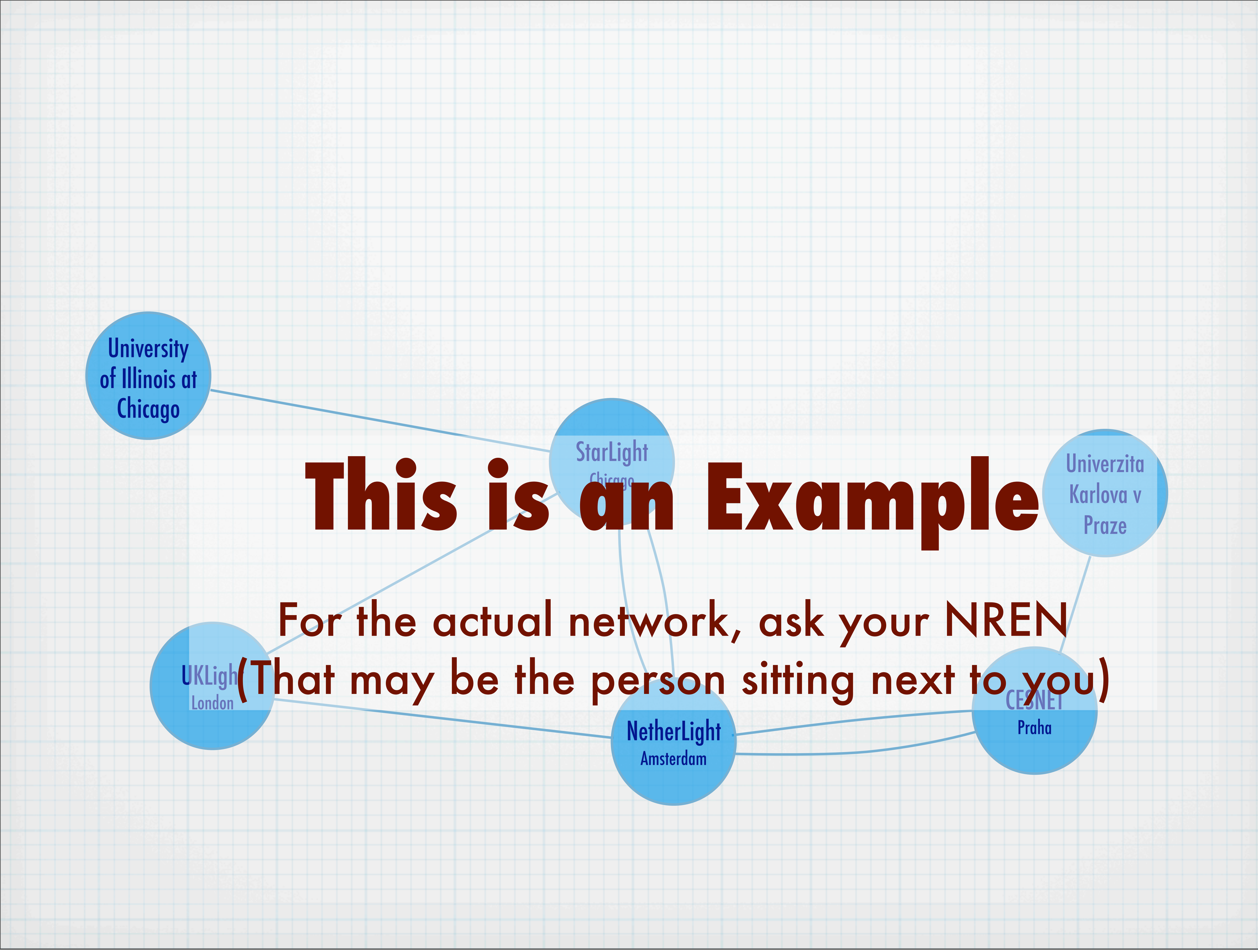
Going in Loops to Reach your Goal

Freek Dijkstra
Universiteit van Amsterdam
with help of:
Jeroen van der Ham,
Paola Grosso,
Bert Andree, Karst Koymans, Cees de Laat
Fernando Kuipers (TU Delft)

