Hybrid Networking for eScience

Cees de Laat

EU

SURFnet

SURF-eScience

NWO

University of Amsterdam
LOFAR as a Sensor Network

LOFAR is a large distributed research infrastructure:

- **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

20 flops/byte

2 Tflops/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 + uplink to all

C. E-Science applications, distributed data processing, all sorts of grids
   Need very fat pipes, limited multiple Virtual Organizations, P2P, few to few

For the Netherlands 2010
\[ \Sigma A = \Sigma B = \Sigma C \approx \text{Tb/s} \]

However:
- A -> all connects
- B -> on several
- C -> just a few (SP, LHC, LOFAR)

Ref: Cees de Laat, Erik Radius, Steven Wallace, "The Rationale of the Current Optical Networking Initiatives"
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 ≈ 2-3 k$/port  
L2 ≈ 2-5 k$/port  
L3 ≈ 50+ k$/port
How low can you go?

Application Endpoint A

Router
Ethernet
SONET
DWDM
Fiber

Local Ethernet
WSS
Regional dark fiber
15454
6500
HDXc

Application Endpoint B

POS
Trans-Oceanic
StarLight
GLIF

CERN
NetherLight
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
Alien light
From idea to realisation!

40Gb/s alien wavelength transmission via a multi-vendor 10Gb/s DWDM infrastructure

Alien wavelength advantages
- Direct connection of customer equipment → cost savings
- Avoid DEO regeneration → power savings
- Faster time to service → time savings
- Support of different modulation formats → extend network lifetime

Alien wavelength challenges
- Complex end-to-end optical path engineering in terms of linear (i.e., OSNR, dispersion) and non-linear (FWM, SPM, XPM, Raman) transmission effects for different modulation formats.
- Complex interoperability testing.
- End-to-end monitoring, fault isolation and resolution.
- End-to-end service activation.

In this demonstration we will investigate the performance of a 40Gb/s PM-QPSK alien wavelength installed on a 10Gb/s DWDM infrastructure.

New method to present fiber link quality, FoM (Figure of Merit)
In order to quantify optical link grade, we propose a new method of representing system quality: the FOM (Figure of Merit) for concatenated fiber spans.

Conclusions
- We have investigated experimentally the all-optical transmission of a 40Gb/s PM-QPSK alien wavelength via a concatenated native and third party DWDM system that both were carrying live 10Gb/s wavelengths.
- The end-to-end transmission system consisted of 1056 km of TWR5 (TrueWave Reduced Slope) transmission fiber.
- We demonstrated error-free transmission (i.e., BER below 10-15) during a 24 hour period.
- More detailed system performance analysis will be presented in an upcoming paper.
Diagram for SAGE video streaming to ATS

Lab 10, Nortel

- SAGE Display
- SAGE Servers
- MERS
- User
- Regular Browser

Netherlight Canarie

Internet

UvA, Amsterdam

- comp clusters
- Traffic Generators
- Content Portal
- Streaming Server
- 100 TB Storage

Content Request

Content Choice
Experimental Data

PBT is **SIMPLE** and **EFFECTIVE** technology to build a shared Media-Ready Network.

- **Sage without background traffic**
- **Sage with background traffic**

10 Second Traffic bursts with No PBT

10 Second Traffic bursts with PBT
Complexity
Network Description Language

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

```
Subject: name
Predicate: description
Object: locatedAt
```

```
Subject: hasInterface
Predicate: capacity
Object: encodingLabel
```

```
Subject: connectedTo
Predicate: encodingType
Object: hasInterface
```

```
Subject: device
Predicate: interface
Object: link
```
<?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:501/2"/>
        <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"/>
        <ndl:hasInterface rdf:resource="#tdm3.amsterdam1.netherlight.net:503/3"/>
        <ndl:hasInterface rdf:resource="#tdm3.amsterdam1.netherlight.net:503/4"/>
        <ndl:hasInterface rdf:resource="#tdm3.amsterdam1.netherlight.net:504/1"/>
        <ndl:hasInterface rdf:resource="#tdm3.amsterdam1.netherlight.net:504/2"/>
        <ndl:hasInterface rdf:resource="#tdm3.amsterdam1.netherlight.net:504/3"/>
        <ndl:hasInterface rdf:resource="#tdm3.amsterdam1.netherlight.net:504/4"/>
    </ndl:Device>
    <!-- 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>
</rdf:RDF>
Topology Aggregation
Topology Aggregation - Initial
Multi-layer descriptions in NDL

IP layer

Ethernet layer

STS layer

OC-192 layer

SONET switch with Ethernet intf.

