Practical demonstration of Network Descriptions

February 08, 2006
GLIF meeting, Albuquerque.

Andree Toonk

Network specialist
SARA - High Performance Networking
Amsterdam, The Netherlands

Andree@sara.nl

SARA Computing & Networking Services

Departement High performance networking:

- SURFnet5 & SURFnet6 network operations centre (NOC)
- **Netherlight** network operations centre (NOC)
- Partner in Research on Networks GigaPort project

National Supercomputers

Visualization in the CAVE

Resource Description Format (RDF)

- Presentation Jeroen van der Ham (UvA) Glif meeting september 2005
  See: http://www.glif.is/meetings/2005/tech/vdham-rdf.pdf

- Resource Description Format (RDF) is a Semantic Web technique.

Can be used to:

1. *Provide an overview of resources*
2. *Make path discovery easier*
3. Do simple problem detection

Example network description

![sara]

- Netherlight network description in RDF format:

```
<!- Description of 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:503/3"/>
</ndl:Device>
```

Example network description

- Netherlight network description in RDF format:

```
<!- Description of 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:503/3"/>
</ndl:Device>


<!- Description of interfaces 503/3  TDM3.amsterdam1.netherlight.net -->
<ndl:Interface rdf:about="#tdm3.amsterdam1.netherlight.net:503/3">
    <ndl:name>tdm3.amsterdam1.netherlight.net:POS503/3</ndl:name>
    <ndl:connectedTo rdf:resource="#tdm1.geneva1.netherlight.net:5/1"/>
    <rdf:capacity rdf:resource="#OC192"/>
 </ndl:Interface>
```

Provide an overview of resources

- Create an overview / list of devices and interfaces
- Create an overview / list of connections (this is used for the dot file)
- Visualization of the network (demonstration)
- Create a graph (demonstration application)

This can all be done by using SPARQL queries

Provide an overview of resources

- Provide an overview of resources by visualization

- Graphviz - Graph Visualization Software
  - uses .dot file as input

- Relatively easy to generate a .dot input file from an RDF file
- Python script to generate a .dot file from an RDF file*

- Then generate a png,jpg,gif and lots more..
  - ```
    dot -o netherlight.png -Tpng netherlight.dot
    ```

  Let's see the result…. ☺

  * Author: Jeroen van der Ham, University of Amsterdam

Provide an overview of resources

Visualization of Netherlight network elements.

(Netherlight elements only)

tdm4.amsterdam1.netherlight.net

5/1

501/1

tdm1.amsterdam1.netherlight.net 12/1   501/2   tdm3.amsterdam1.netherlight.net 504/4   6/1   tdm1.geneva1.netherlight.net
6/1   505/1   5/1
                                       503/3

Visualization of Netherlight network elements including edge devices on next slide

tdm4.amsterdam1.netherlight.net

tdm1.geneva1.netherlight.net

tdm3.amsterdam1.netherlight.net

tdm1.amsterdam1.netherlight.net

#Nordunet:6/1

#RIPN-Gloriad:0/0

#br1.amsterdam1.surf.net:5/1

#Force10-Cern:0/0

#tdm1.Praha1:6/1

#UvA-Force10:4/1

#Starlight-HDXc:502/2

#Manlan-HDXc:503/4

#Starlight-HDXc:503/2

#Manlan-HDXc:504/1

#GEANT-M160:0/0

#Manlan-HDXc:503/2

#UvA-Force10:4/0

5/1
501/1
504/4
6/1
6/1
503/3
5/1
505/1
12/1
501/2
13/1
17/1
5/1
16/1

Path discovery

- Provisioning a circuit through a 'big' network can be time consuming
    1) finding a (protected) path across the network
    2) enough time slots available?

Might sound easy but can be difficult ➔

- Tool written by SARA's network research group to automate this
- Uses an RDF network description as topology description
- Uses a MySQL database that has knowledge of all current cross connects in the network

Path discovery

■ Tool written in Perl *

usage: get_path ne1 slot1-port1 ne2 slot2-port2 #timeslots

shortest path between ne1 and ne2

e.g. `./get_path.pl \`