This work is funded by the Gigaport Project

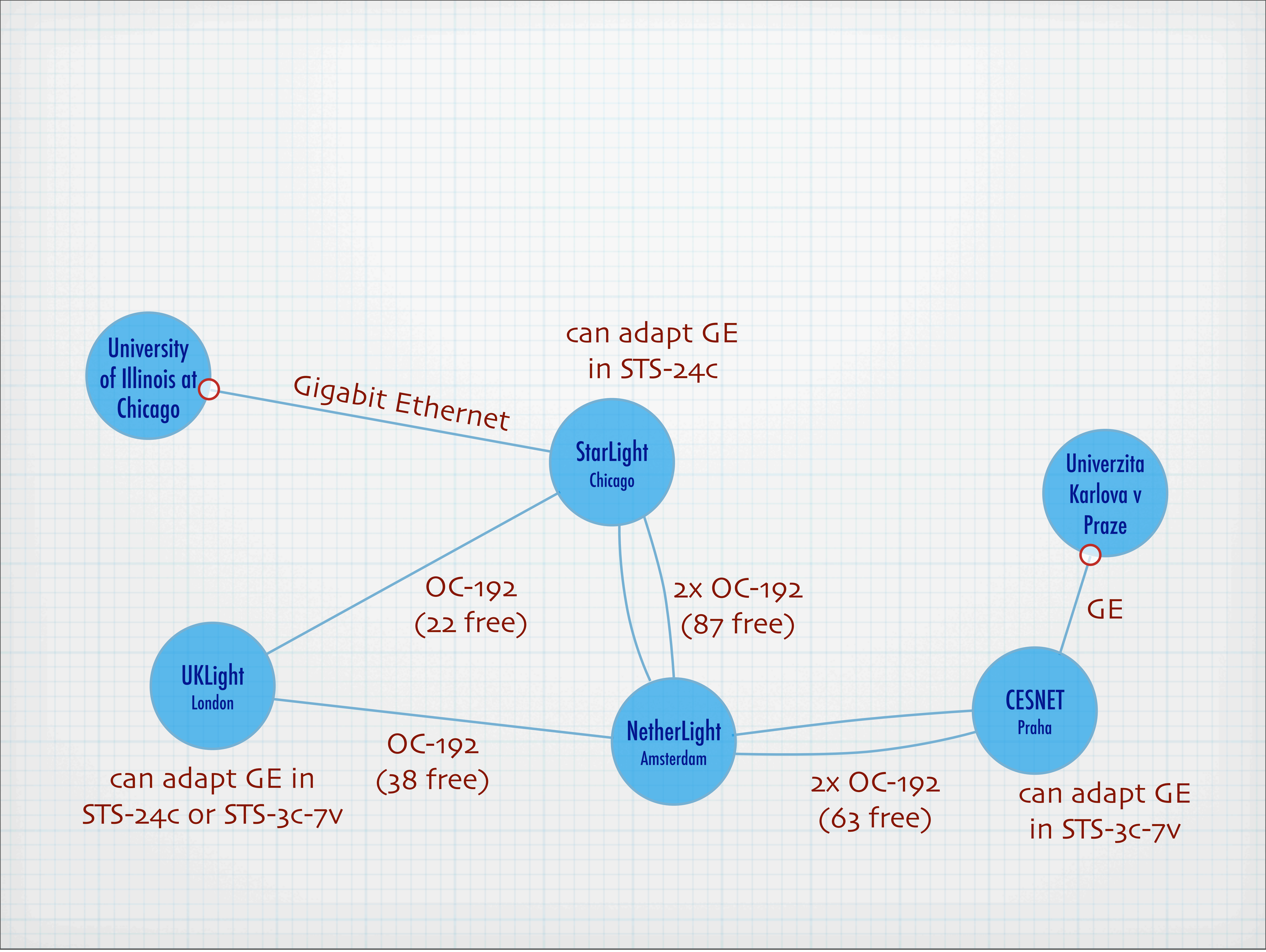
dinsdag 18 september 2007

Multi-layer path finding is a problem. I will not give an answer to that, but I will show that there is a problem, and give you both an information and a data model to use.



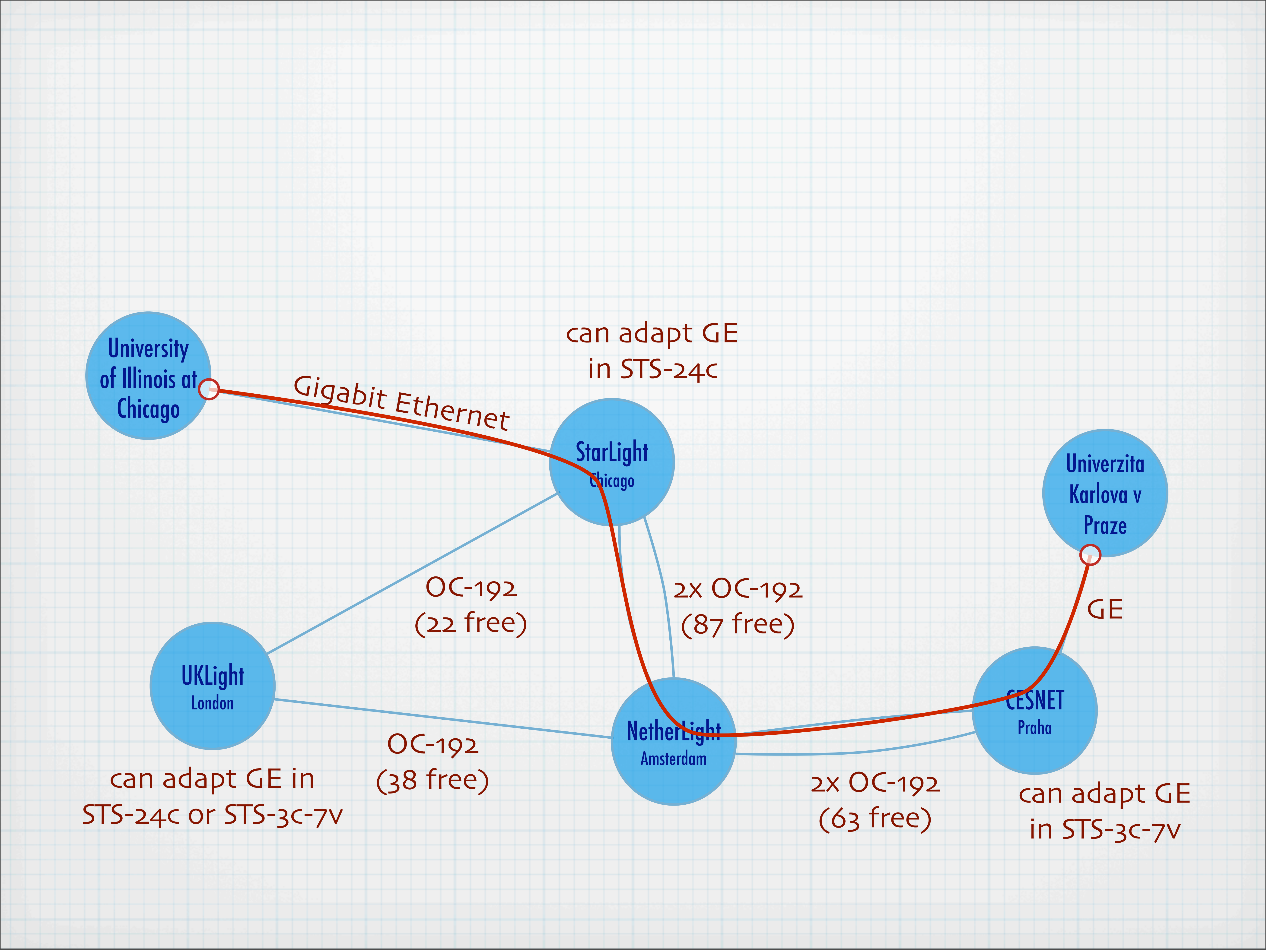
dinsdag 18 september 2007

This is our network. It's an example based on a practical problem in the GLIF about 2004. Hybrid: different technologies/layers. Different adaptations between layers. Not all available. Question for the network engineers in the audience: please find the shortest working path from University of Illinois at Chicago to Charles University in Prague.