End host

SONET switch

Ethernet & SONET switch

SONET switch

SONET switch with Ethernet intf.

End host

Université du Quebec

CAhesive Net Canada

StarLight Chicago

MAN LAN New York

NetherLight Amsterdam

Universiteit van Amsterdam
A weird example

Thanks to Freek Dijkstra & team
A weird example

can adapt GE in STS-24c

can adapt GE in STS-24c or STS-3c-7v

2x OC-192 (87 free)

2x OC-192 (63 free)

2x OC-192 (22 free)

can adapt GE in STS-3c-7v

Gigabit Ethernet

Thanks to Freek Dijkstra & team
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-domain 2-layer networks

How do multi-domain 2-layer networks look like?

Guess: Projection algorithm (2-layer: Ethernet / WDM)

Steps:
1. Generate a multi-domain graph by BA-algorithm
2. Generate a graph for each domain by BA-algorithm
3. For each domain graph project random nodes onto WDM layer
4. Connect the domains at each layer according to the multi-domain graph
5. Assign random wavelengths to the adaptation links

Advantage:
• Number of adaptations determined by the degree of the projected nodes
• Multi-domain Ethernet-layer as well as the multi-domain WDM-layer graph are not necessarily connected.

Input parameters:
• Number domains, number of nodes (devices) per domain
• Ratio of Ethernet-devices over WDM-devices per domain
• Distribution of wavelength
Multi-domain 2-layer networks

Projection algorithm

BA-algorithm to generate a graph for each domain
Project random nodes onto WDM layer
**Prolog rule:**

\[
\text{linkedto}(\text{Intf1}, \text{Intf2}, CurrWav) :- \\
\text{rdf}\_\text{db}::\text{rdf}(\text{Intf1}, \text{ndl}:'layer', \text{Layer}), \\
\text{Layer} == \text{wdm}\#\text{LambdaNetworkElement}, \\
\text{rdf}\_\text{db}::\text{rdf}(\text{Intf1}, \text{ndl}:'\text{linkedTo}', \text{Intf2}), \\
\text{rdf}\_\text{db}::\text{rdf}(\text{Intf2}, \text{wdm}:'\text{wavelength}', W2), \\
\text{compatible}\_\text{wavelengths}(\text{CurrWav}, W2).
\]

%-- is there a link between Intf1 and Intf2 for wavelength CurrWav ?
%-- get layer of interface Intf1 \rightarrow Layer
%-- are we at the WDM-layer ?
%-- is Intf1 linked to Intf2 in the RDF file?
%-- get wavelength of Intf2 \rightarrow W2
%-- is CurrWav compatible with W2 ?

\[
\text{linkedto}(B4, D4, CurrWav) \text{ is true for any value of CurrWav}
\]

\[
\text{linkedto}(D2, C3, CurrWav) \text{ is true if CurrWav == 1310}
\]
Multi-layer Network PathFinding

Path between interfaces A1 and E1:
A1-A2-B1-B4-D4-D2-C3-C4-C1-C2-B2-B3-D3-D1-E2-E1

Scaling: Combinatorial problem
# Prolog pathfinding results

<table>
<thead>
<tr>
<th>#Domains (#Ether:#WDM) (&lt;#Intf&gt;)(&lt;#Adap&gt;)</th>
<th>Prolog time [ms] (\mu(\sigma))</th>
<th>Timeouts</th>
<th>Success %</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 (9:6)(55)(11)</td>
<td>20(4)</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>4 (48:32)(377)(73)</td>
<td>2620(8245)</td>
<td>74</td>
<td>92.6</td>
</tr>
<tr>
<td>4 (96:64)(771)(147)</td>
<td>6592(11802)</td>
<td>207</td>
<td>79.3</td>
</tr>
<tr>
<td>3 (9:6)(55)(11)</td>
<td>20(4)</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>4 (48:32)(377)(73)</td>
<td>1303(5052)</td>
<td>22</td>
<td>97.8</td>
</tr>
<tr>
<td>4 (96:64)(771)(147)</td>
<td>3910(10045)</td>
<td>51</td>
<td>94.9</td>
</tr>
<tr>
<td>3 (9:6)(55)(11)</td>
<td>20(4)</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>4 (48:32)(377)(73)</td>
<td>755(3210)</td>
<td>8</td>
<td>98.9</td>
</tr>
<tr>
<td>4 (96:64)(771)(147)</td>
<td>3240(9052)</td>
<td>38</td>
<td>96.1</td>
</tr>
</tbody>
</table>

