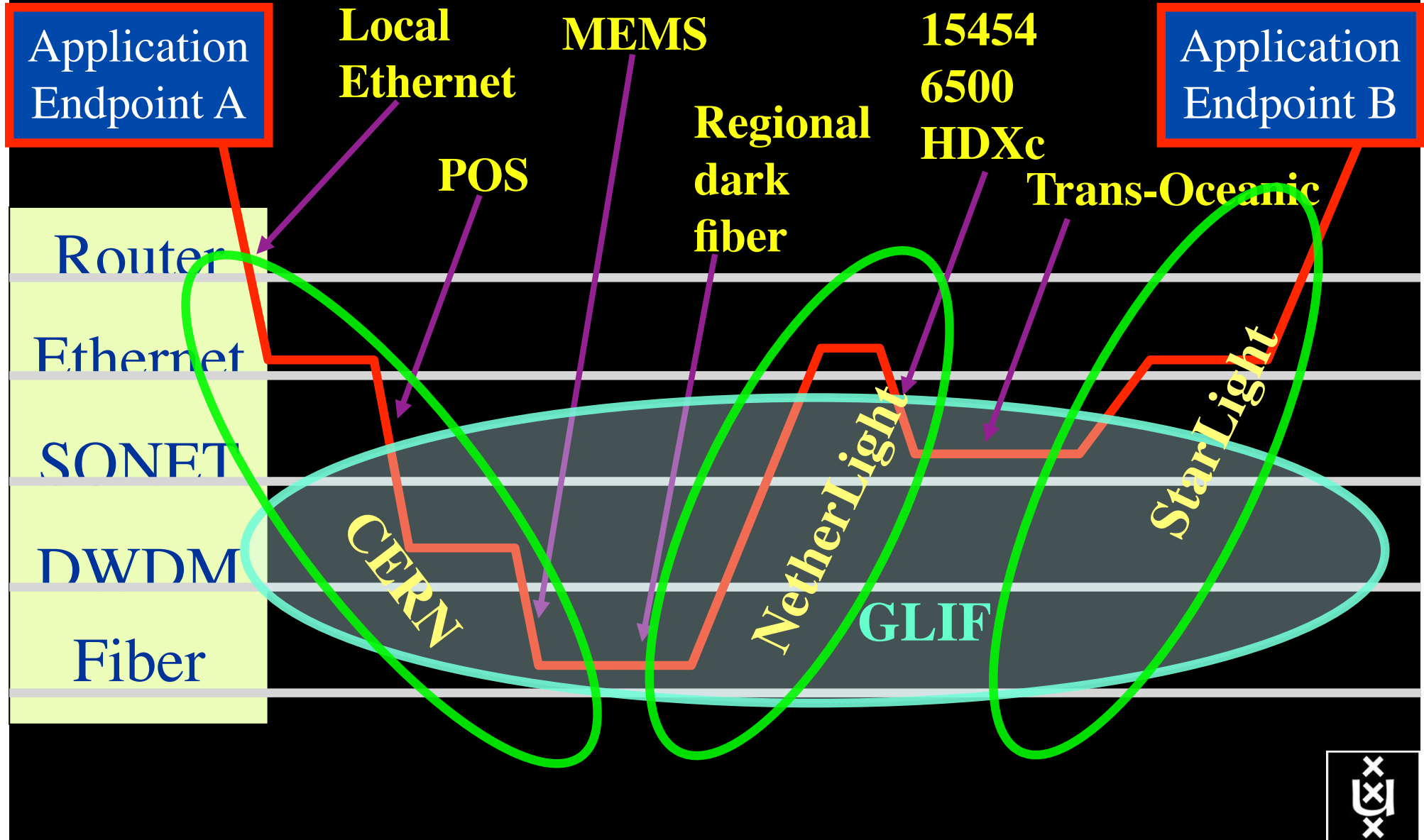
What matters:

Energy consumption/multiplication

Energy consumption/bit transported



How low can you go?

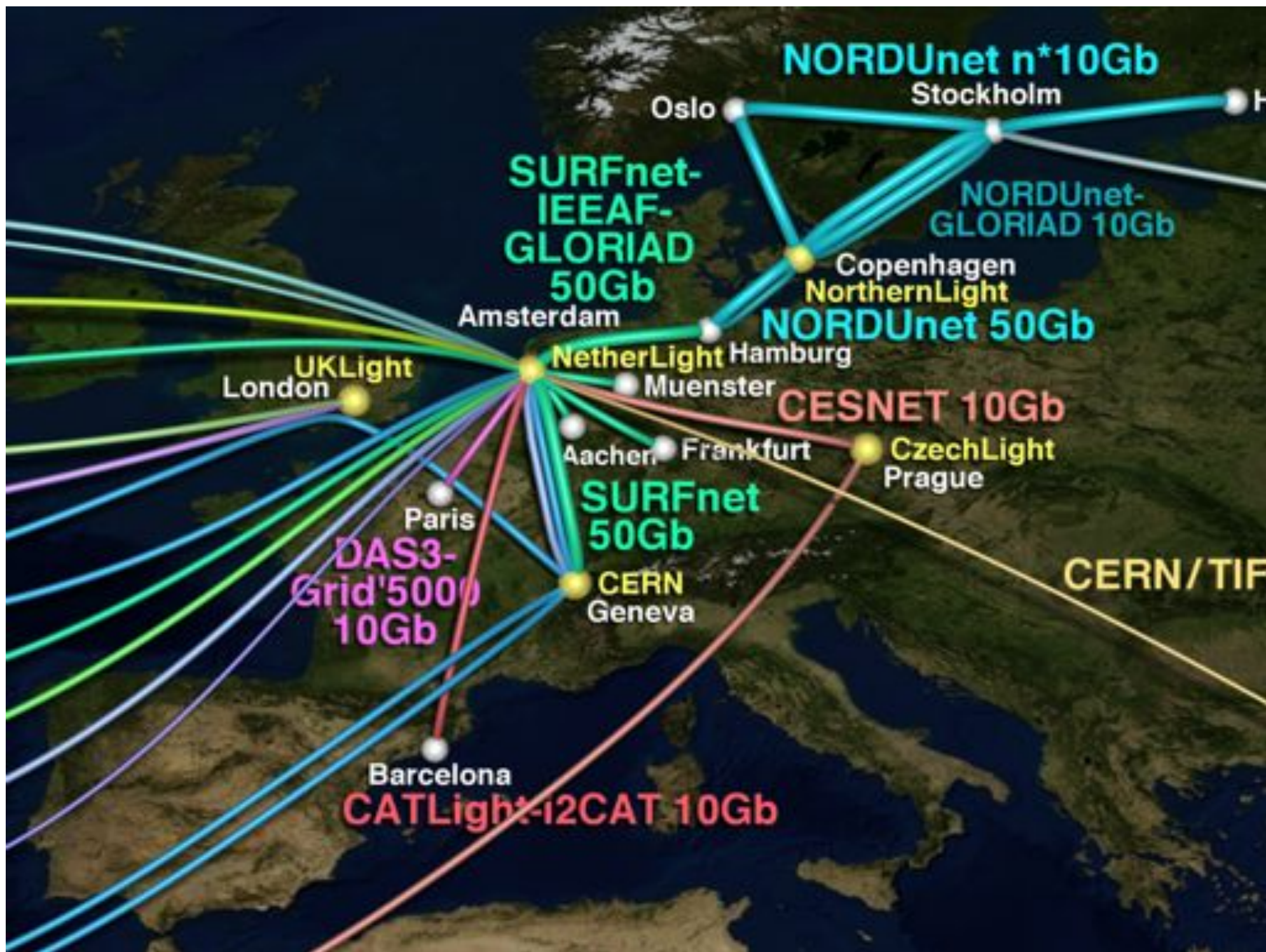




GLIF 2008

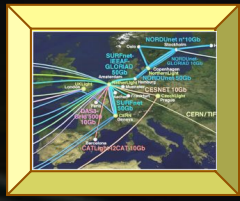
**Visualization courtesy of Bob Patterson, NCSA
Data collection by Maxine Brown.**





•VIZ

- DataExploration
- RemoteControl
- TV
- Medical
- CineGrid
- Gaming
- Conference



•DATA

- Management
- Backup
- Mining
- Web2.0
- Media
- Visualisation
- Security
- Meta



NetherLight

•GRID

- Workflow
- Clouds
- Distributed
- EventProcessing



•SUPER

- Simulations
- StreamProcessing
- Predictions





In The Netherlands SURFnet connects between 180:

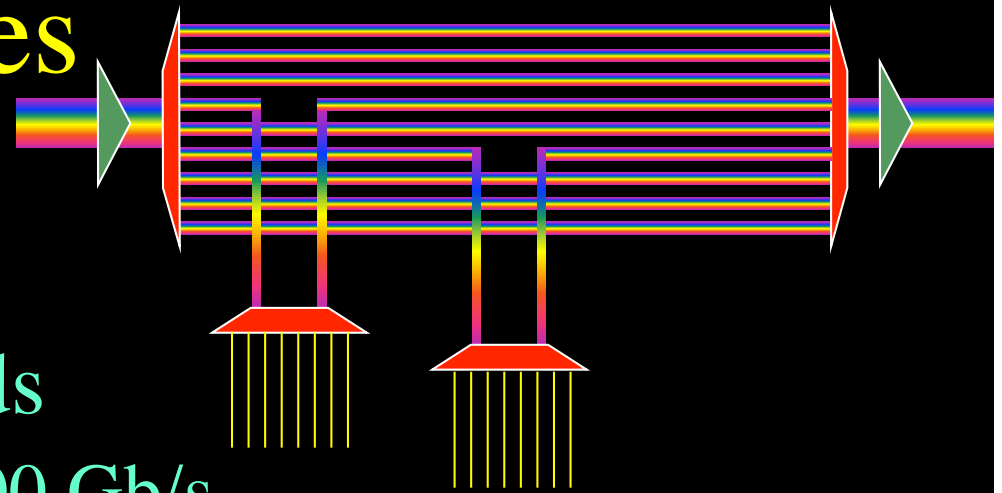
- universities;
- academic hospitals;
- most polytechnics;
- research centers.

with an indirect ~750K user base

~ 8860 km
scale
comparable
to railway
system

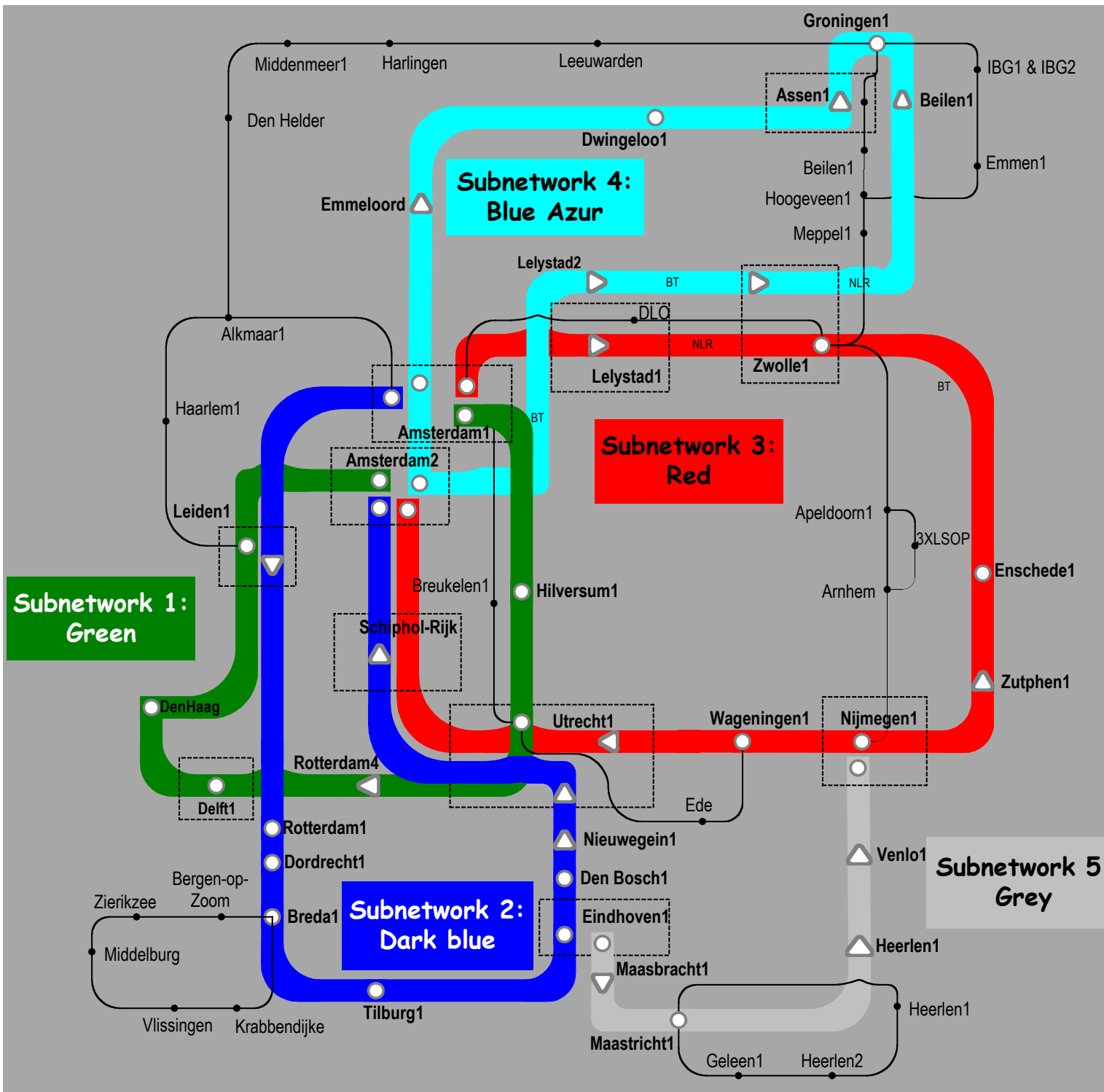


SURFnet 6 principles



- Based on dark fiber
- 4 DWDM rings of 9 bands
 - Each capable of 10, 40, 100 Gb/s
 - each 4 (100 GHz spacing) or 8 (50 GHz spacing) colors
- Universities each have 1 band to connect their Routers +LightPaths
- Connect with 1 or 10 Gb/s Ethernet LanPhy
- Routing in Amsterdam in 2 core POP's!
- International connectivity in Amsterdam
- Lambda service between ring POP's and to NetherLight





Common Photonic Layer (CPL) in SURFnet6

supports up to 72 Lambda's of 10 G each
40 G soon.