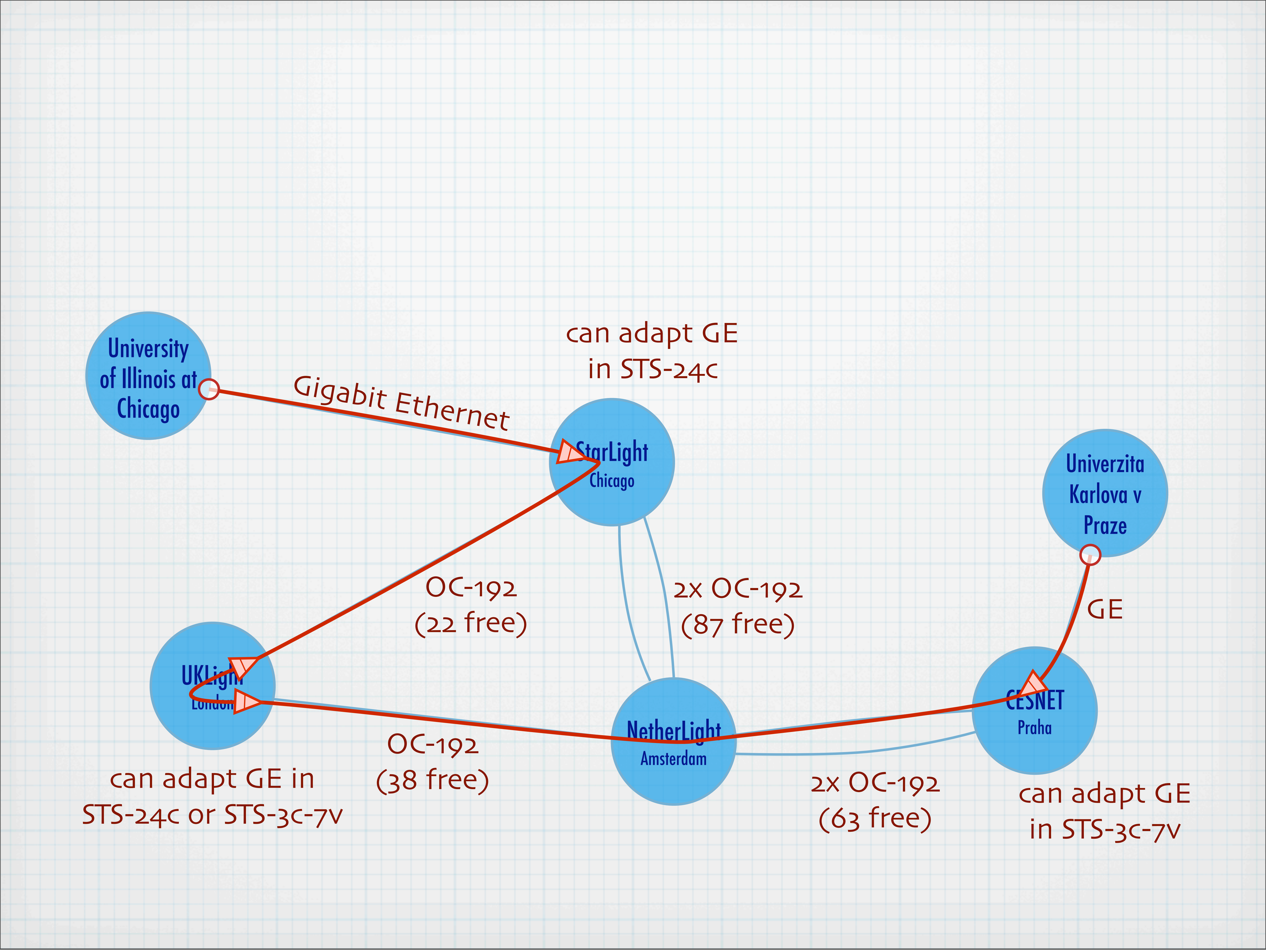
dinsdag 18 september 2007

This is our network. It's an example based on a practical problem in the GLIF about 2004.
Hybrid: different technologies/layers. Different adaptations between layers. Not all available.
Question for the network engineers in the audience: please find the shortest working path from University of Illinois at Chicago to Charles University in Prague.