DFS path constraints:

- Number of different wavelength
- No max #wav
- #wav ≤3
- #wav ≤2
## Prolog pathfinding results

Parallel calls: A->B and B->A

### Projection: A->B

<table>
<thead>
<tr>
<th>#Domains (#Ether:#WDM) (#Intf) (#Adap)</th>
<th>Prolog time [ms]</th>
<th>Timeouts</th>
<th>Success %</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( \mu(\sigma) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 (9:6)(55)(11)</td>
<td>20(4)</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>4 (48:32)(377)(73)</td>
<td>755(3210)</td>
<td>8</td>
<td>98.9*</td>
</tr>
<tr>
<td>4 (96:64)(771)(147)</td>
<td>3240(9052)</td>
<td>38</td>
<td>96.1*</td>
</tr>
</tbody>
</table>

### Projection: first of A->B and B->A

<table>
<thead>
<tr>
<th>#Domains (#Ether:#WDM) (#Intf) (#Adap)</th>
<th>Prolog time [ms]</th>
<th>Timeouts</th>
<th>Success %</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( \mu(\sigma) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 (9:6)(55)(11)</td>
<td>19(1)</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>4 (48:32)(377)(73)</td>
<td>144(486)</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>4 (96:64)(771)(147)</td>
<td>601(2722)</td>
<td>2</td>
<td>99.6*</td>
</tr>
</tbody>
</table>

*false negatives also taken into account
Performance Prolog Depth-First Search

Performance drop mainly due to Timeout limit
False negatives due to max #wavelengths clip less than 1% of #paths
DAS3 cluster

Time Prolog Depth-First Search

- \#Eth. dev. / \#WDM dev. = 3/2
- \#Eth. dev. / \#WDM dev. = 2/3

clips: TIMEOUT LIMIT=180 sec.
max \#wavs per path = 2

Graph showing the relationship between the number of interfaces and time in milliseconds.
Standardization

- OGF-NML is slowly progressing
  - Schema Document
- OGF-NSI is working frantically
  - Terminology Glossary
  - Architecture Document
  - NSI Protocol Document
• Network descriptions are in NDL

• Use Prolog, a *logical programming* language:
  – **clauses**: facts and rules
  – **goals**: reached through backward chaining (goal-driven)

• Multi-layer pathfinding is a combinatorial bomb.

• Need features of networks to force Prolog to backtrack if it looks for an unnecessary long path.

• Introducing features (heuristics) speeds up the pathfinding but may lead to false negatives too.

• Constructed large set of multi-domain 2-layer networks of different sizes with the Barabási-Albert algorithm.

• Studied shortest paths between randomly chosen src-dst pairs by means of an memory unfriendly algorithm.
RDF describing Infrastructure

Application: find video containing x, then trans-code to it view on Tiled Display

RDF/CG

RDF/CG

content

content

RDF/ST

RDF/NDL

RDF/NDL

RDF/CPU

RDF/VIZ

PG&CdL
Applications and Networks become aware of each other!
Use AAA concept to split (time consuming) service authorization process from service access using secure tokens in order to allow fast service access.
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 adds a 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 returns 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
Hybrid computing

- Routers ↔ Supercomputers
- Ethernet switches ↔ Grid & Cloud
- Photonic transport ↔ GPU’s

What matters:
- Energy consumption/multiplication
- Energy consumption/bit transported
Questions?

CookReport
feb 2009 and feb-mar 2010

november ’08
interview with
Kees Neggers (SURFnet),
Cees de Laat (UvA)

and furthermore
on november ’09

Wim Liebrandt (SURF),
Bob Hertzberger (UvA) and
Hans Dijkman (UvA)

BSIK projects
GigaPort &
VL-e / e-Science

ext.delaat.net