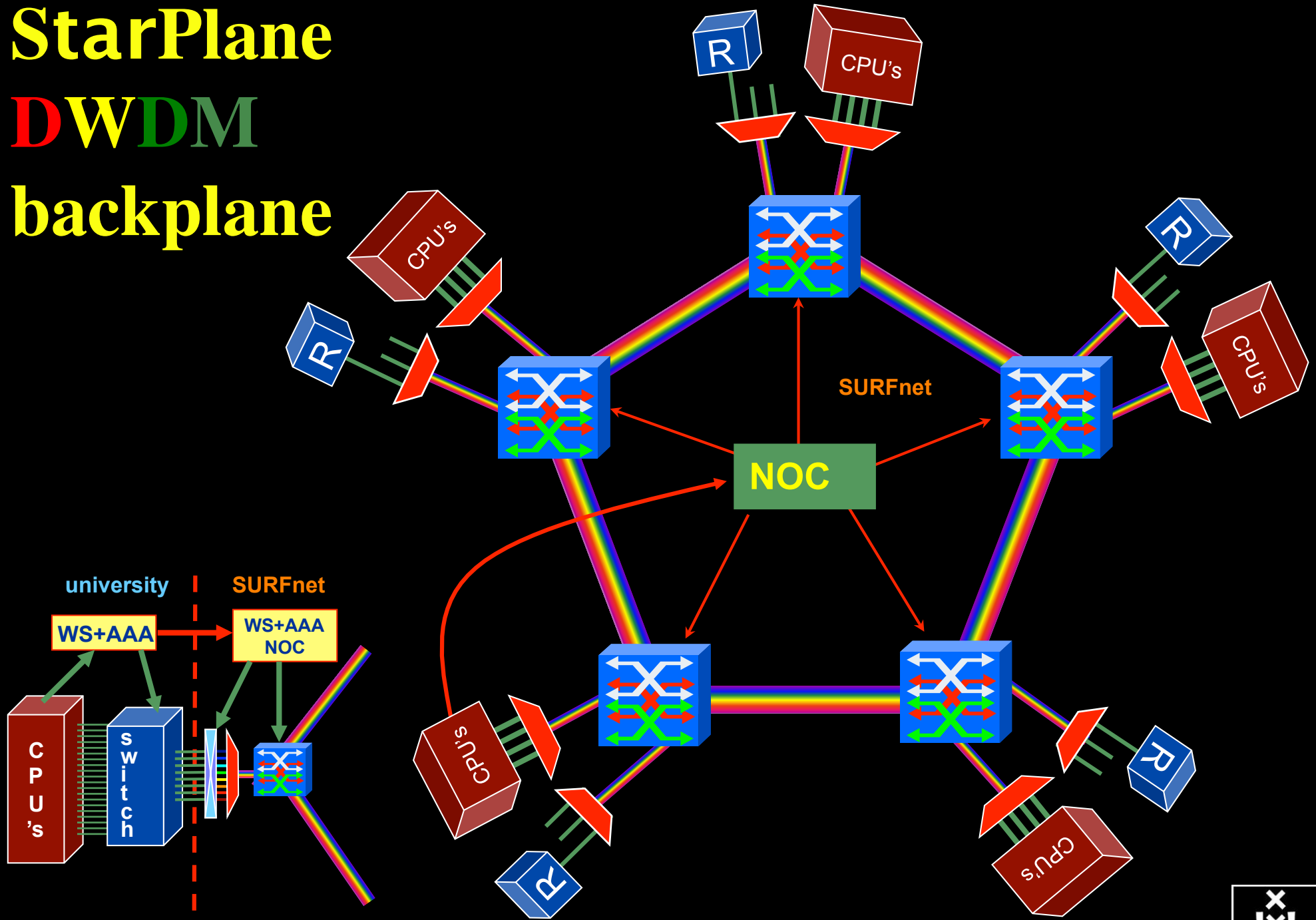


Contents

1. The need for hybrid networking
2. StarPlane; a grid controlled photonic network
3. Cross Domain Authorization using Tokens
4. RDF/Network Description Language
5. Tera-networking
6. Programmable networks

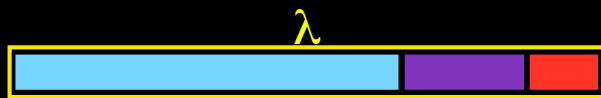


StarPlane DWDM backplane

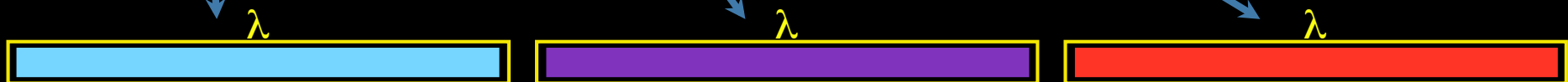


QOS in a non destructive way!

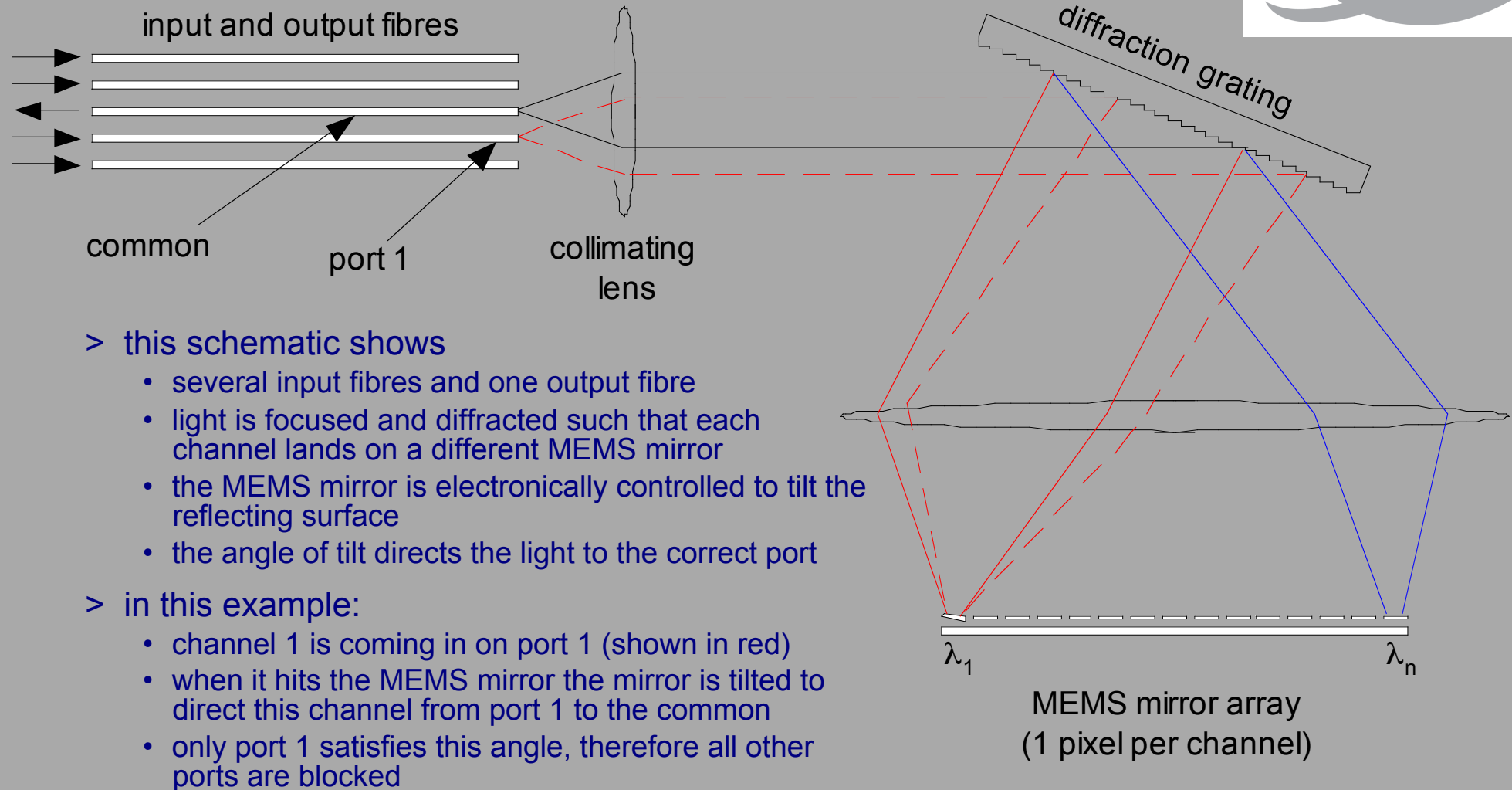
- Destructive QOS:
 - have a link or λ
 - set part of it aside for a lucky few under higher priority
 - rest gets less service



- Constructive QOS:
 - have a λ
 - add other λ 's as needed on separate colors
 - move the lucky ones over there
 - rest gets also a bit happier!



Module Operation



> this schematic shows

- several input fibres and one output fibre
- light is focused and diffracted such that each channel lands on a different MEMS mirror
- the MEMS mirror is electronically controlled to tilt the reflecting surface
- the angle of tilt directs the light to the correct port

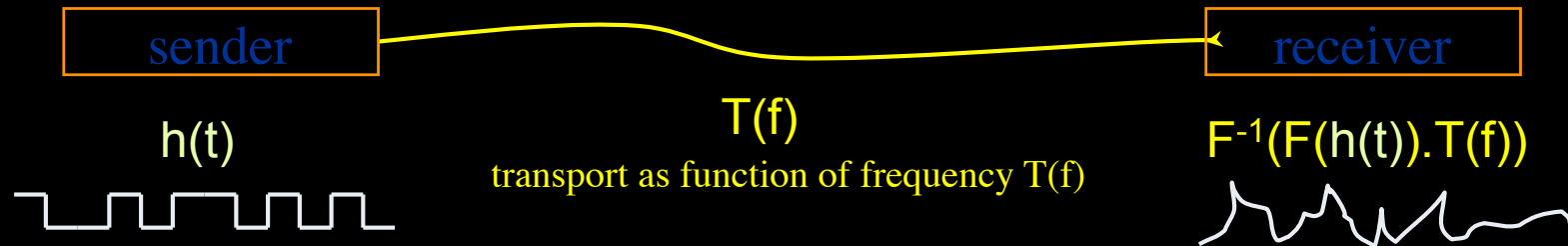
> in this example:

- channel 1 is coming in on port 1 (shown in red)
- when it hits the MEMS mirror the mirror is tilted to direct this channel from port 1 to the common
- only port 1 satisfies this angle, therefore all other ports are blocked

MEMS mirror array
(1 pixel per channel)

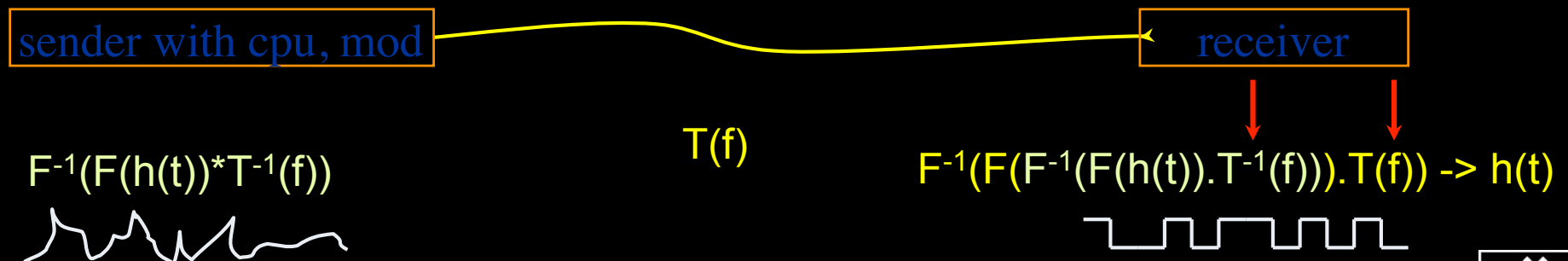
Dispersion compensating modem: eDCO from NORTEL

(Try to Google eDCO :-)



Solution in 5 easy steps for dummy's :

1. try to figure out $T(f)$ by trial and error
2. invert $T(f) \rightarrow T^{-1}(f)$
3. computationally multiply $T^{-1}(f)$ with Fourier transform of bit pattern to send
4. inverse Fourier transform the result from frequency to time space
5. modulate laser with resulting $h'(t) = F^{-1}(F(h(t)).T^{-1}(f))$

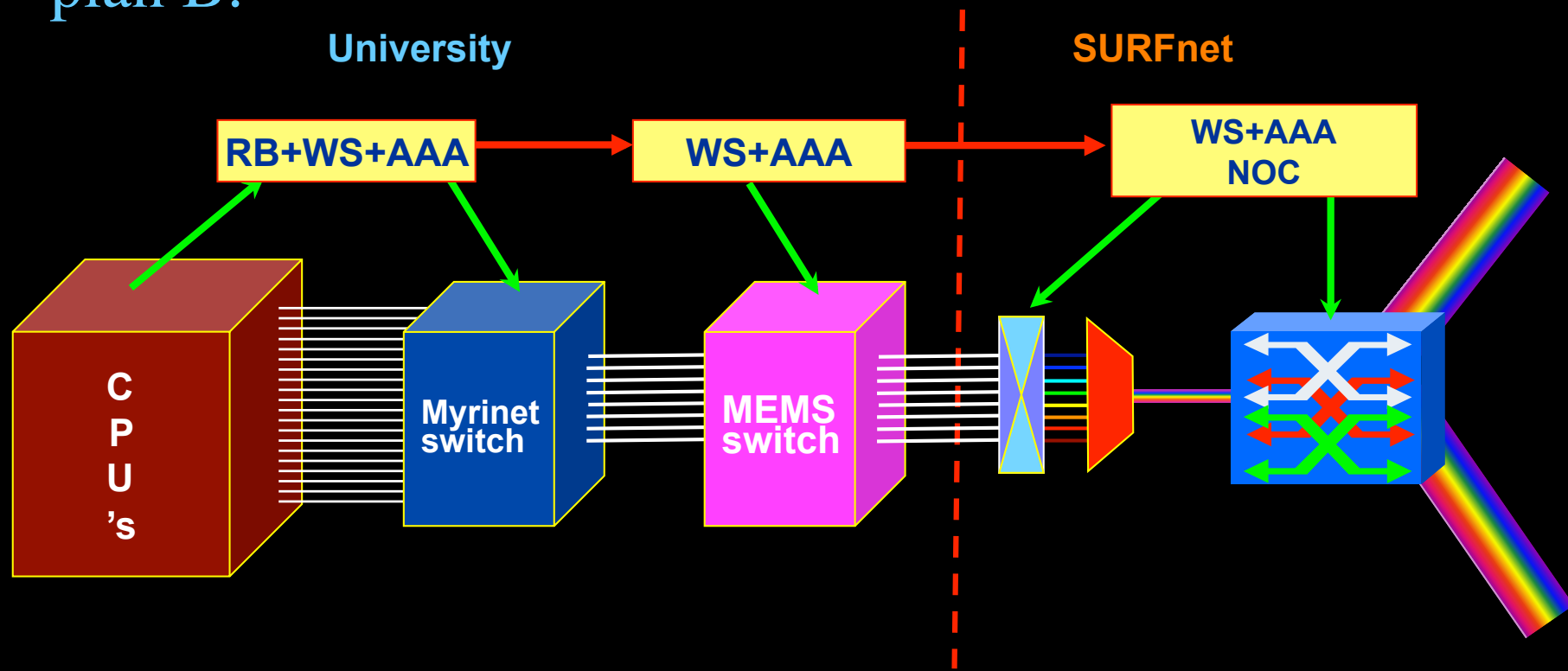


(ps. due to power \sim square E the signal to send **looks** like uncompensated received but is not)

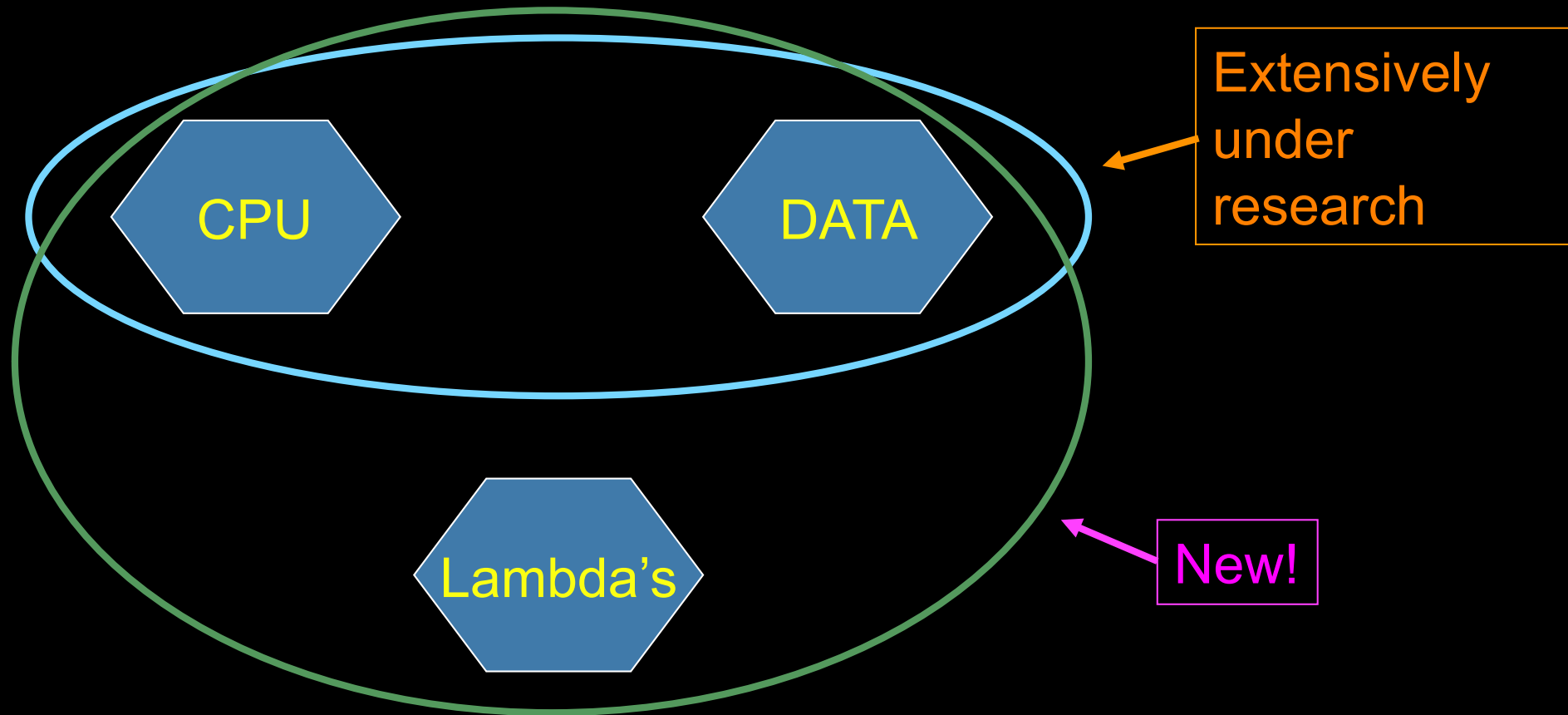


The challenge for sub-second switching

- bringing up/down a λ takes minutes
 - this was fast in the era of old time signaling (phone/fax)
 - $\lambda \ 2 \ \lambda$ influence (Amplifiers, non linear effects)
 - however minutes is historically grown, 5 nines, up for years
 - working with Nortel to get setup time significantly down
- plan B:



GRID Co-scheduling problem space



The StarPlane vision is to give flexibility directly to the applications by allowing them to choose the logical topology in real time, ultimately with sub-second lambda switching times on part of the SURFnet6 infrastructure.