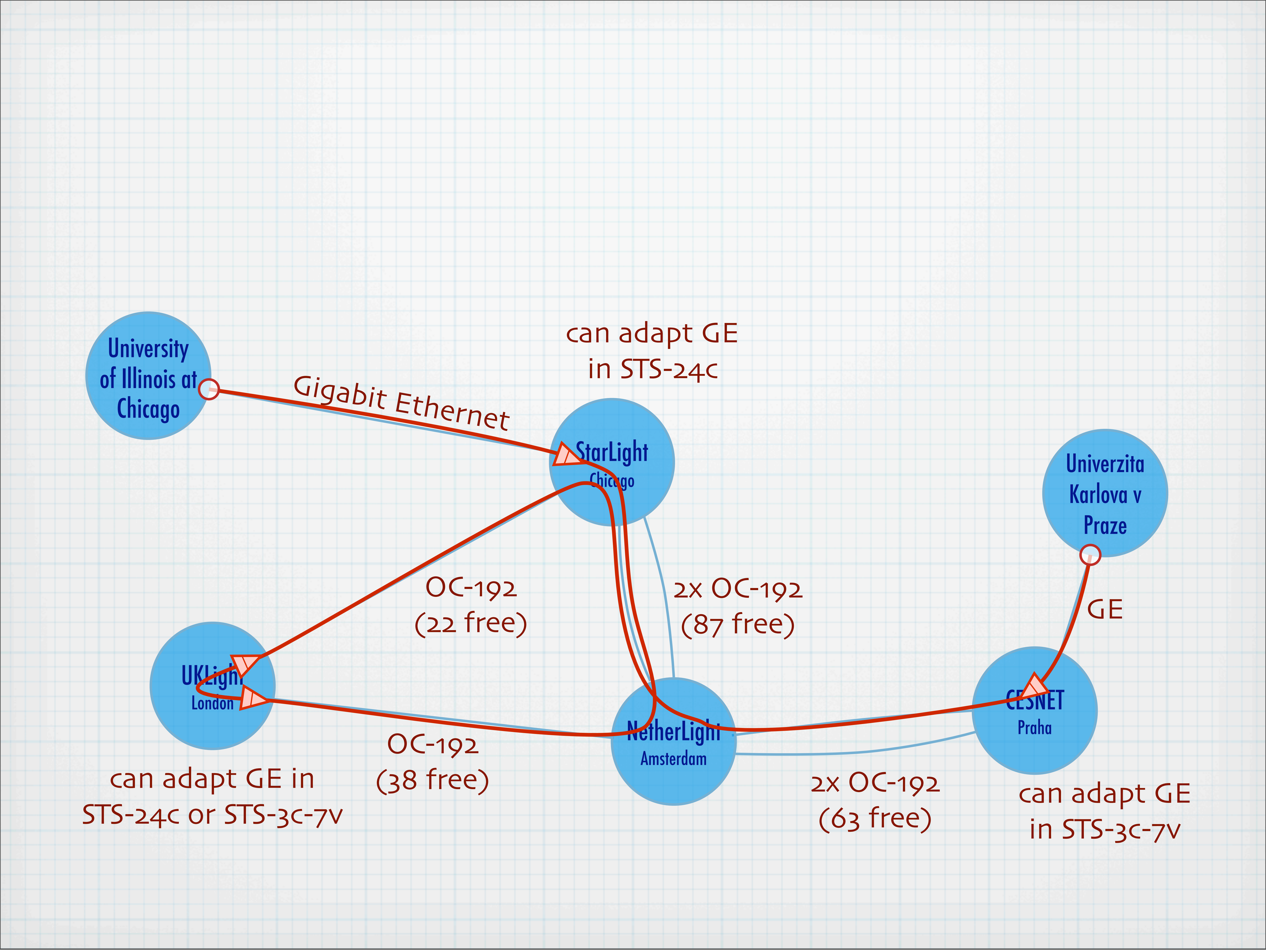
dinsdag 18 september 2007

First attempt: shortest path if you consider this as a graph.
Link constraints don't work: every path is available.



dinsdag 18 september 2007

We need to take adaptation into account: StarLight and CESNET us different adaptations of GE in STS channels.



dinsdag 18 september 2007

This is the shortest path through this network. You can not just consider one layer in this example: Quebec and Amsterdam do not even know about SDH. NetherLight does not understand Ethernet. Adaptations are important.
We need a new “Dijkstra”.

Create a **computer-readable network description**, that provides enough **information for path finding** in **multi-layer networks**

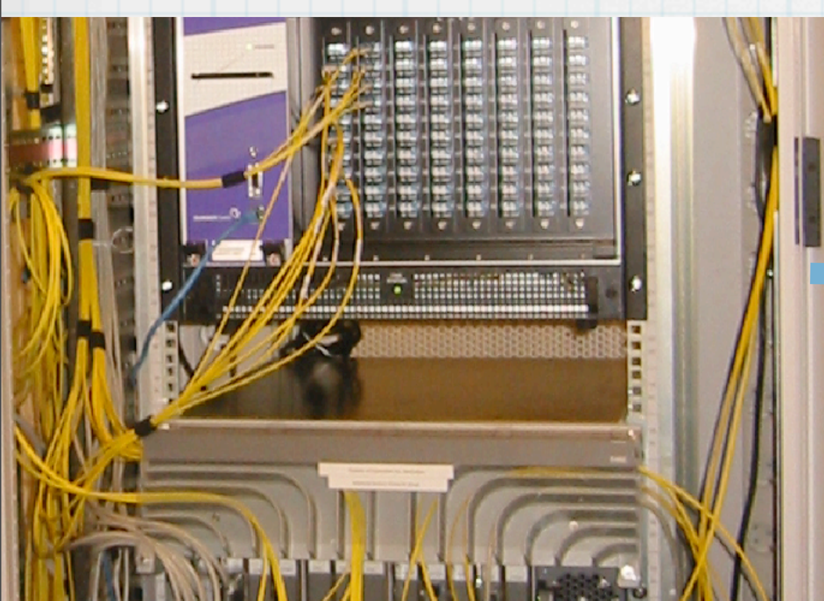
Goal

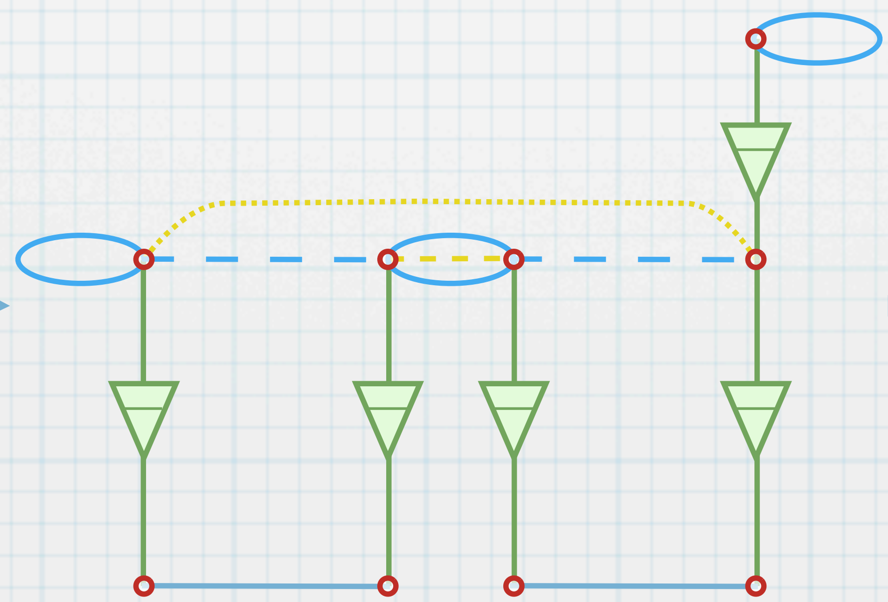
dinsdag 18 september 2007

3 tasks ahead: make model, make syntax, make software.