```
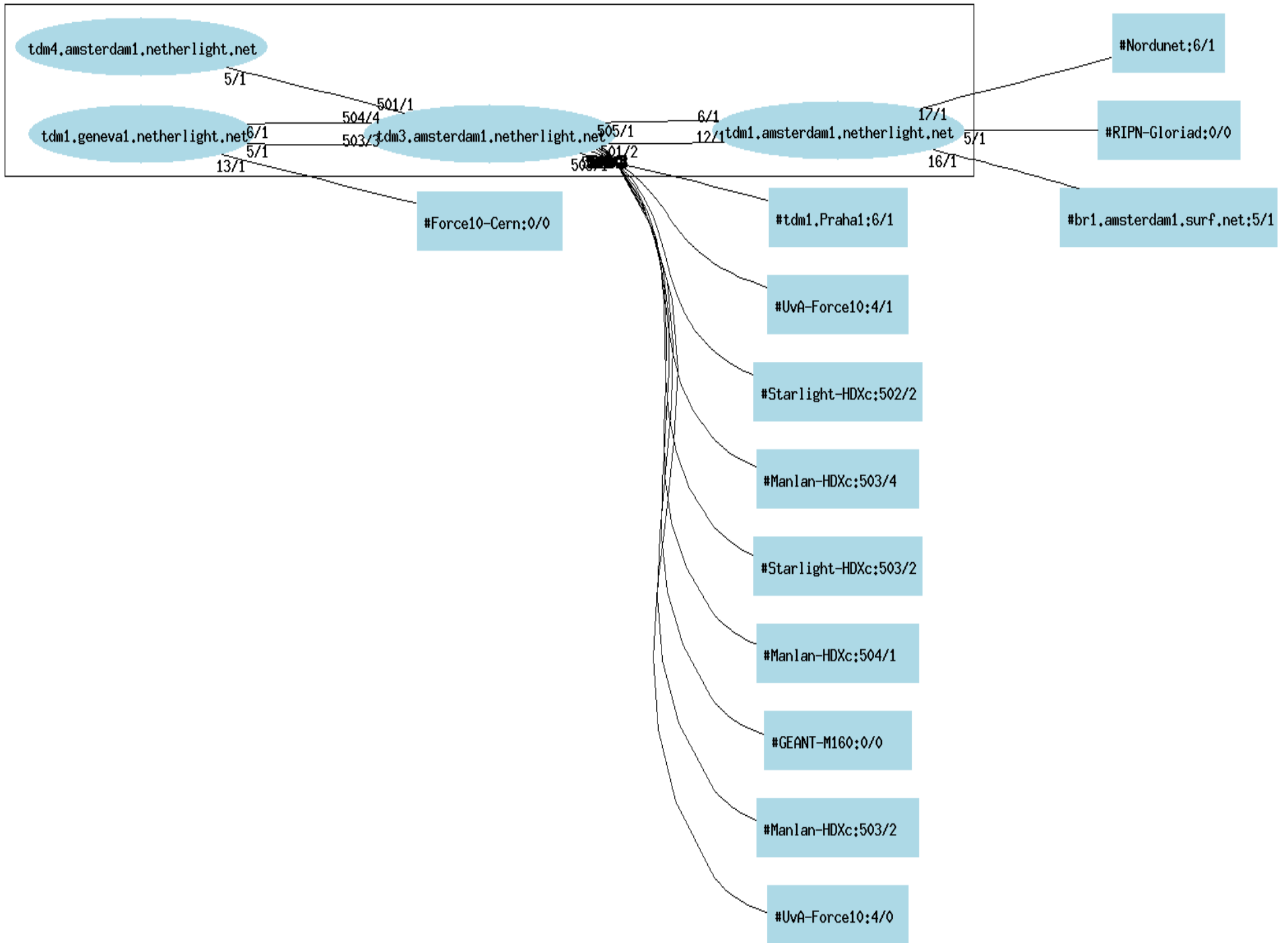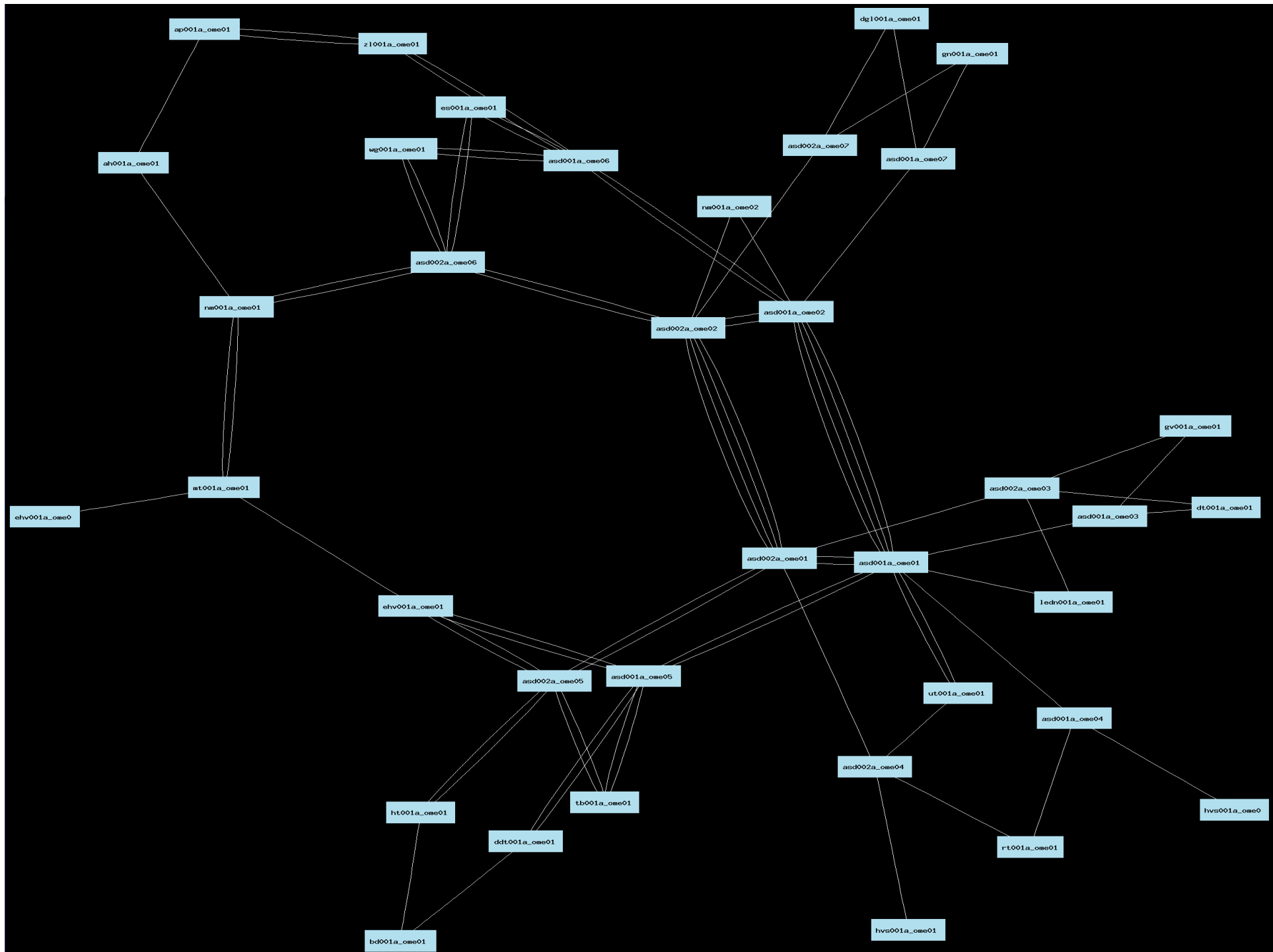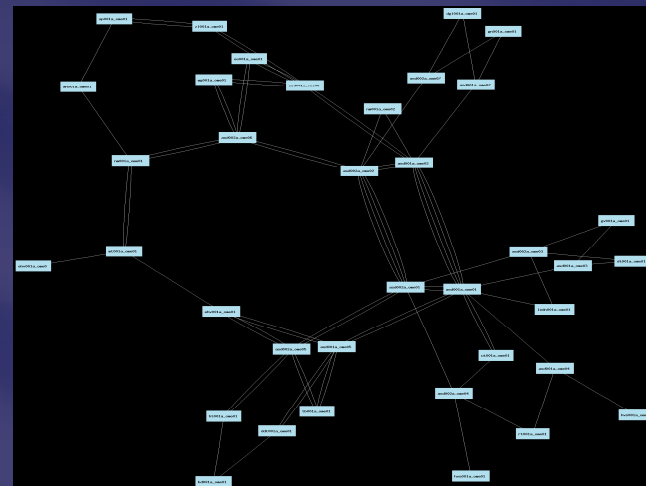tdm1.amsterdam1.netherlight.net 6-1 \
tdm1.geneva1.netherlight.net  3-5 \
21
```

The example above will show you how to provision a circuit between
tdm1.geneva1.netherlight.net interface: 6/1 and tdm1.amsterdam1.netherlight.net interface
3/5 with a capacity of 21sts containers (1Gb/s using GFP/VCAT)

*Authors:
- Ronald van der Pol (SARA's network research group)
- Andree Toonk (SARA's network research group)

Provisioning tool

- With the RDF information we build a graph representing the network.
  - SPARQL query that gets all connected interfaces
  - Check if these interfaces have enough free timeslots available (MySQL)
  - Add the two interfaces as vertices to the graph
  - Add the link between them as an edge of the graph
  - Add edges to the graph for all internal connections between all the interfaces of one network element

  - **Now a graph representing the network has been built.**
  - Use the Dijkstra algorithm to find the shortest path
  - Returns all the information to create the (VC4 / STS3c) cross connects on the network elements,
    - Includes: network elements, slots, ports, timeslots (MySQL) all the information to create the cross connects on the network elements,