Overview Throughput Load Ping UDP Plot

Scroll line: Last 7 days:

12:30:01 30 min.

Overview Net Tests between DAS-3 Hosts

- [Authenticate here](#) to store the current table settings in your cookies file.
- See the [getting started](#) introduction or the [user guide](#) for a description of the table below.
- See also the [hosts documentation](#).
- Some [observations](#) about the package and the required bandwidth.

Select ping value: [min](#), [avg](#), [max](#), [all](#), [hist](#).

Select UDP value: [rate](#), [test](#).

MAY 31th 2007

DAS-3 Net Test Results

Date: 31/05/2007

Time: 12:30:01

Load

VU-083	VU-085	LIACS-125	LIACS-127	UvA-236	UvA-239	UvA-236-M	UvA-239-M
0	0	0.097	0	0.013	0.01	0.017	0.15

Ping Min [ms]

(see 30 columns)

	VU-083	VU-085	LIACS-125	LIACS-127	UvA-236	UvA-239	UvA-236-M	UvA-239-M
VU-083	---				0.696		---	---
VU-085		---	1.390				---	---
LIACS-125		1.390	---				---	---
LIACS-127				---		1.230	---	---
UvA-236	0.696				---		---	---
UvA-239				1.230		---	---	---
UvA-236-M							---	0.025
UvA-239-M							0.025	---

Throughput [Mbit/s]

(see 30 columns)

	VU-083	VU-085	LIACS-125	LIACS-127	UvA-236	UvA-239	UvA-236-M	UvA-239-M
VU-083	---				4684.22		---	---
VU-085		---	4621.05				---	---



Name: Overview Throughput Scroll size: Last 7 days

Repeat: Load Ping UDP Plot <<< << >> >>> 12:30:01 30 min

	YU-083	YU-085	LIACS-125	LIACS-127	UvA-236	UvA-239	UvA-236-M	UvA-239-M
YU-083	---				4684.22		---	---
YU-085		---	4621.05				---	---
LIACS-125		4776.33	---				---	---
LIACS-127				---		4235.37	---	---
UvA-236	4227.76				---		---	---
UvA-239				4992.85		---	---	---
UvA-236-M	---	---	---	---	---	---	---	4111.01
UvA-239-M	---	---	---	---	---	---	5404.32	---

UDP Data Rate [Mbit/s]

(see test column)

	YU-083	YU-085	LIACS-125	LIACS-127	UvA-236	UvA-239	UvA-236-M	UvA-239-M
YU-083	---				6550.02		---	---
YU-085		---	6549.81				---	---
LIACS-125		6547.25	---				---	---
LIACS-127				---		6546.23	---	---
UvA-236	6550.12				---		---	---
UvA-239				6549.81		---	---	---
UvA-236-M	---	---	---	---	---	---	---	6550.43
UvA-239-M	---	---	---	---	---	---	6564.47	---

The load, roundtrip, throughput and UDP data series are each scaled with their private color distributions as is displayed below:

load	0	0.25	0.5	0.75	1	1.25	1.5	1.75	2
ping min [ms]	0.025	0.394	0.364	0.533	0.703	0.872	1.041	1.211	1.38
throughput [Mbit/s]	4111.01	4272.674	4434.338	4596.001	4757.665	4919.329	5080.993	5242.656	5404.32
UDP rate [Mbit/s]	6546.23	6548.51	6550.79	6553.07	6555.35	6557.63	6559.91	6562.19	6564.47

• Download the raw, zipped [data file](#). Download this [version](#) of the package to view it locally.

Net **Repl**

Ping All [ms] from / to node125.das3.liacs.nl (LIACS-125)

Skipped tests: UvA-236-M, UvA-239-M

Date	Time	>> YU-083	<< YU-083	>> YU-085	<< YU-085	>> LIACS-127	<< LIACS-127	>> UvA-236	<< UvA-236	>> UvA-239	<< UvA-239
31/05/2007	12:30:01			1.380 / 1.382 / 1.410	1.380 / 1.383 / 1.420						
31/05/2007	12:00:01			1.380 / 1.383 / 1.410	1.380 / 1.384 / 1.450						
31/05/2007	11:30:01			1.380 / 1.383 / 1.410	1.380 / 1.382 / 1.390						
31/05/2007	11:00:02			1.380 / 1.382 / 1.410	1.380 / 1.382 / 1.400						
31/05/2007	10:30:01			1.380 / 1.383 / 1.390	1.380 / 1.382 / 1.390						
31/05/2007	10:00:01			1.380 / 1.382 / 1.410	1.380 / 1.383 / 1.410						
31/05/2007	09:30:01			1.380 / 1.384 / 1.410	1.380 / 1.382 / 1.400						
31/05/2007	09:00:01			1.380 / 1.382 / 1.410	1.380 / 1.383 / 1.400						
31/05/2007	08:30:02			1.380 / 1.383 / 1.410	1.380 / 1.382 / 1.400						
31/05/2007	08:00:01			1.380 / 1.383 / 1.410	1.380 / 1.383 / 1.410						
31/05/2007	07:30:02			1.380 / 1.382 / 1.390	1.380 / 1.383 / 1.390						
31/05/2007	07:00:01			1.380 / 1.382 / 1.410	1.380 / 1.383 / 1.400						
31/05/2007	06:30:01			1.380 / 1.383 / 1.410	1.380 / 1.382 / 1.390						
31/05/2007	06:00:01			1.380 / 1.382 / 1.410	1.380 / 1.382 / 1.420						
31/05/2007	05:30:01			1.380 / 1.382 / 1.400	1.380 / 1.382 / 1.410						
31/05/2007	05:00:01			1.380 / 1.382 / 1.410	1.380 / 1.382 / 1.390						
31/05/2007	04:30:01			1.380 / 1.381 / 1.390	1.380 / 1.383 / 1.390						
31/05/2007	04:00:01			1.380 / 1.382 / 1.410	1.380 / 1.384 / 1.410						
31/05/2007	03:30:02			1.380 / 1.384 / 1.410	1.380 / 1.382 / 1.400						
31/05/2007	03:00:02			1.380 / 1.382 / 1.410	1.380 / 1.382 / 1.400						
31/05/2007	02:30:01			1.380 / 1.382 / 1.400	1.380 / 1.382 / 1.400						
31/05/2007	02:00:01			1.380 / 1.383 / 1.410	1.380 / 1.384 / 1.410						
31/05/2007	01:30:01			1.380 / 1.382 / 1.410	1.380 / 1.382 / 1.390						
31/05/2007	01:00:01			1.380 / 1.382 / 1.410	1.380 / 1.383 / 1.400						

Very constant and predictable!

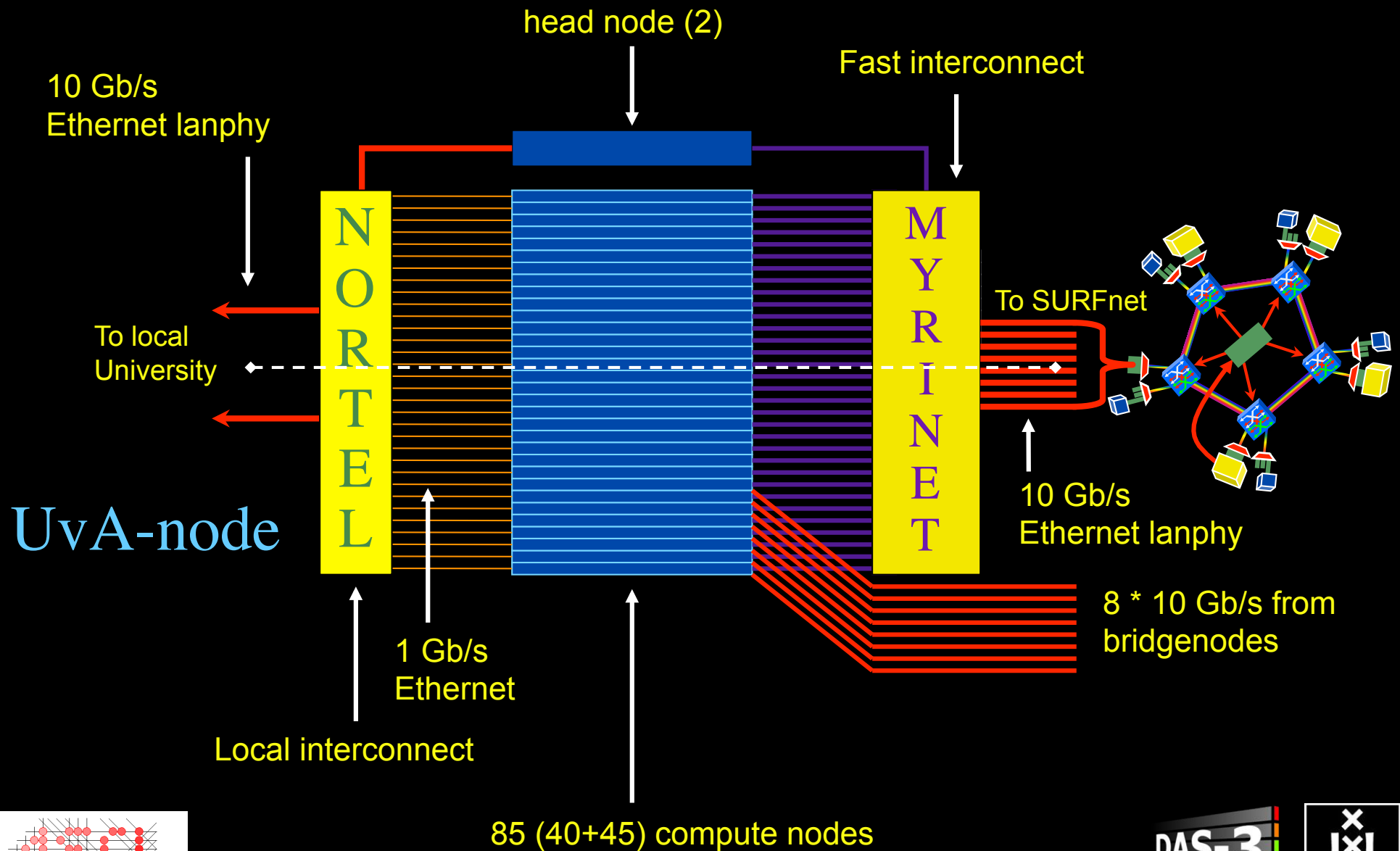


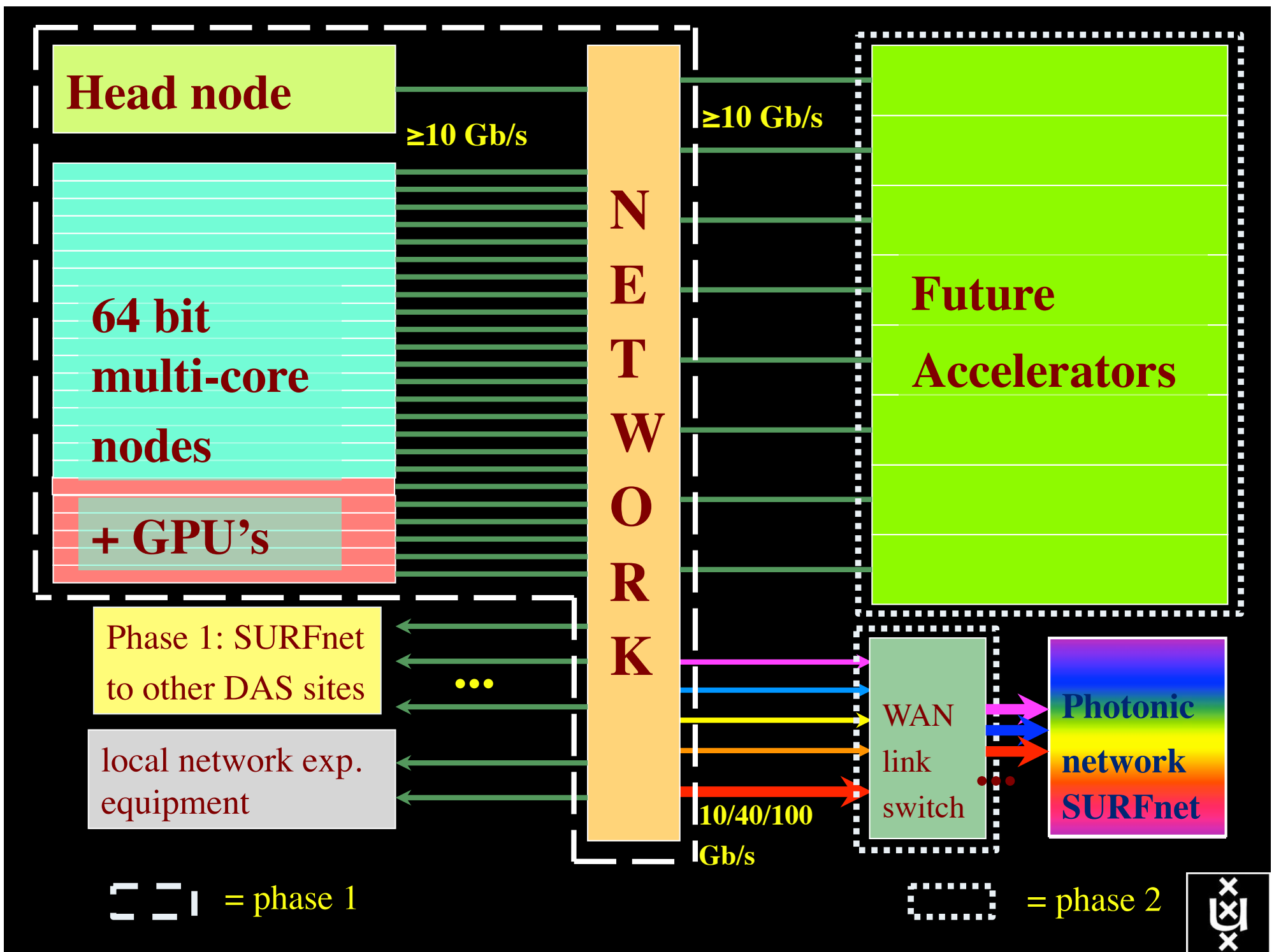
What makes StarPlane fly?