The Modelling Process

Network Elements → Functional Elements → Syntax

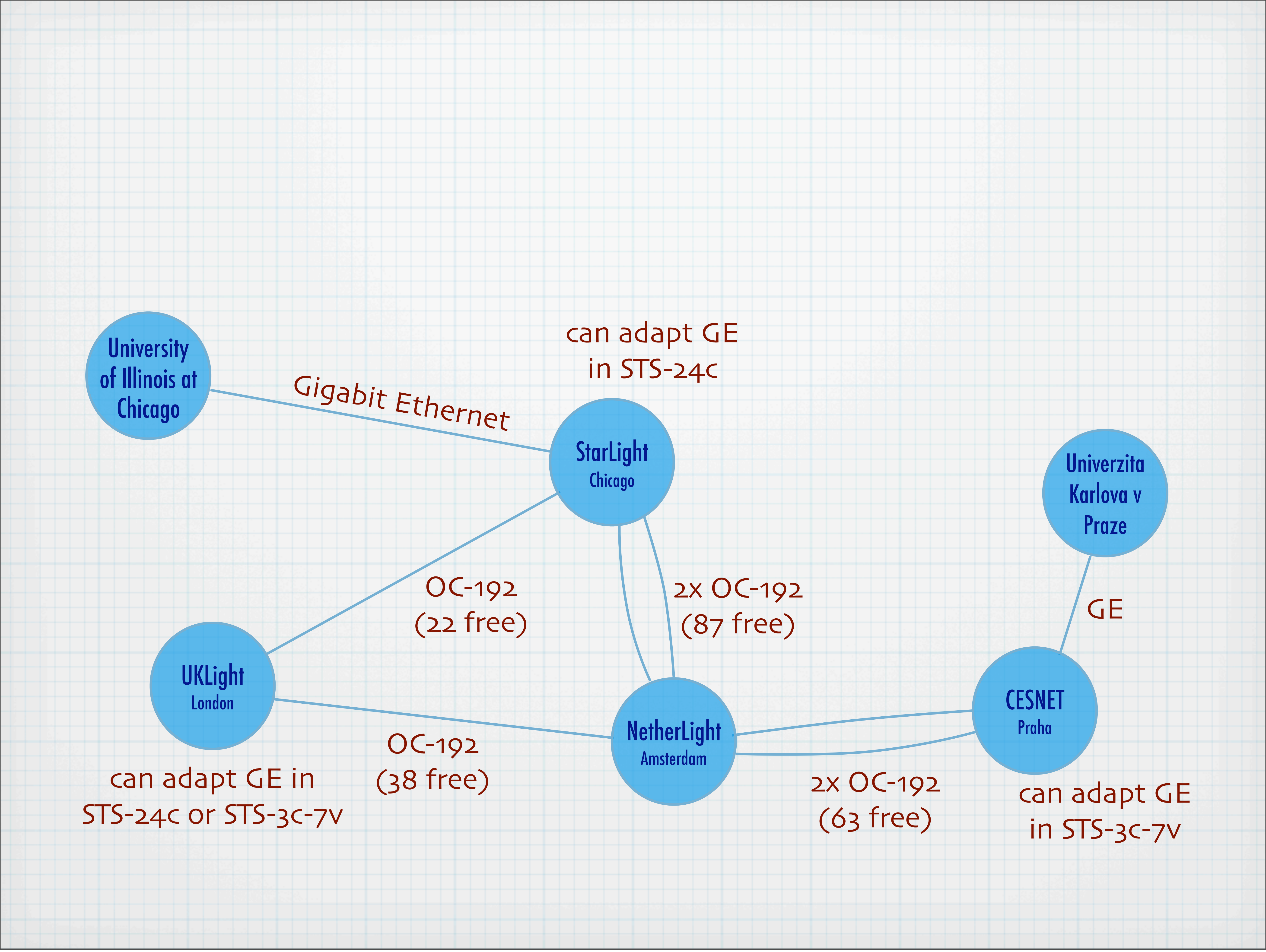





```xml
<ndl:Device rdf:about="#Force10">
   <ndl:hasInterface rdf:resource=
       "#Force10:te6/0"/>
</ndl:Device>
<ndl:Interface rdf:about="#Force10:te6/0">
   <rdfs:label>te6/0</rdfs:label>
   <ndl:capacity>1.25E6</ndl:capacity>
   <ndlconf:multiplex>
      <ndlcap:adaptation rdf:resource=
          "#Tagged-Ethernet-in-Ethernet"/>
      <ndlconf:serverPropertyValue
           rdf:resource="#MTU-1500byte"/>
   </ndlconf:multiplex>
   <ndlconf:hasChannel>
      <ndlconf:Channel rdf:about=
          "#Force10:te6/0:vlan4">
         <ndleth:hasVlan>4</ndleth:hasVlan>
         <ndlconf:switchedTo rdf:resource=
             "#Force10:gi5/1:vlan7"/>
      </ndlconf:Channel>
   </ndlconf:hasChannel>
</ndl:Interface>
```