```
$ ./get_path.pl tdm3.amsterdam1.netherlight.net 504-3 tdm1.amsterdam1.netherlight.net 17-1 9
path A:
-----
tdm3.amsterdam1.netherlight.net-504-3 1 tdm3.amsterdam1.netherlight.net-501-2 49
tdm3.amsterdam1.netherlight.net-504-3 4 tdm3.amsterdam1.netherlight.net-501-2 52
tdm3.amsterdam1.netherlight.net-504-3 7 tdm3.amsterdam1.netherlight.net-501-2 55
-----
tdm1.amsterdam1.netherlight.net-12-1 49 tdm1.amsterdam1.netherlight.net-17-1 25
tdm1.amsterdam1.netherlight.net-12-1 52 tdm1.amsterdam1.netherlight.net-17-1 28
tdm1.amsterdam1.netherlight.net-12-1 55 tdm1.amsterdam1.netherlight.net-17-1 31


path B:
-----
tdm3.amsterdam1.netherlight.net-504-3 1 tdm3.amsterdam1.netherlight.net-505-1 73
tdm3.amsterdam1.netherlight.net-504-3 4 tdm3.amsterdam1.netherlight.net-505-1 76
tdm3.amsterdam1.netherlight.net-504-3 7 tdm3.amsterdam1.netherlight.net-505-1 79
-----
tdm1.amsterdam1.netherlight.net-6-1 73 tdm1.amsterdam1.netherlight.net-17-1 25
tdm1.amsterdam1.netherlight.net-6-1 76 tdm1.amsterdam1.netherlight.net-17-1 28
tdm1.amsterdam1.netherlight.net-6-1 79 tdm1.amsterdam1.netherlight.net-17-1 31
```

Future: Provisioning tool

- Next step is to generate the TL1 commands
  - All the necessary cross connect information is available

Syntax:

```
ENT-CRS-STS3C:"NE-name":STS3C-shelf-slotA-portA-timeslotA,STS3C-shelf-slotB-portB-
    timeslotB:CTAG-NUMBER::2WAY:CKTID="name of Xconnect";
```

Example:

```
ENT-CRS-STS3C:"tdm4.amsterdam1.netherlight.net":STS3C-1-6-1-22,STS3C-1-1-3-
    1:66::2WAY:CKTID="DEMO";
```

- SARA already has developed a TL1 toolkit (Perl)*
  - * Easy to execute TL1 commands with TL1 toolkit
  - * Specific functions for: OME6500, CPL, HDXc, ONS15454
  - * Planning to add a `create_Xconnect()` function

* Based on TL1 Perl module of Arien Vijn, Amsterdam Internet Exchange

- SARA's network research group:
  Information about TL1 toolkit & RDF tools.
  http://nrg.sara.nl/
  Email: nrg@sara.nl
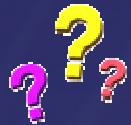
- Network Description Language:
  http://staff.science.uva.nl/~vdham/research/ndl/

That's all Folks!