- Wavelength Selective Switches
 - for the “low cost” photonics
- Sandbox by confining StarPlane to one band
 - for experimenting on a production network
- Optimization of the controls to turn on/off a Lambda
 - direct access to part of the controls at the NOC
- electronic Dynamically Compensating Optics (eDCO)
 - to compensate for changing lengths of the path
- traffic engineering
 - to create the OPN topologies needed by the applications
- Open Source GMPLS
 - to facilitate policy enabled cross domain signaling



DAS-3 Cluster Architecture

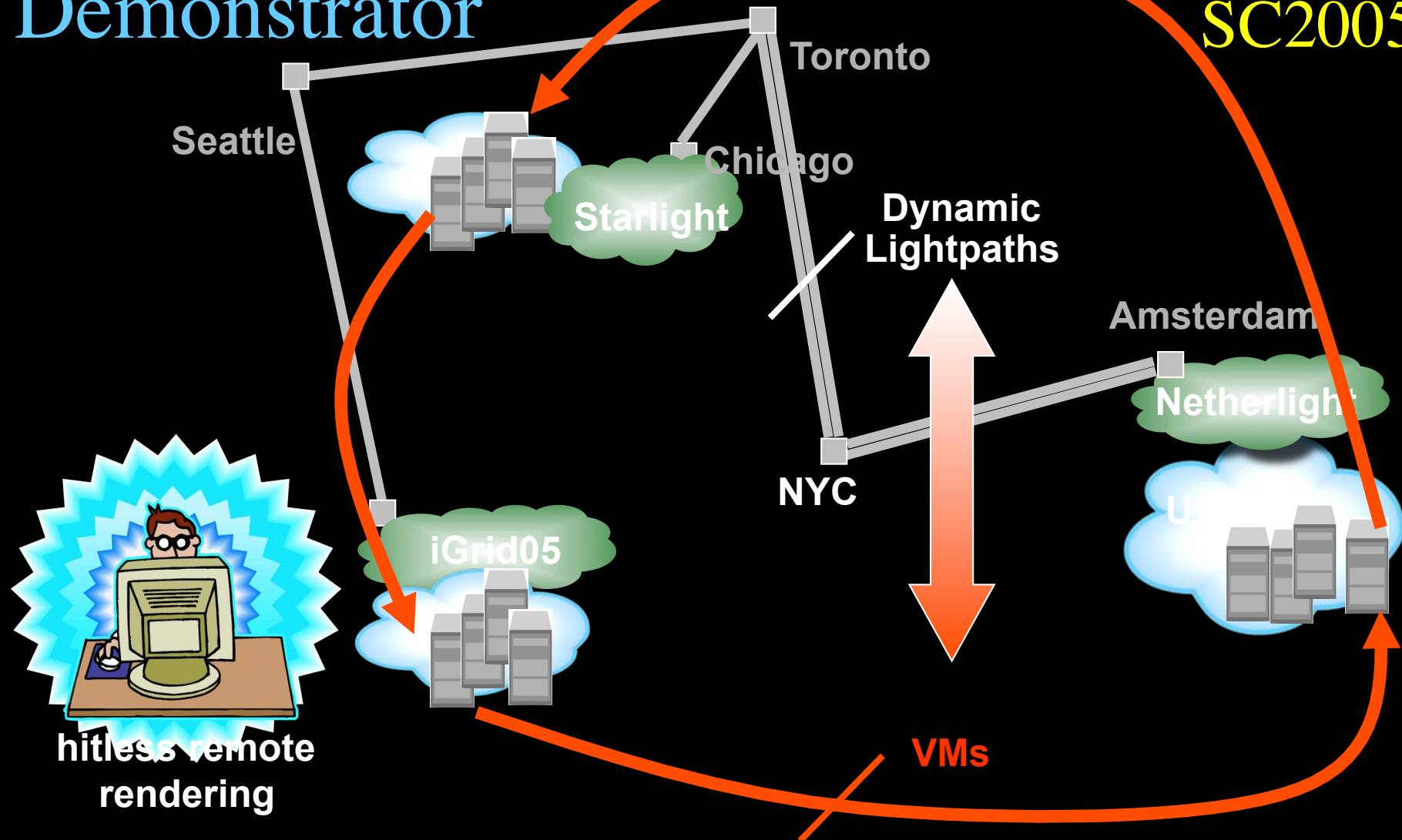




The VM Turntable Demonstrator

iGrid2005

SC2005



The VMs that are live-migrated run an iterative search-refine-search workflow against data stored in different databases at the various locations. A user in San Diego gets hitless rendering of search progress as VMs spin around

Power is a big issue

- UvA cluster uses (max) 30 kWh
- 1 kWh ~ 0.1 €
- per year -> 26 k€/y
- add cooling 50% -> 39 k€/y
- Emergency power system -> 50 k€/y
- per rack 10 kWh is now normal
- **YOU BURN ABOUT HALF THE CLUSTER OVER ITS LIFETIME!**
- Terminating a 10 Gb/s wave costs about 200 W
- Entire loaded fiber -> 16 kW
- Wavelength Selective Switch : few W!



Contents

1. The need for hybrid networking
2. StarPlane; a grid controlled photonic network
3. Cross Domain Authorization using Tokens
4. RDF/Network Description Language
5. Tera-networking
6. Programmable networks



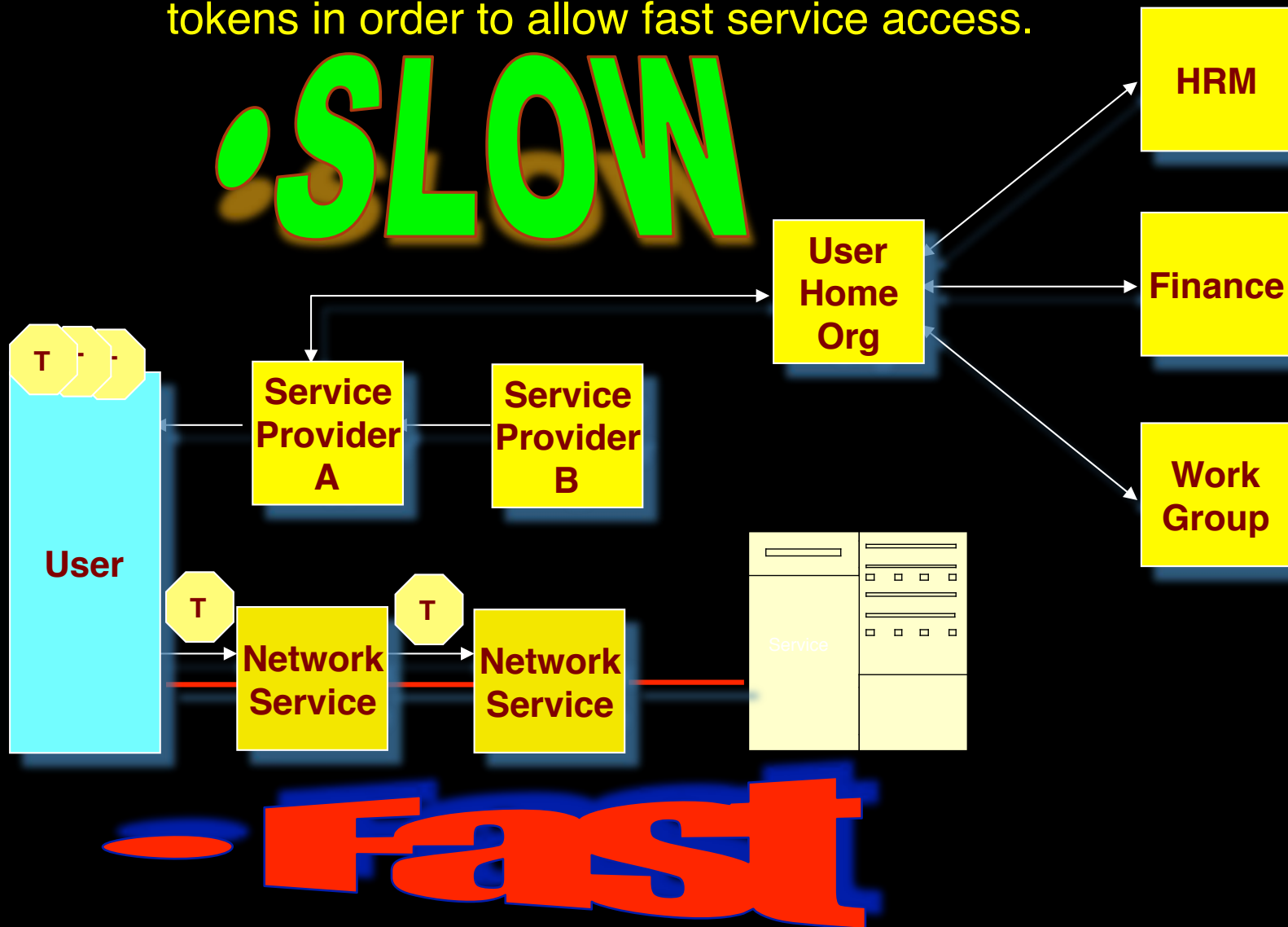
Simple service access



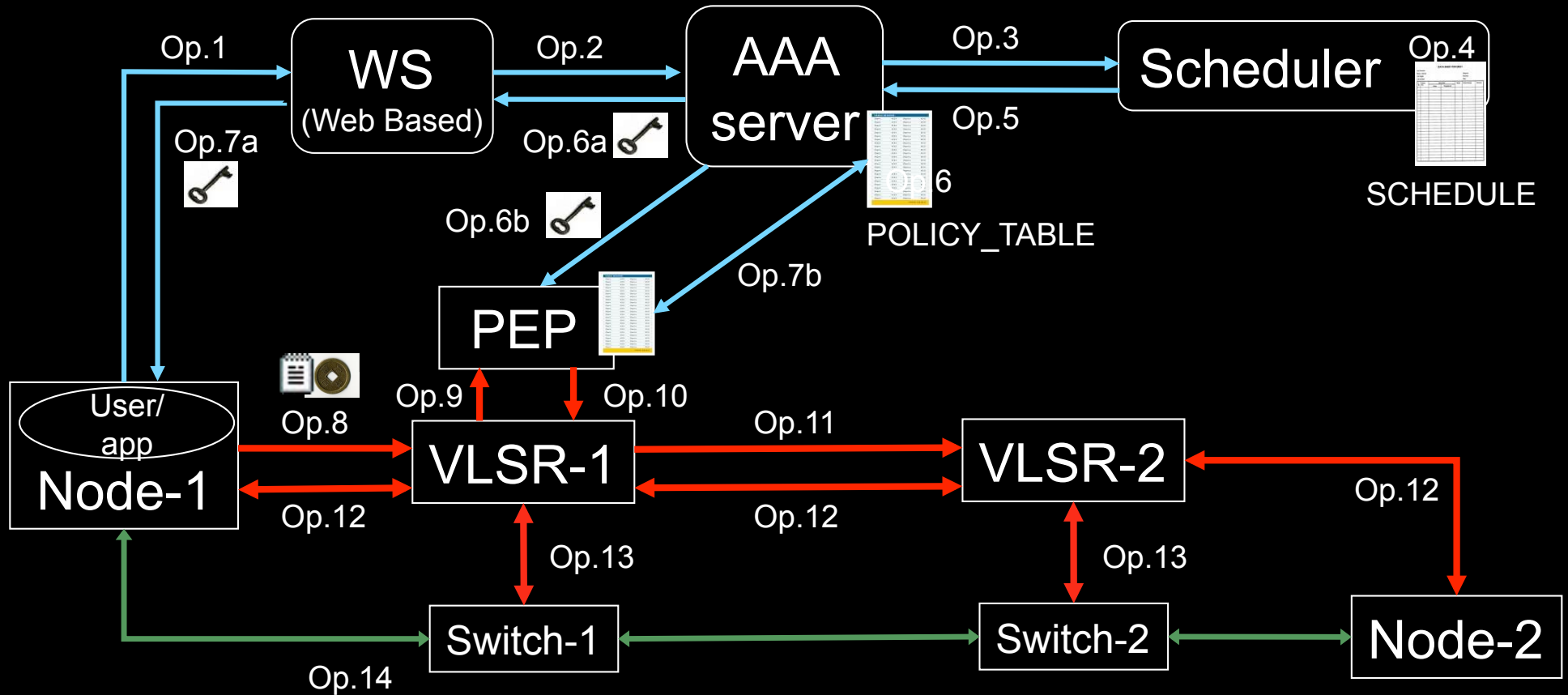
Pitlochry, Scotland - Summer 2005



Use AAA concept to split (time consuming) service authorization process from service access using secure tokens in order to allow fast service access.



DRAGON GMPLS & TBN Demo, SC06 Tampa



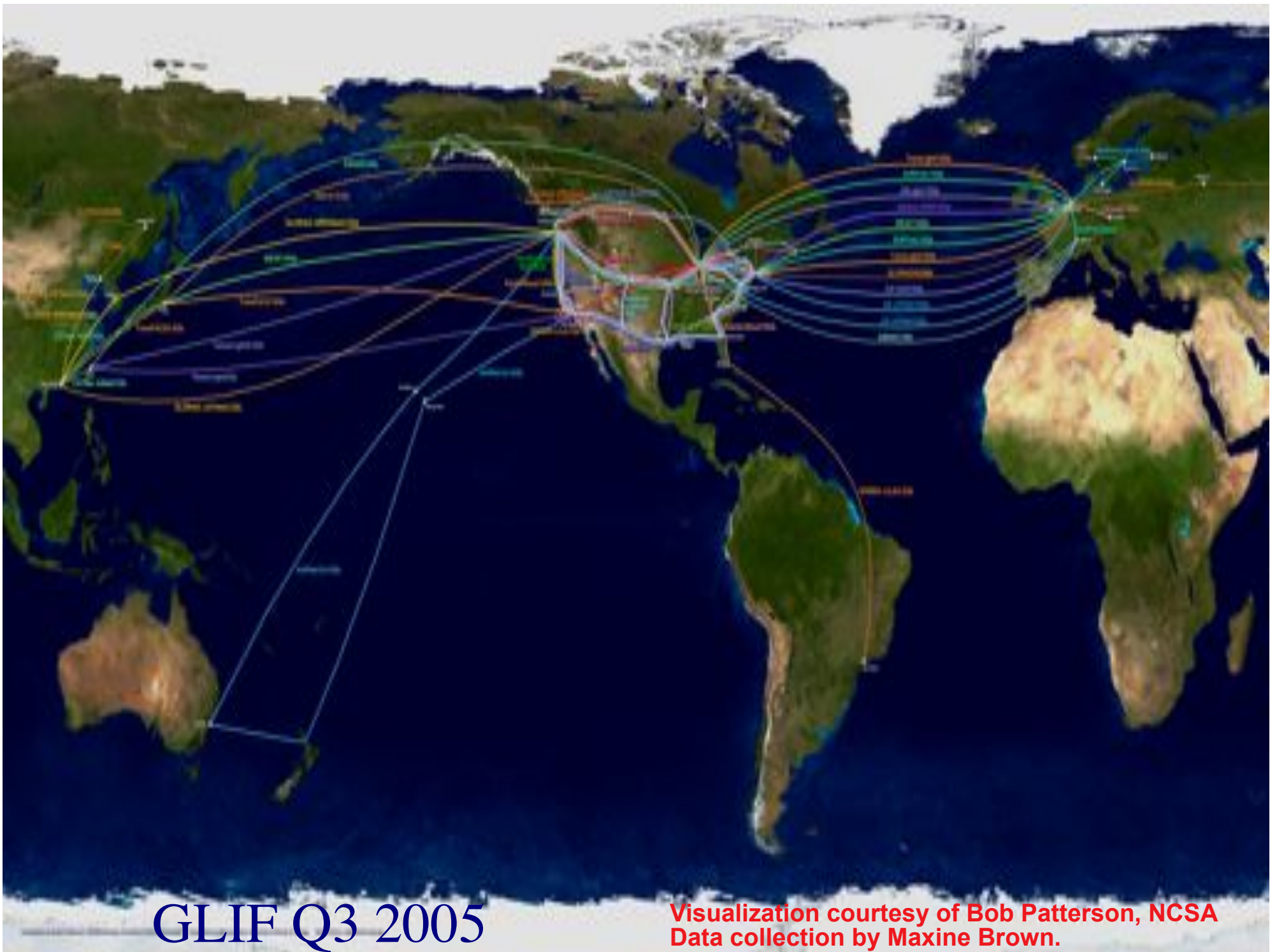
1. User (on Node1) requests a path via web to the WS.
2. WS sends the XML requests to the AAA server.
3. AAA server calculates a hashed index number and submits a request to the Scheduler.
4. Scheduler checks the SCHEDULE and add new entry.
5. Scheduler confirms the reservation to the AAA.
6. AAA server updates the POLICY_TABLE.
- 6a. AAA server issues an encrypted key to the WS.
- 6b. AAA server passes the same key to the PEP.
- 7a. WS passes the key to the user.
- 7b. AAA server interacts with PEP to update the local POLICY_TABLE on the PEP.