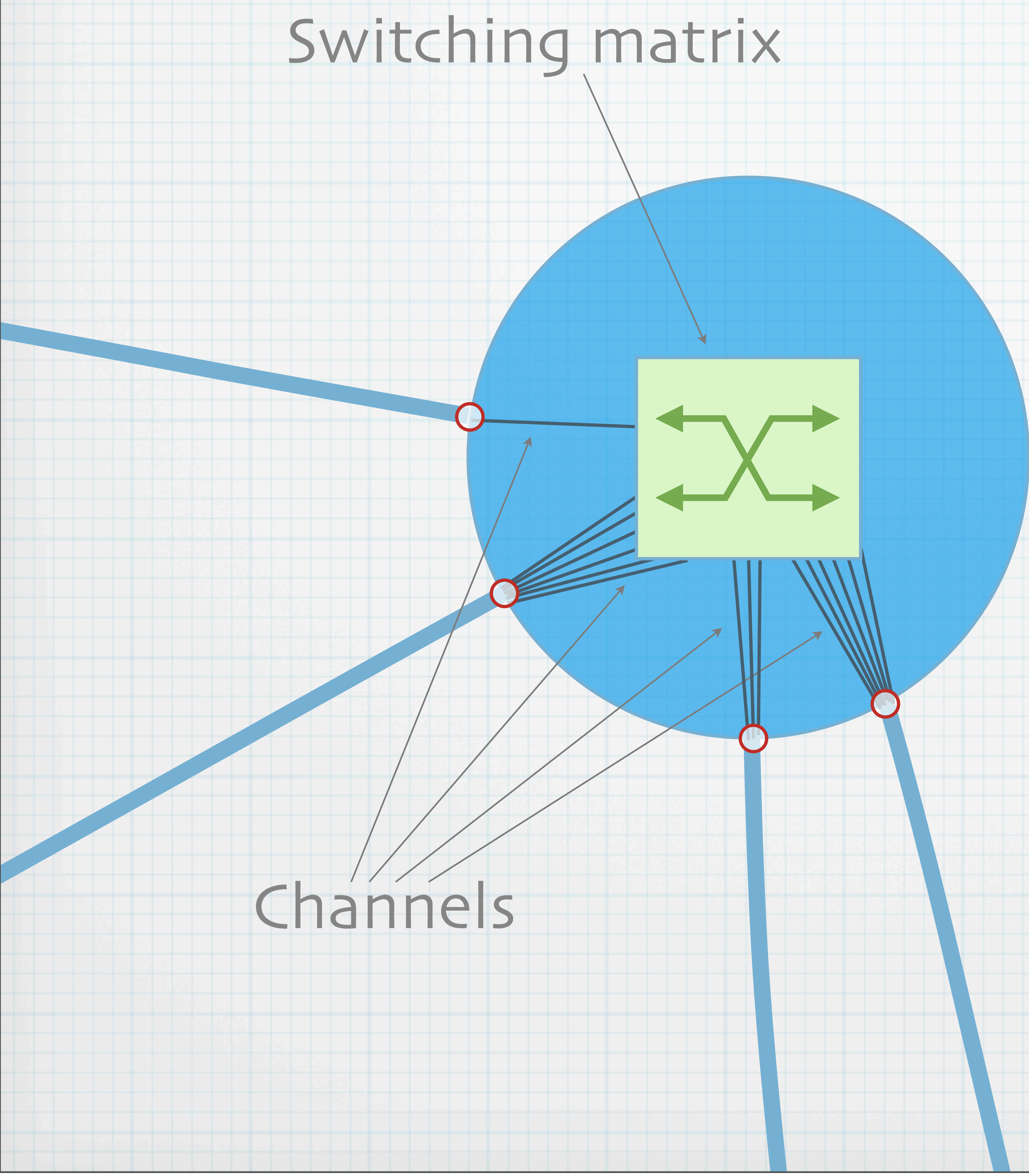

dinsdag 18 september 2007

Let’s remind about the theory of modelling.
- Network elements: physical device
- Map to functional elements (e.g. G.805 elements)
- Rewrite that in a concise, but compact syntax (e.g. NDL)



dinsdag 18 september 2007

Task 1: Model (GMPLS + G.805): next few slides



A devices switches data based on:

- The source interface
- One or more labels

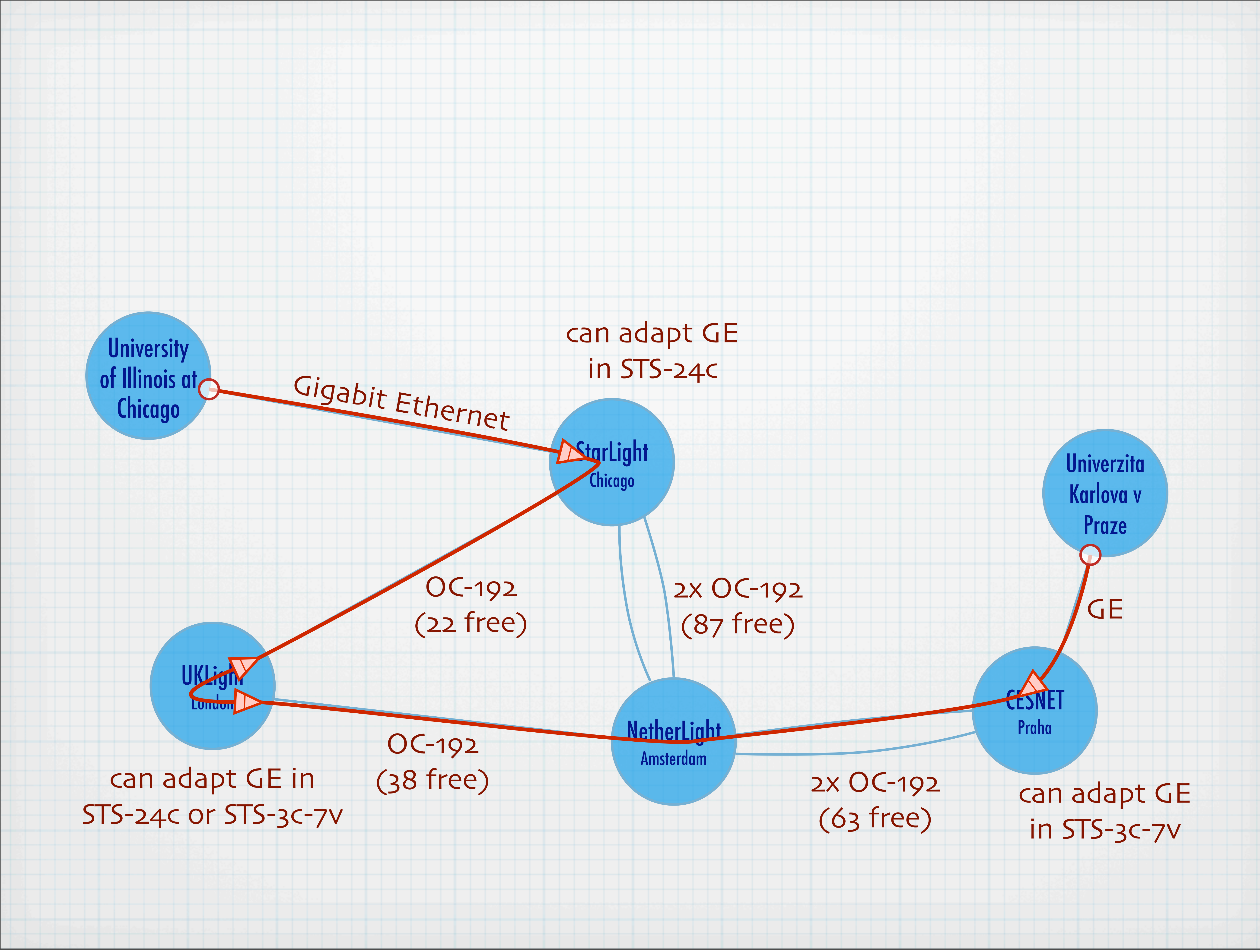
Example label types:

- Ethernet VLAN
- SONET STS Channel
- Wavelength (λ)

For example, all data from channel 31 of interface 2 is forwarded to channel 28 of interface 4.

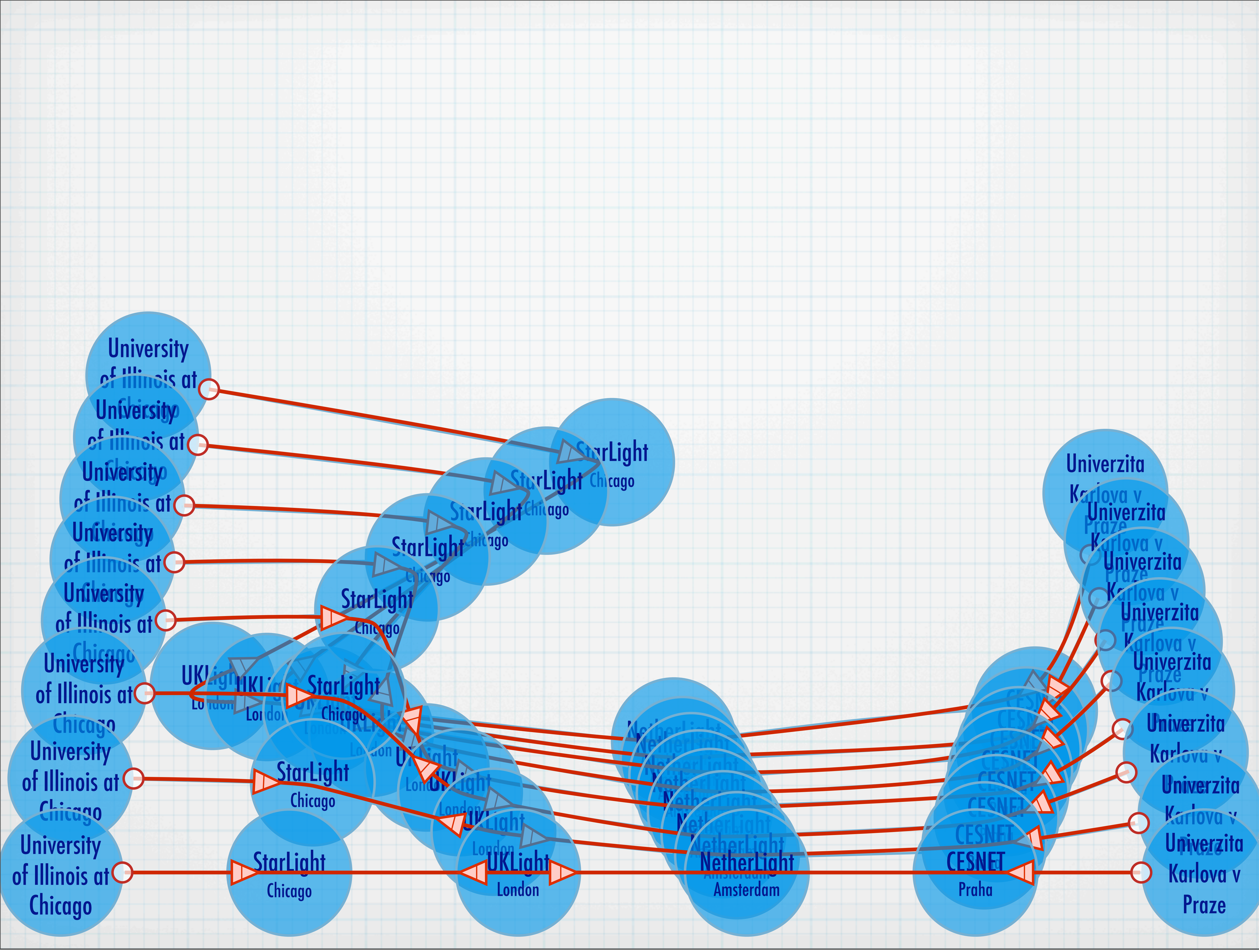
dinsdag 18 september 2007

Core of a device is a switching matrix. Typically, every connected link is split (demuxed) into multiple channels, each of which is connected to the switching matrix.
Any property that is used to make a switching decision is a label type.
GMPLS concept.