8. User constructs the RSVP message with extra Token data by using the key and sends to VLSR-1.
9. VLSR-1 queries PEP whether the Token in the RSVP message is valid.
10. PEP checks in the local POLICY_TABLE and return YES.
11. When VLSR-1 receives YES from PEP, it forwards the RSVP message.
12. All nodes process RSVP message(forwarding/response)
13. The Ethernet switches are configured
14. LSP is set up and traffic can flow

Contents

1. The need for hybrid networking
2. StarPlane; a grid controlled photonic network
3. Cross Domain Authorization using Tokens
4. RDF/Network Description Language
5. Tera-networking
6. Programmable networks



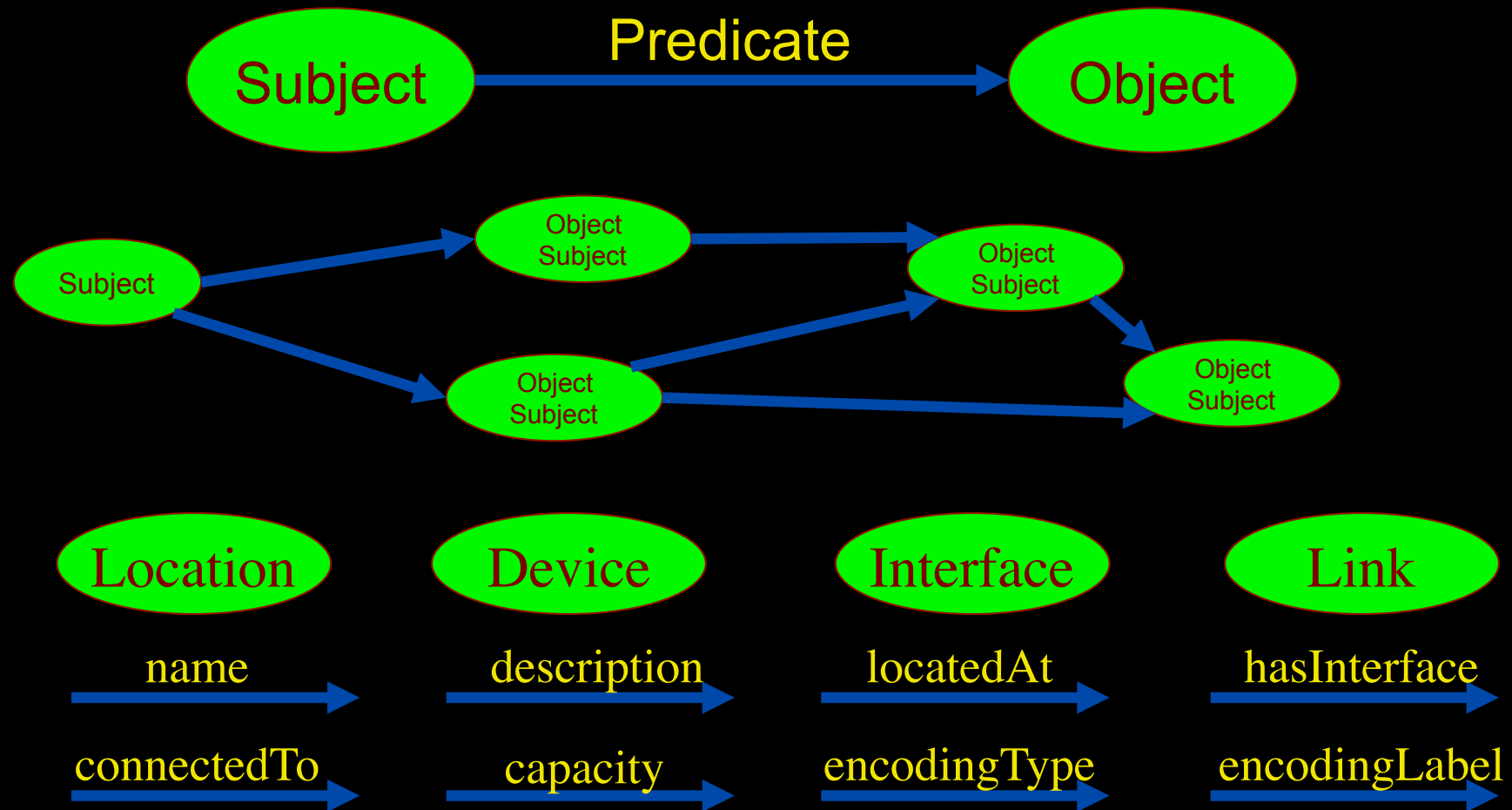


GLIF Q3 2005

Visualization courtesy of Bob Patterson, NCSA
Data collection by Maxine Brown.

Network Description Language

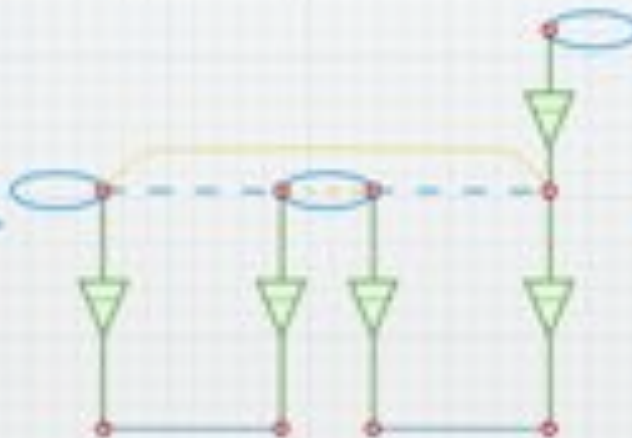
- From semantic Web / Resource Description Framework.
- The RDF uses XML as an interchange syntax.
- Data is described by triplets:



Network Description Language

Choice of RDF instead of flat XML descriptions
Grounded modeling based on G805 description:

Article: F. Dijkstra, B. Andree, K. Koymans, J. van der Ham, P. Grosso, C. de Laat, "A Multi-Layer Network Model Based on ITU-T G.805"

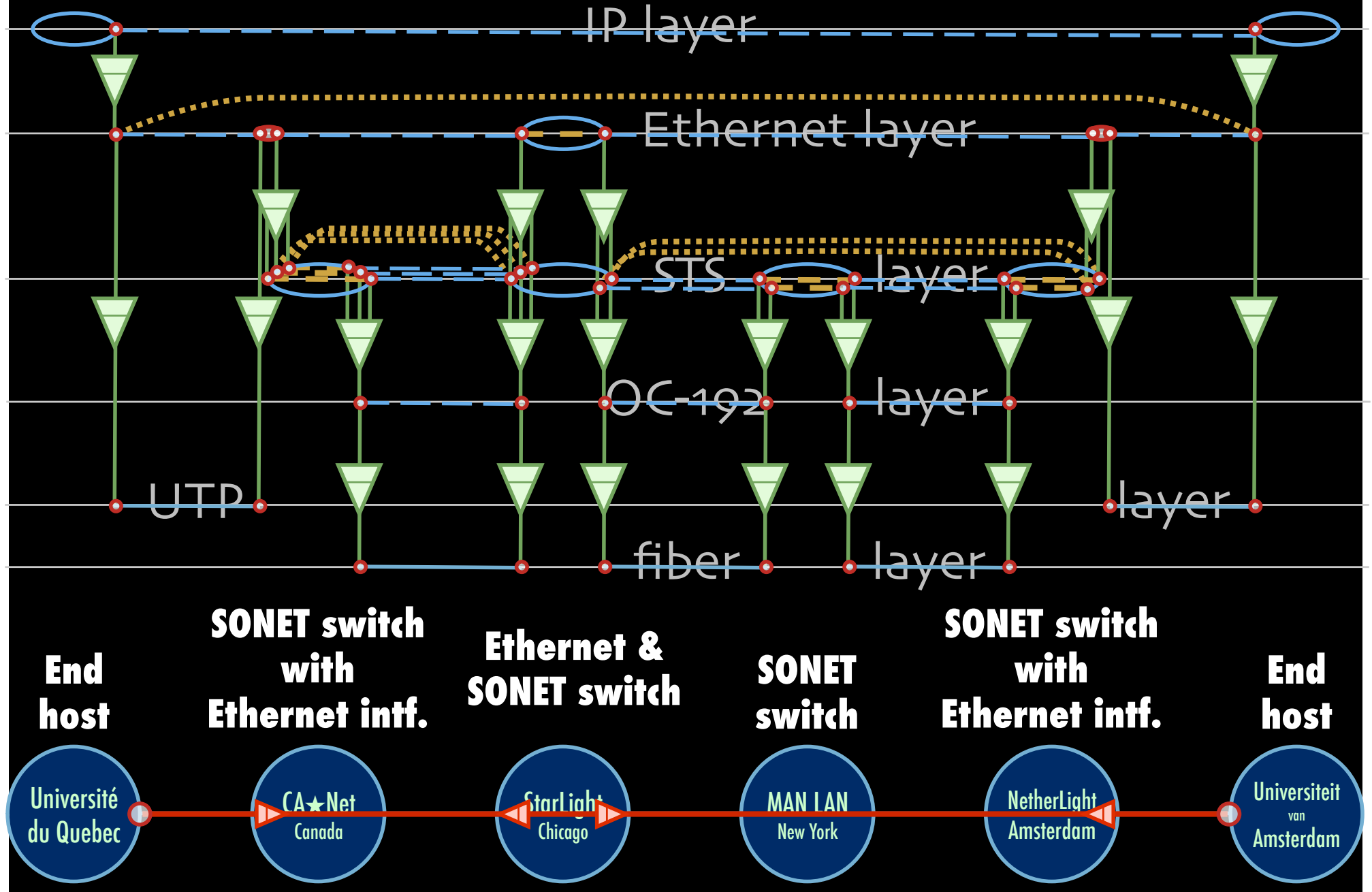


```
<rd:Device rdf:about="#force10">
  <rd:hasInterface rdf:resource="#force10 net/0">
</rd:Device>
<rd:Interface rdf:about="#force10 net/0">
  <rd:label>net/0</rd:label>
  <rd:capacity>1.2588</rd:capacity>
  <rd:conf:multiplex>
    <rd:cap:adaptation rdf:resource="#tagged-Ethernet-in-Ethernet"/>
    <rd:conf:serverPropertyValue
      rdf:resource="#MTU-1500byte"/>
  </rd:conf:multiplex>
  <rd:conf:hasChannel>
    <rd:conf:Channel rdf:about="#force10 10/0 vlan1">
      <rd:conf:hasVlan>4</rd:conf:hasVlan>
      <rd:conf:switchPort rdf:resource="#force10 g3/1 vlan7"/>
    </rd:conf:Channel>
  </rd:conf:hasChannel>
</rd:Interface>
```

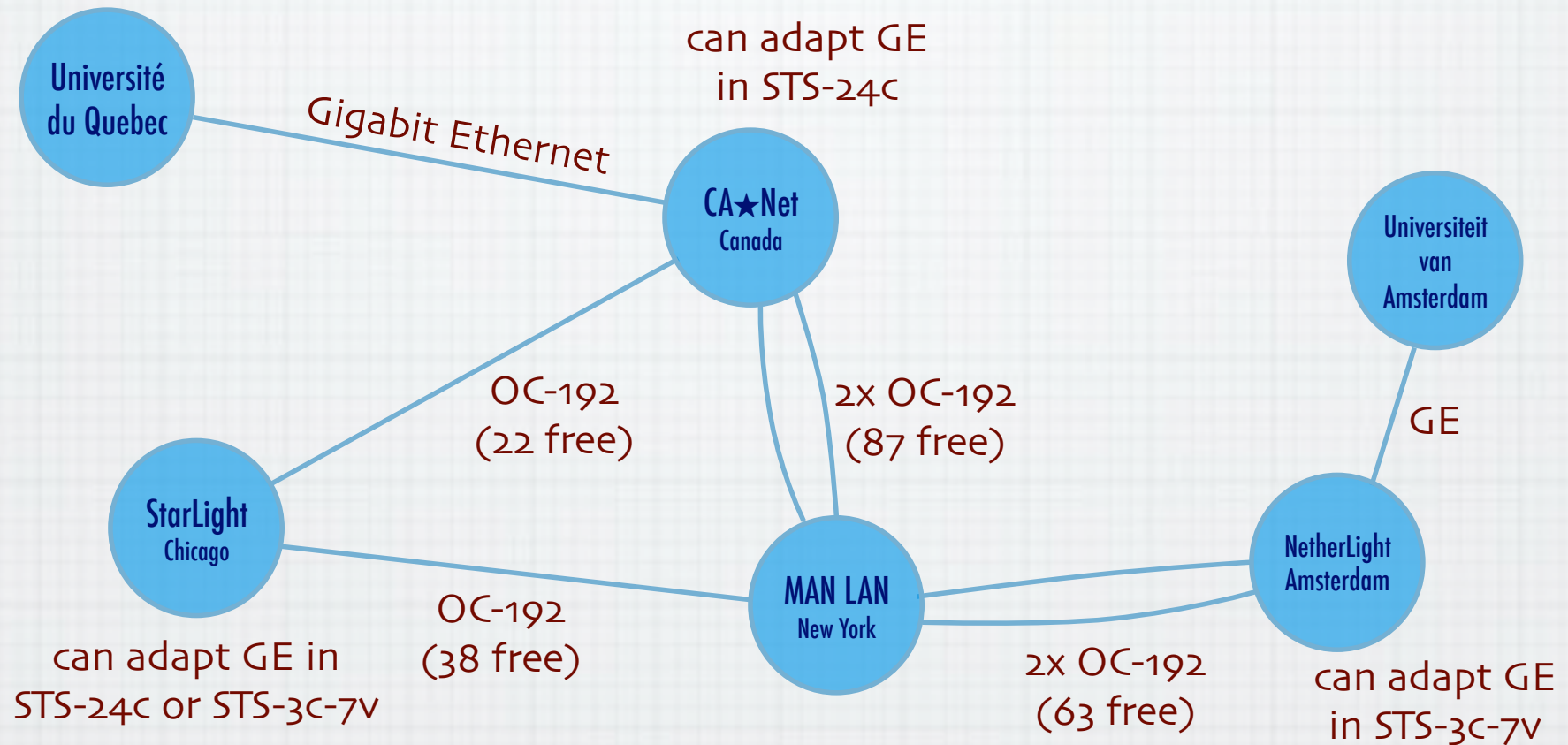

NetherLight in RDF

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ndl="http://www.science.uva.nl/research/air/ndl#">
  <!-- Description of Netherlight -->
  <ndl:Location rdf:about="#Netherlight">
    <ndl:name>Netherlight Optical Exchange</ndl:name>
  </ndl:Location>
  <!-- TDM3.amsterdam1.netherlight.net -->
  <ndl:Device rdf:about="#tdm3.amsterdam1.netherlight.net">
    <ndl:name>tdm3.amsterdam1.netherlight.net</ndl:name>
    <ndl:locatedAt rdf:resource="#amsterdam1.netherlight.net"/>
    <ndl:hasInterface rdf:resource="#tdm3.amsterdam1.netherlight.net:501/1"/>
    <ndl:hasInterface rdf:resource="#tdm3.amsterdam1.netherlight.net:501/3"/>
    <ndl:hasInterface rdf:resource="#tdm3.amsterdam1.netherlight.net:501/4"/>
    <ndl:hasInterface rdf:resource="#tdm3.amsterdam1.netherlight.net:503/1"/>
    <ndl:hasInterface rdf:resource="#tdm3.amsterdam1.netherlight.net:503/2"/>
    <!-- all the interfaces of TDM3.amsterdam1.netherlight.net -->
    <ndl:Interface rdf:about="#tdm3.amsterdam1.netherlight.net:501/1">
      <ndl:name>tdm3.amsterdam1.netherlight.net:POS501/1</ndl:name>
      <ndl:connectedTo rdf:resource="#tdm4.amsterdam1.netherlight.net:5/1"/>
    </ndl:Interface>
    <ndl:Interface rdf:about="#tdm3.amsterdam1.netherlight.net:501/2">
      <ndl:name>tdm3.amsterdam1.netherlight.net:POS501/2</ndl:name>
      <ndl:connectedTo rdf:resource="#tdm1.amsterdam1.netherlight.net:12/1"/>
    </ndl:Interface>
```

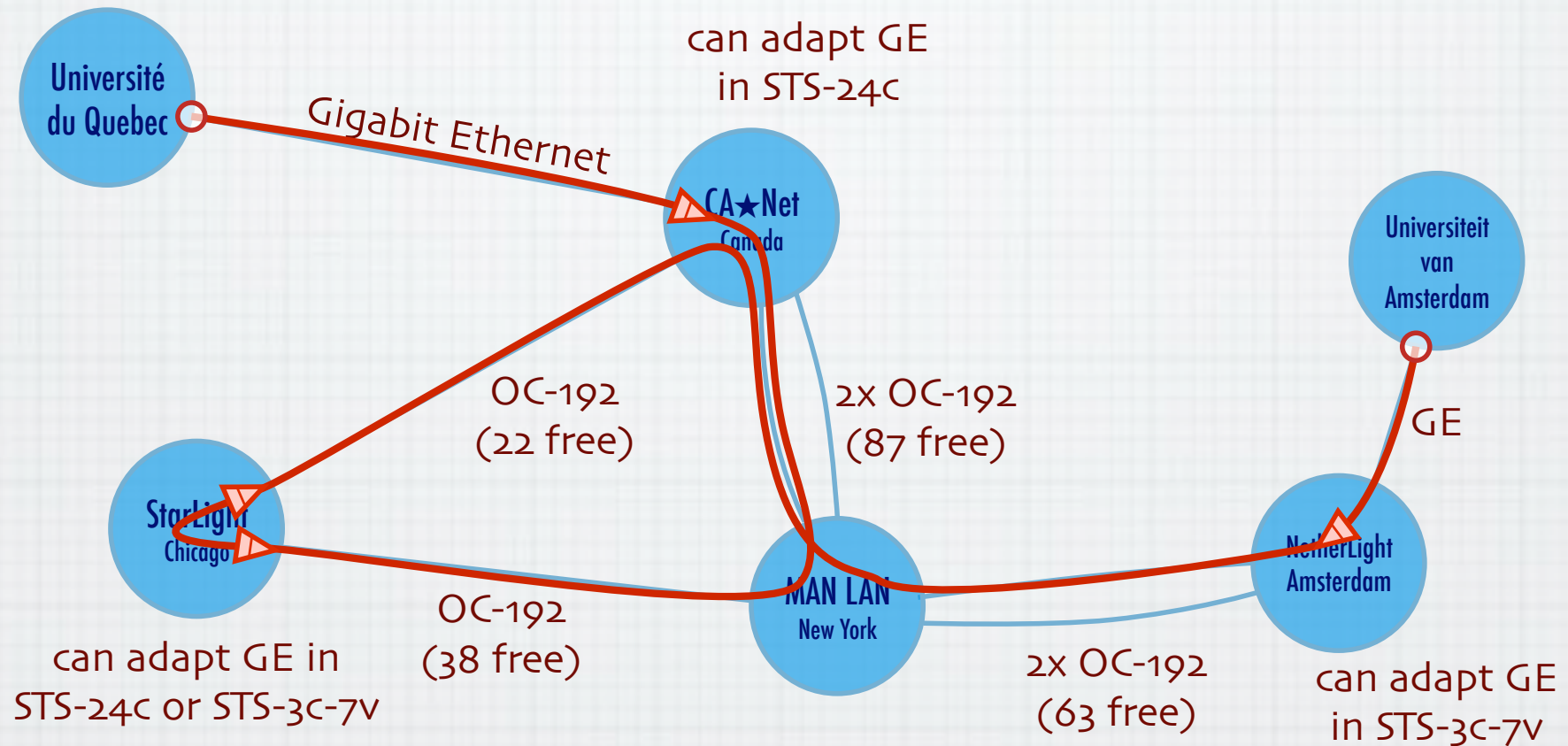
Multi-layer descriptions in NDL



A weird example

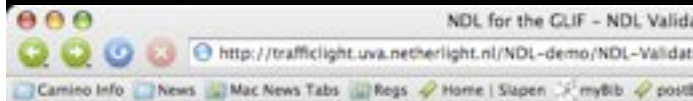


The result :-)



Thanks to Freek Dijkstra & team

NDL Generator and Validator



NDL for the GLIF - NDL Validator

NDL - Network Description Language - is an ontology for description of (hybrid) networks, air provisioning. The GLIF collaboration makes use of NDL to describe each individual domain, maps.

This page will provide you with tools to validate an NDL file. We provide here two types of validation:

- Syntax validation
- Content validation

Syntax validation

We can validate that the NDL file you generated is written following the latest NDL schema. You will get back feedback on its validity.

Please paste your NDL file below:

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:ndl="http://www.science.uva.nl/research/sne/ndl#"
  xmlns:geo="http://www.w3.org/2003/01/geo/wgs84_pos#"
  >
  <!-- Description of foo -->
  <ndl:Location rdf:about="#foo">
    <ndl:name>bar</ndl:name>
    <geo:lat>04</geo:lat>
    <geo:long>0</geo:long>
  </ndl:Location>
  <!-- Rem2 -->
  <ndl:Device rdf:about="#Rem2">
    <ndl:name>Rem2</ndl:name>
    <ndl:locatedAt rdf:resource="#foo"/>
    <ndl:hasInterface rdf:resource="#Rem2:eth0"/>
  </ndl:Device>
  <!-- GLIF -->
  <ndl:Domain rdf:about="2403.1.2.0">
```

Submit

Content validation

Often NDL files reference information contained in other files managed by others. Such as for example when an interface on a local device connects to an interface to a remote device. The content validator performs a few basic checks to see that the information contained in cross-referencing NDL files is consistent.

Please enter the URL of the NDL file to be validated:

Submit

Step 1 - Location

Indicate the name and a short description of the network that is going to be described in NDL.

Name Description

Provide also the latitude and the longitude of this location: this will aid the visualization programs.

Both latitude and longitude should use floating point notation.

Latitude Longitude

Step 2 - Devices

Indicate the name of all the devices present in the network. If you need to describe more than 3 devices just "Add a Device"

Device

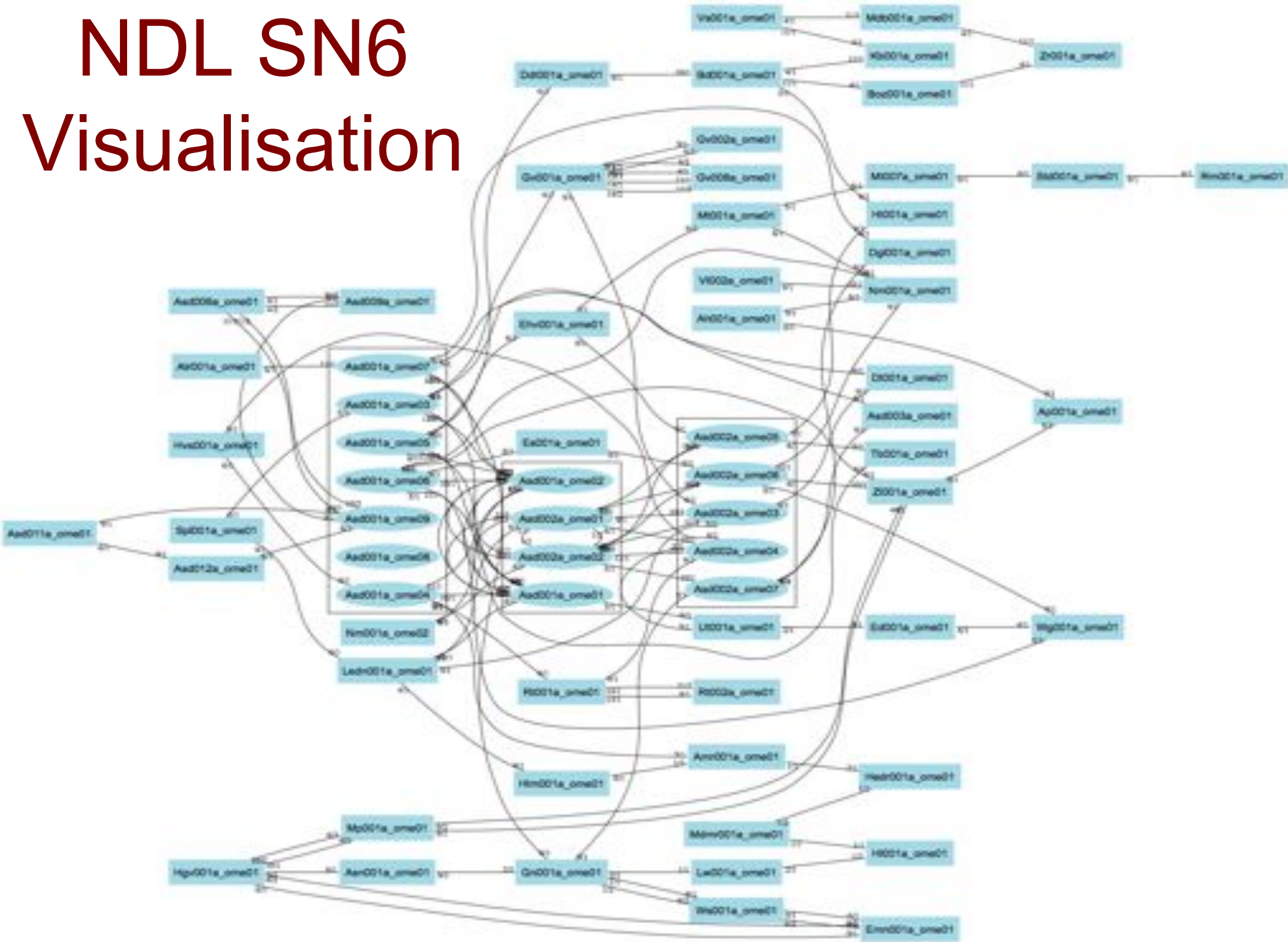
Device

Device

Add a Device

see <http://trafficlight.uva.netherlight.nl/NDL-demo/>

NDL SN6 Visualisation

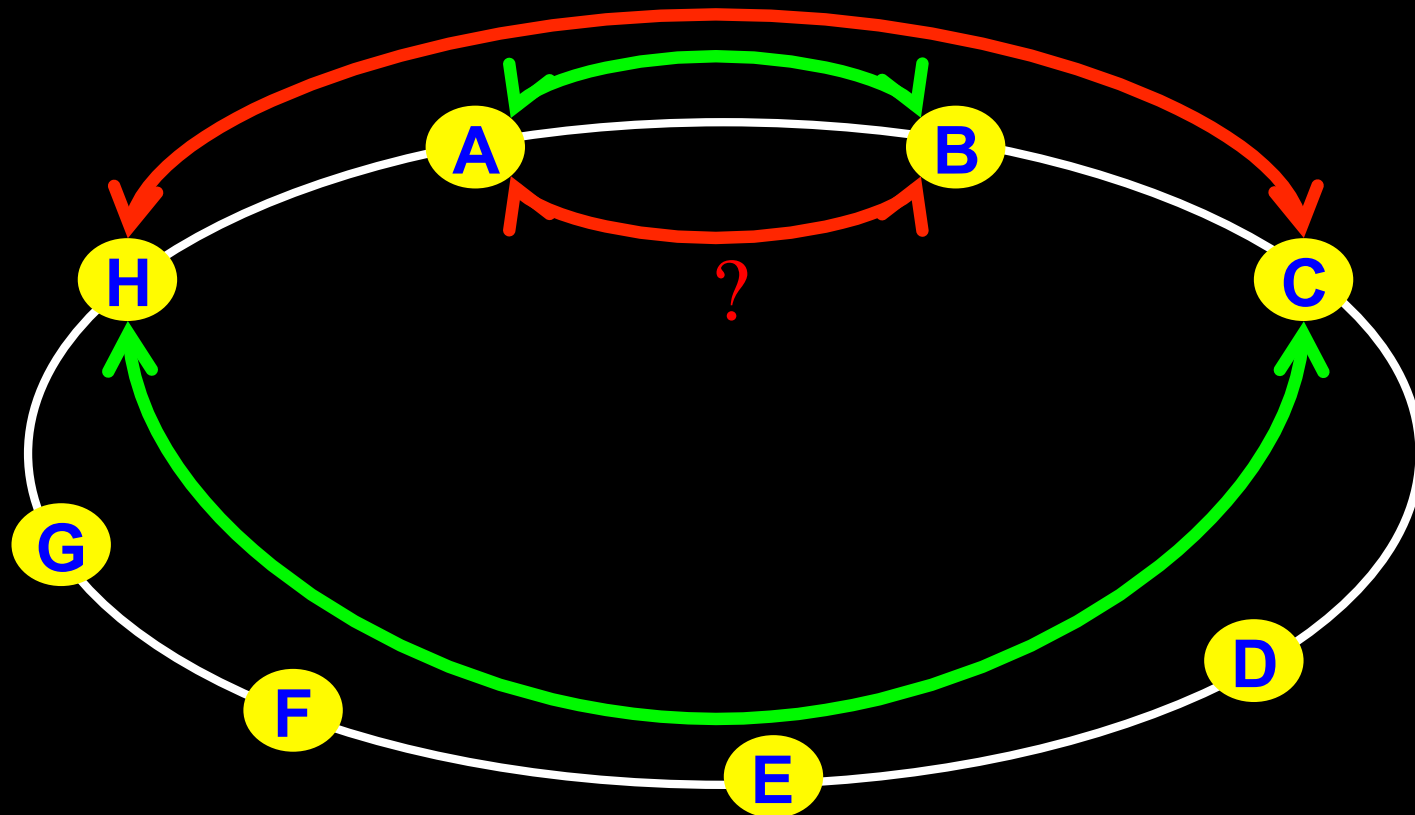


The Problem

I want HC and AB

Success depends on the order

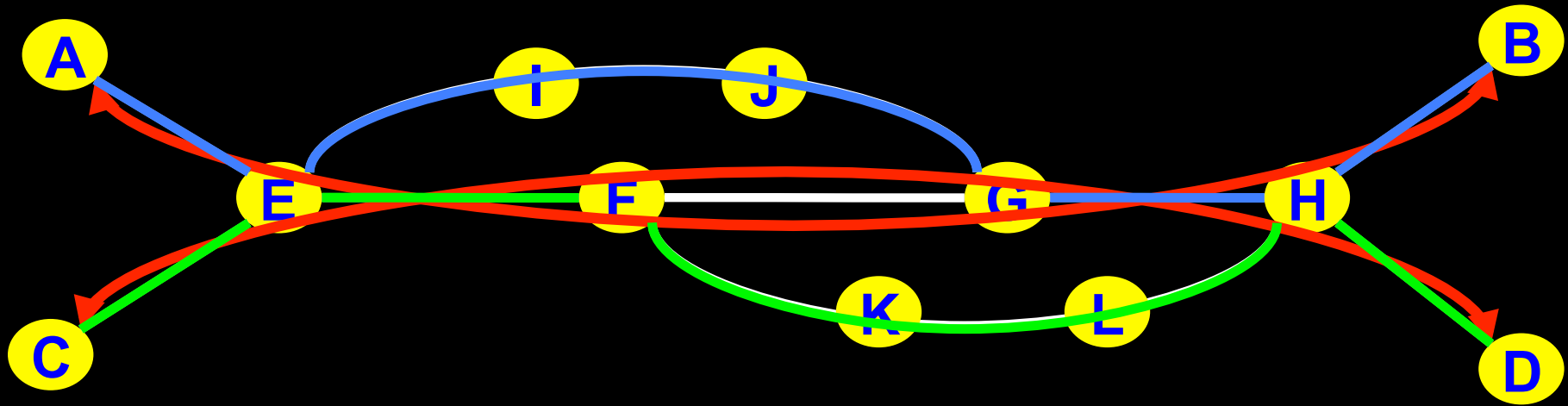
Wouldn't it be nice if I could request [HC, AB, ...]



Another one ☺

I want AB and CD

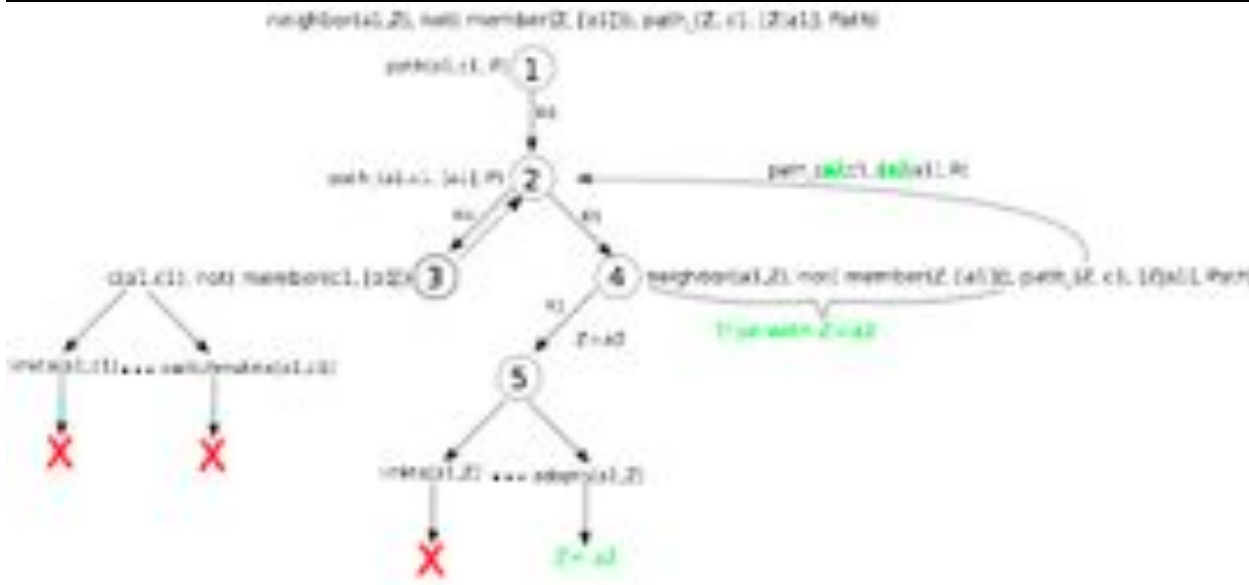
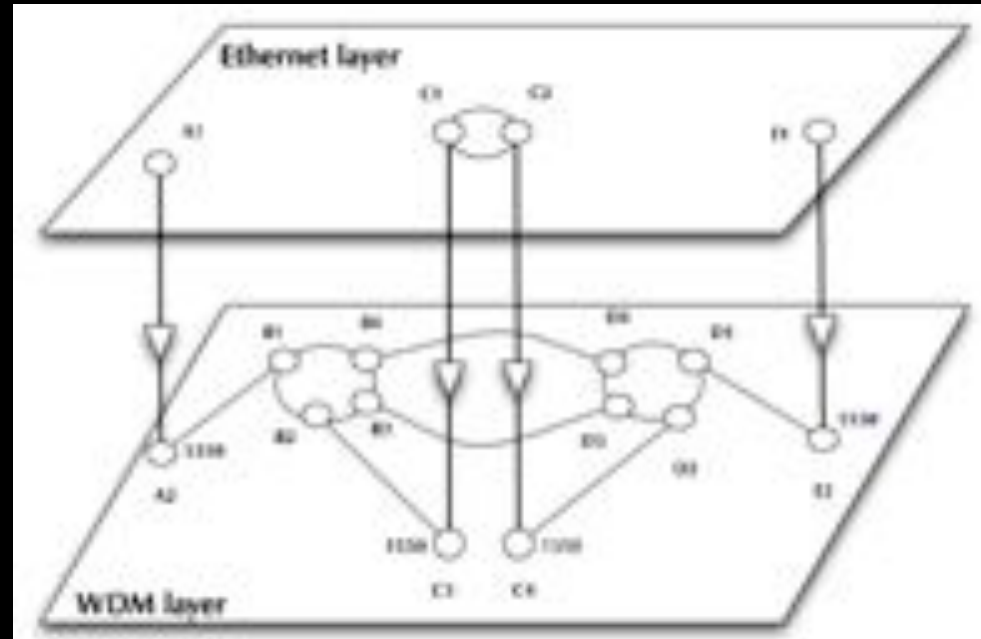
Success does not even depend on the order!!!



NDL + PROLOG

Research Questions:

- order of requests
- complex requests
- usable leftovers



•Reason about graphs

•Find sub-graphs that comply with rules

Multi layer multi domain networks

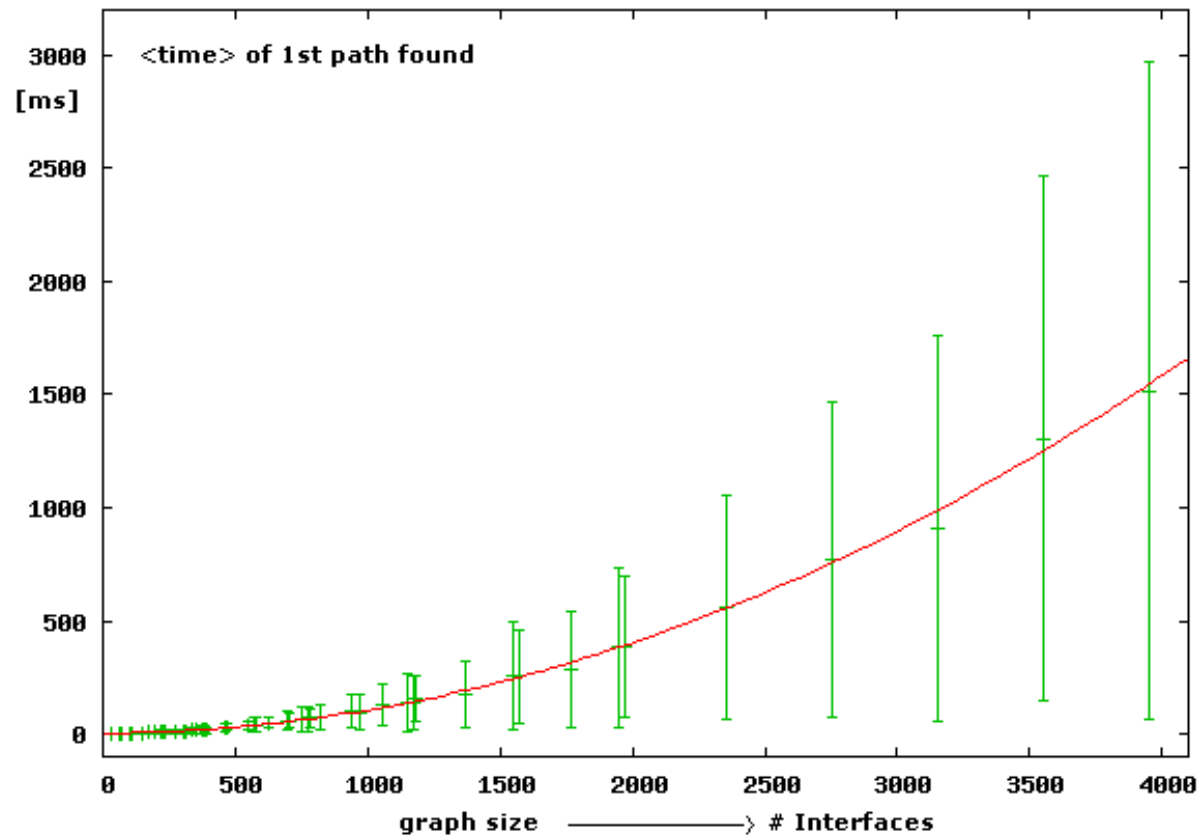
The networks for e-Science where applications use dedicated optical circuits.

Is declarative programming more suitable to find paths in multi-domain multi-layer networks? Especially in presence of constraints and complex requests?

Our approach:

1. We generate BA network graphs with a varying number of domains and nodes. Barabasi-Albert scale free graphs are a good representation of these networks.
2. We represent the graphs in NDL – Network Description Language, the RDF schemas.
3. We load the RDF files in Prolog and Python programs
4. We perform a modified DFS –Depth First Search- algorithm to find paths.

Single layer networks: results

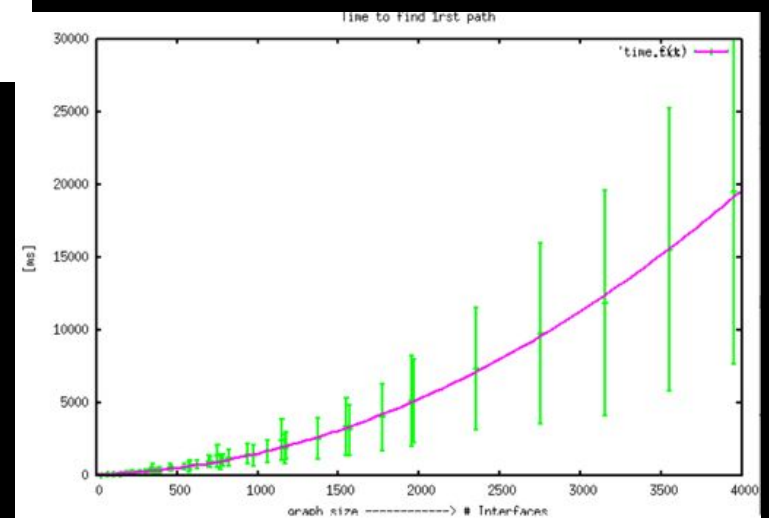


- Number of interfaces,
- given N nodes per domain D
- $4*(D-2) + D*4*(N-2)$ for $D > 2$

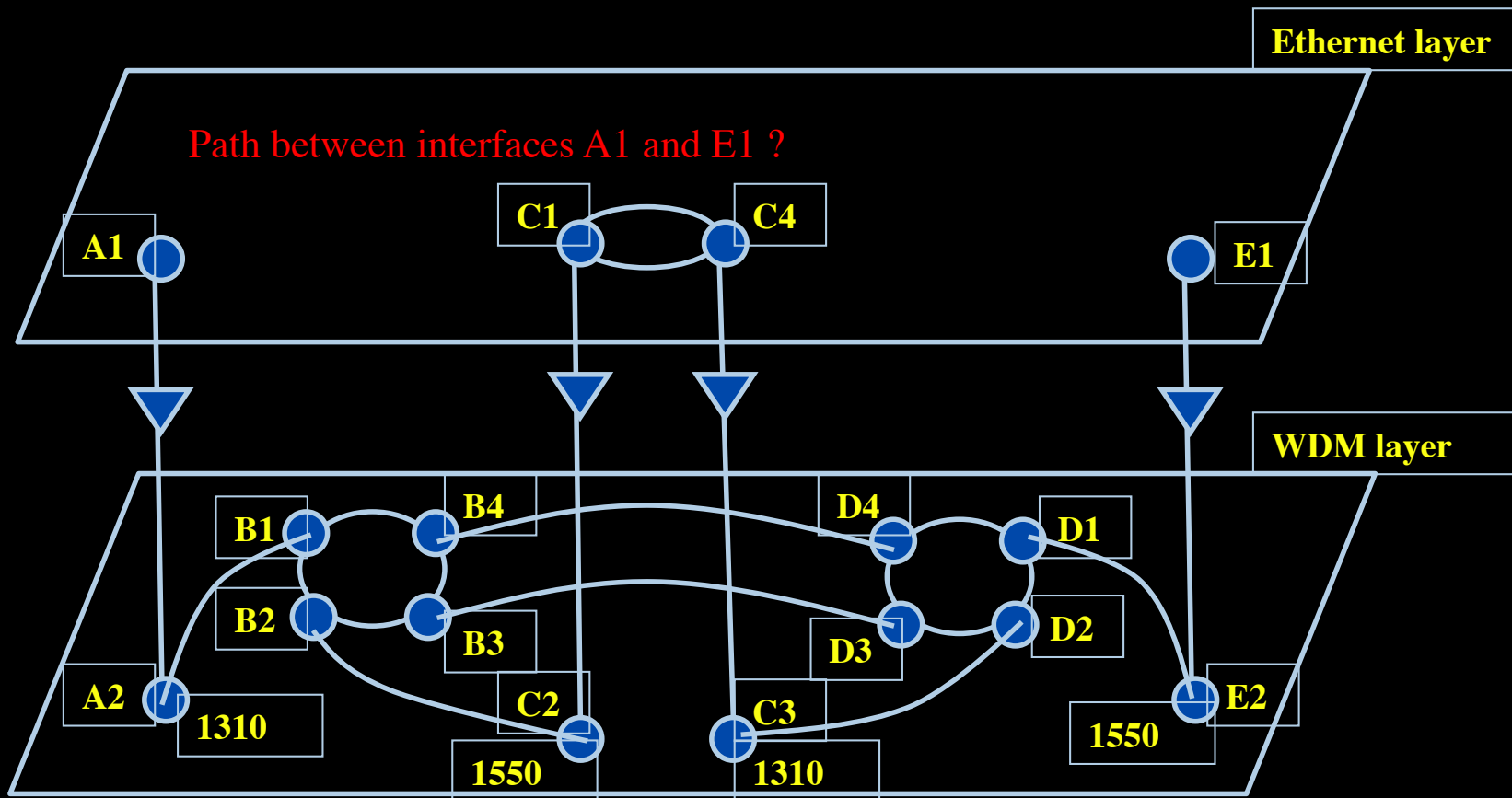
Pynt-based DFS

Prolog DFS

- Prolog time to find first path shorter than Python time.
- We observe a quadratic dependence.
- Length of paths found comparable.



Multi-layer network



Prolog rule:

`linkedto(Intf1, Intf2, CurrWav):-`

`rdf_db:rdf(Intf1, ndl:'layer', Layer),`

`Layer == 'wdm#LambdaNetworkElement',`

`rdf_db:rdf(Intf1, ndl:'linkedTo', Intf2),`

`rdf_db:rdf(Intf2, wdm:'wavelength', W2),`

`compatible_wavelengths(CurrWav, W2).`

`%-- is there a link between Intf1 and Intf2 for wavelength CurrWav ?`

`%-- get layer of interface Intf1 → Layer`

`%-- are we at the WDM-layer ?`

`%-- is Intf1 linked to Intf2 in the RDF file?`

`%-- get wavelength of Intf2 → W2`

`%-- is CurrWav compatible with W2 ?`

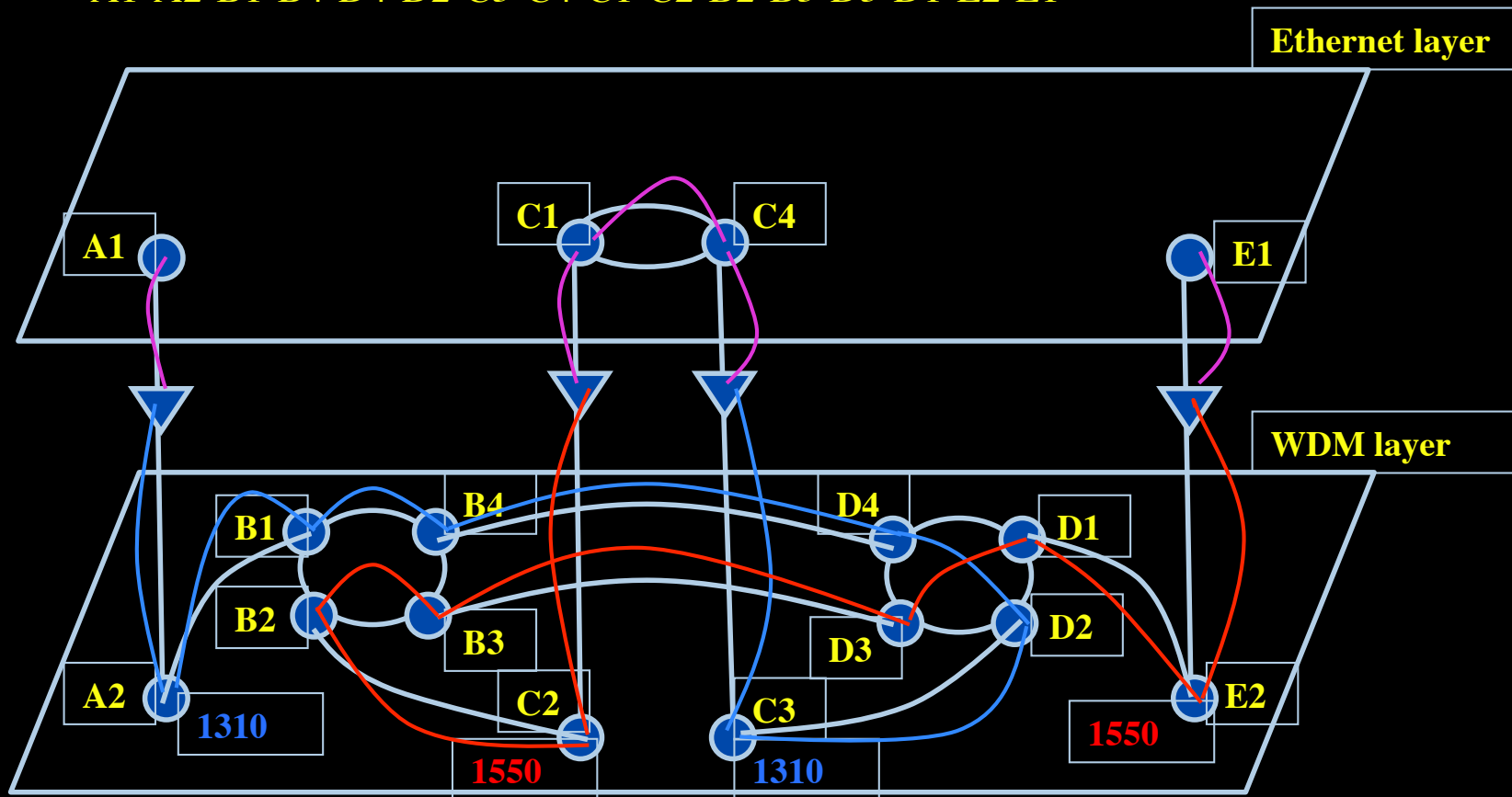
`linkedto(B4, D4, CurrWav)` is true for any value of `CurrWav`

`linkedto(D2, C3, CurrWav)` is true if `CurrWav == 1310`

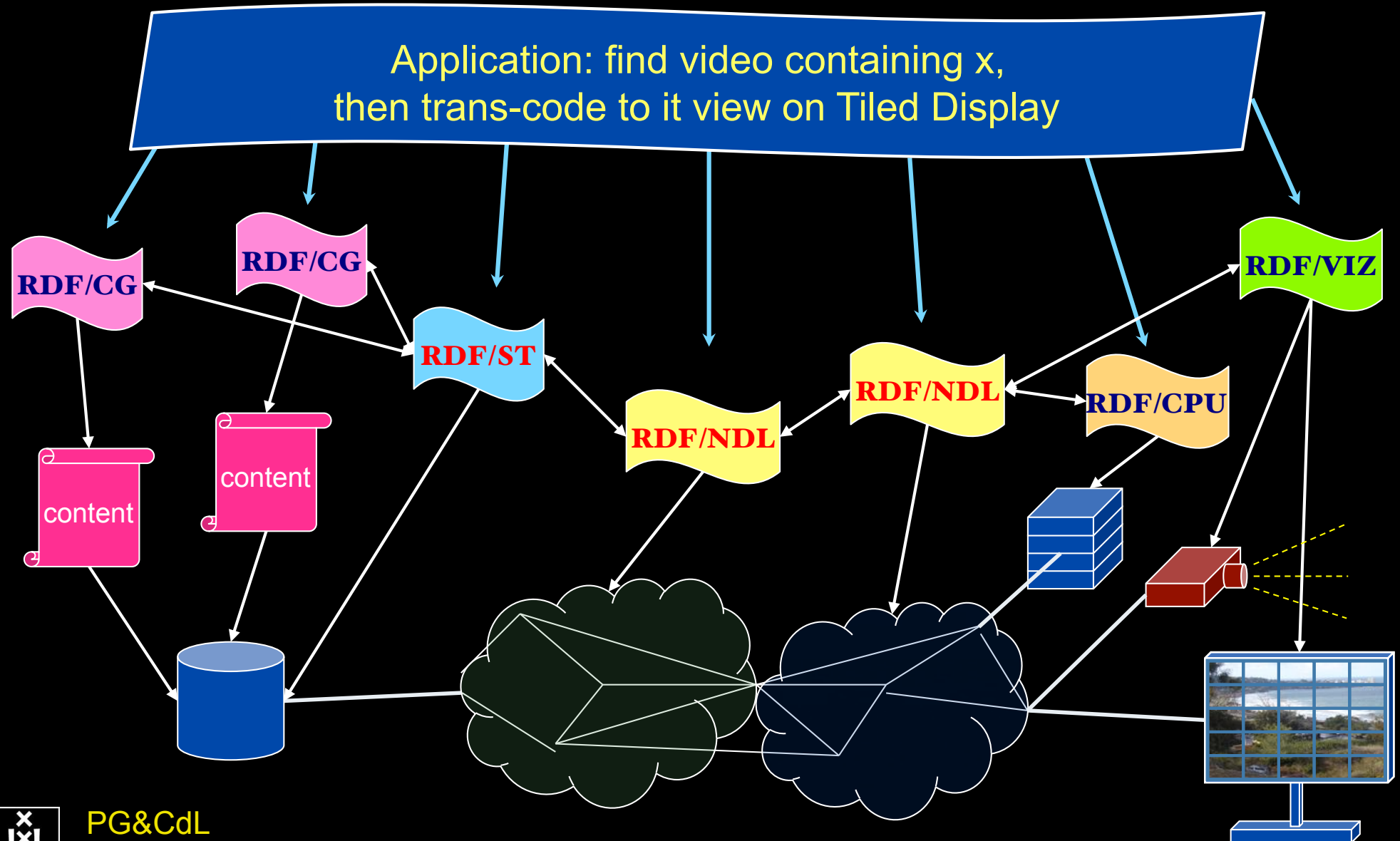
Multi-layer

Path between interfaces A1 and E1:

A1-A2-B1-B4-D4-D2-C3-C4-C1-C2-B2-B3-D3-D1-E2-E1



RDF describing Infrastructure



Applications and Networks become aware of each other!

CineGrid Description Language

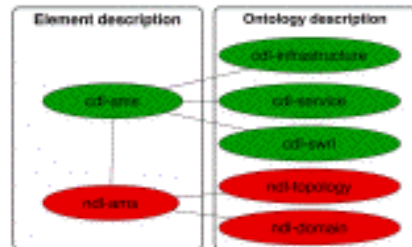
CineGrid is an initiative to facilitate the exchange, storage and display of high-quality digital media.

The CineGrid Description Language (CDL) describes CineGrid resources. Streaming, display and storage components are organized in a hierarchical way.

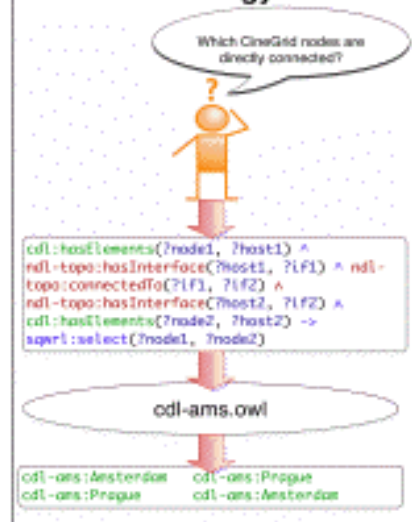
CDL has bindings to the NDL ontology that enables descriptions of network components and their interconnections.

With CDL we can reason on the CineGrid infrastructure and its services.

UML representation of CDL



SQWRL is used to query the Ontology.



CDL links to NDL using the *owl:SameAs* property. CDL defines the services, NDL the network interfaces and links. The combination of the two ontologies identifies the host pairs that support matching services via existing network connections.



Contents

1. The need for hybrid networking
2. StarPlane; a grid controlled photonic network
3. Cross Domain Authorization using Tokens
4. RDF/Network Description Language
5. Tera-networking
6. Programmable networks



TeraThinking

- What constitutes a Tb/s network?
- CALIT2 has 8000 Gigabit drops ?->? Terabit Lan?
- look at 80 core Intel processor
 - cut it in two, left and right communicate 8 TB/s
- think back to teraflop computing!
 - MPI makes it a teraflop machine
- massive parallel channels in hosts, NIC's
- TeraApps programming model supported by
 - TFlops -> MPI / Globus
 - TBytes -> OGSA/DAIS
 - TPixels -> SAGE
 - TSensors -> LOFAR, LHC, LOOKING, CineGrid, ...
 - Tbit/s -> ?



Need for discrete parallelism

- it takes a core to receive 1 or 10 Gbit/s in a computer
- it takes one or two cores to deal with 10 Gbit/s storage
- same for Gigapixels
- same for 100's of Gflops
- Capacity of every part in a system seems of same scale
- look at 80 core Intel processor
 - cut it in two, left and right communicate 8 TB/s
- massive parallel channels in hosts, NIC's
- Therefore we need to go massively parallel allocating complete parts for the problem at hand!



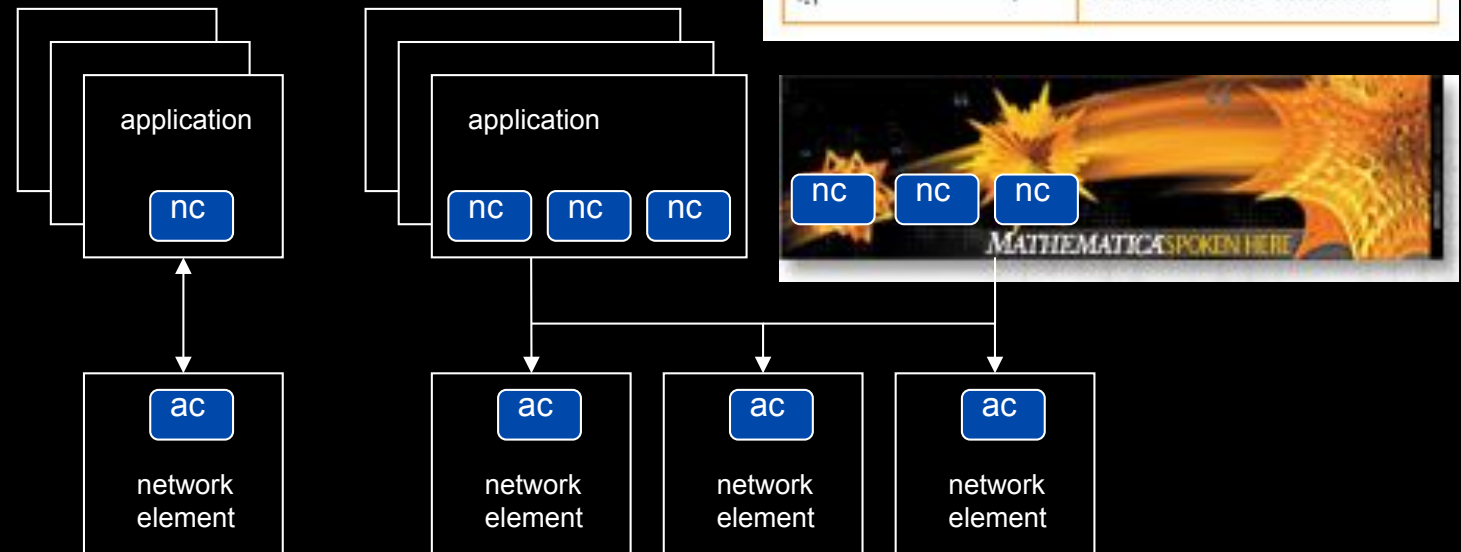
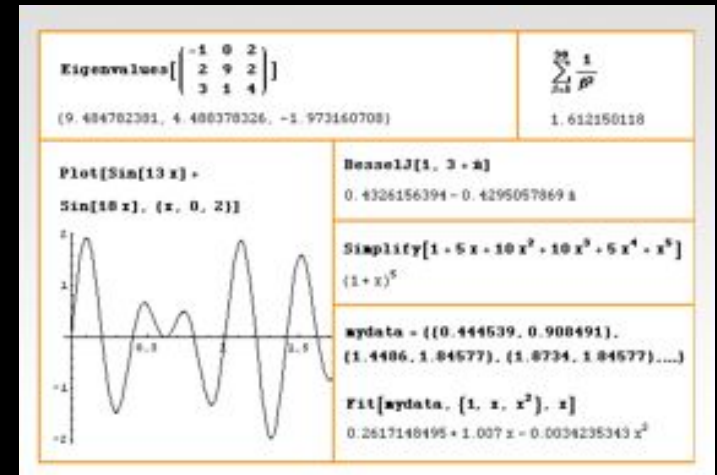
Contents

1. The need for hybrid networking
2. StarPlane; a grid controlled photonic network
3. Cross Domain Authorization using Tokens
4. RDF/Network Description Language
5. Tera-networking
6. Programmable networks



User Programmable Virtualized Networks allows the results of decades of computer science to handle the complexities of application specific networking.

- The network is virtualized as a collection of resources
- UPVNs enable network resources to be programmed as part of the application
- Mathematica, a powerful mathematical software system, can interact with real networks using UPVNs



Mathematica enables advanced graph queries, visualizations and real-time network manipulations on UPVNs

Topology matters can be dealt with algorithmically

Results can be persisted using a transaction service built in UPVN

Initialization and BFS discovery of NEs

```
Needs["WebServices`"]
<<DiscreteMath`Combinatorica`
<<DiscreteMath`GraphPlot`
InitNetworkTopologyService["edge.ict.tno.nl"]
```

Available methods:

```
{DiscoverNetworkElements, GetLinkBandwidth, GetAllLinks, Remote,
NetworkTokenTransaction}
```

```
Global`upvnverbose = True;
```

```
AbsoluteTiming[nes = BFSDiscover["139.63.145.94"];][[1]]
```

```
AbsoluteTiming[result = BFSDiscoverLinks["139.63.145.94", nes];][[1]]
```

```
Getting neighbours of: 139.63.145.94
```

```
Internal links: {192.168.0.1, 139.63.145.94}
```

```
(...)
```

```
Getting neighbours of: 192.168.2.3
```

Transaction on shortest path with tokens

```
nodePath = ConvertIndicesToNodes[
Internal links: {192.168.2.3, 192.168.3.4, 139.63.77.30, 139.63.77.49}
ShortestPath[
g,
Node2Index[nids, "192.168.3.4"],
Node2Index[nids, "139.63.77.49"],
nids];
```

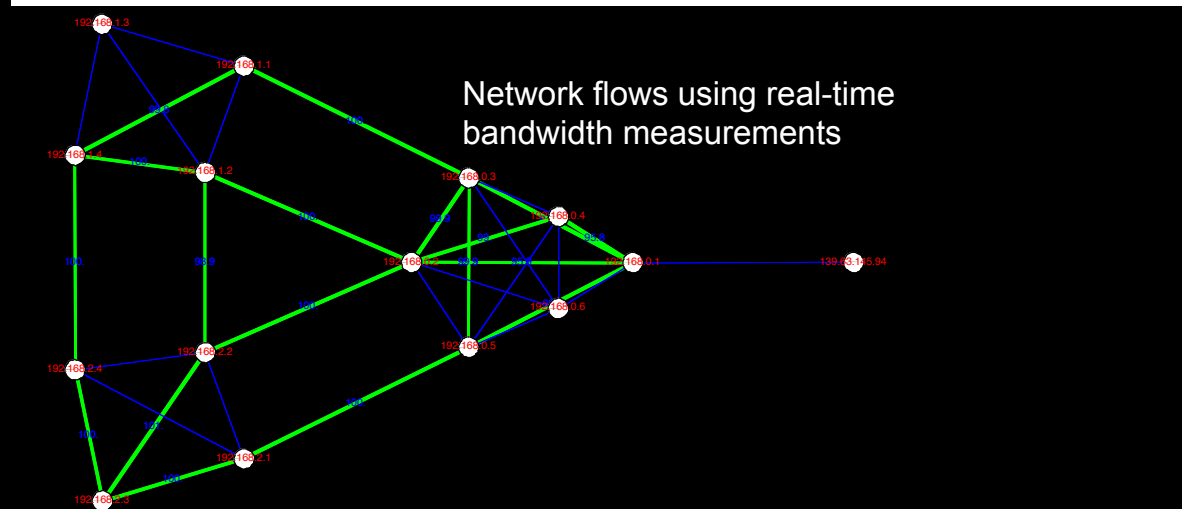
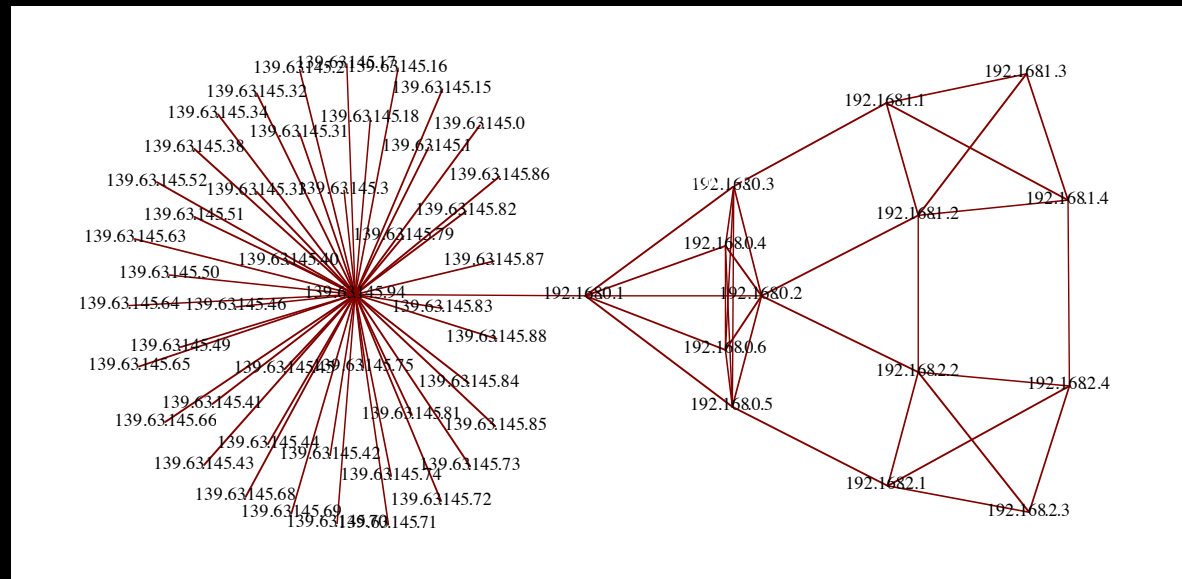
```
Print["Path: ", nodePath];
```

```
If[NetworkTokenTransaction[nodePath, "green"]==True,
Print["Committed"], Print["Transaction failed"]];
```

```
Path:
```

```
{192.168.3.4, 192.168.3.1, 139.63.77.30, 139.63.77.49}
```

```
Committed
```



ref: Robert J. Meijer, Rudolf J. Strijkers, Leon Gommans, Cees de Laat, User Programmable Virtualized Networks, accepted for publication to the IEEE e-Science 2006 conference Amsterdam.

StarPlane



TouchTable Demonstration @ SC08



Interactive programmable networks



Questions ?