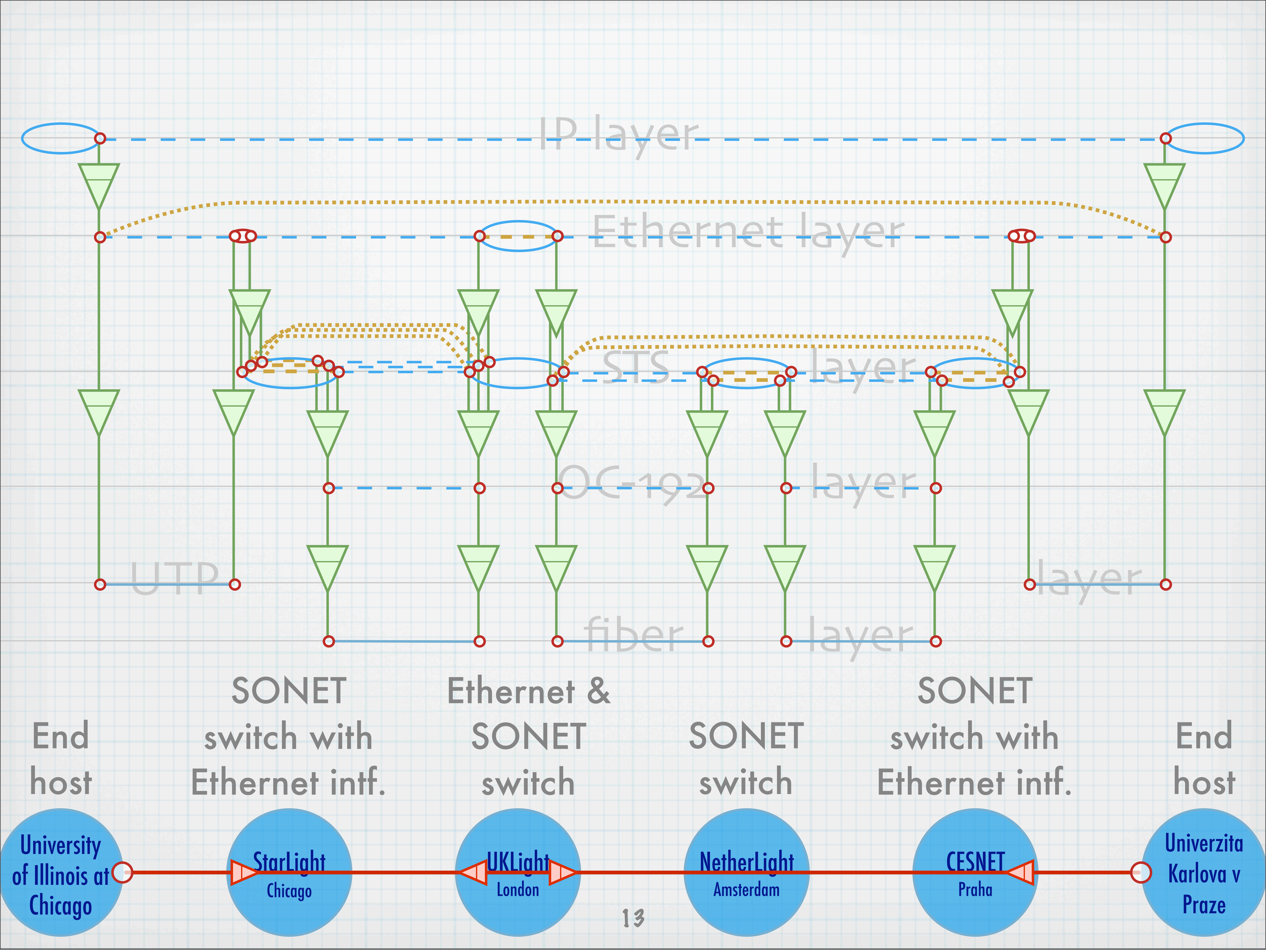


dinsdag 18 september 2007

G.805. Let’s go back to our second attempt and examine the adaptation incompatibilities.

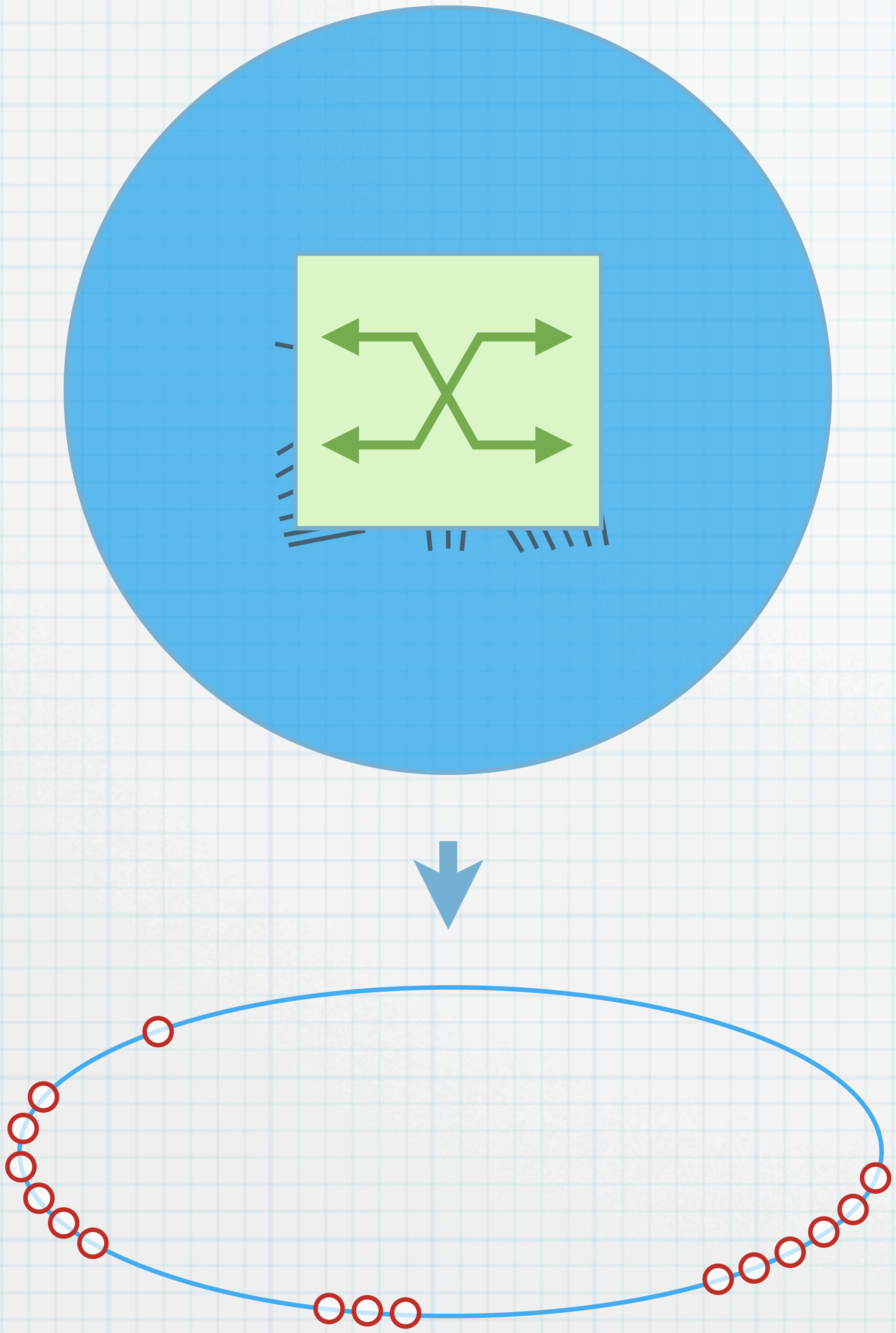


dinsdag 18 september 2007



dinsdag 18 september 2007

We use G.805 functional elements for our information model.
subnetwork, connection points (few per interface), adaptation (+termination) functions, links, link connections, subnetwork connections (configuration), network connections.
In addition, we use the label concept of GMPLS (not explained in presentation).



Subnetwork with 433 connection points

Device

switchingCapability → LabelType

Can switch, but not change label. E.g. from STS 31 of interface 2 to STS 31 of interface 4.

swappingCapability → LabelType

Can change label. E.g. from STS 31 of interface 2 to STS 28 of interface 4.

Interface

hasLabel → Label

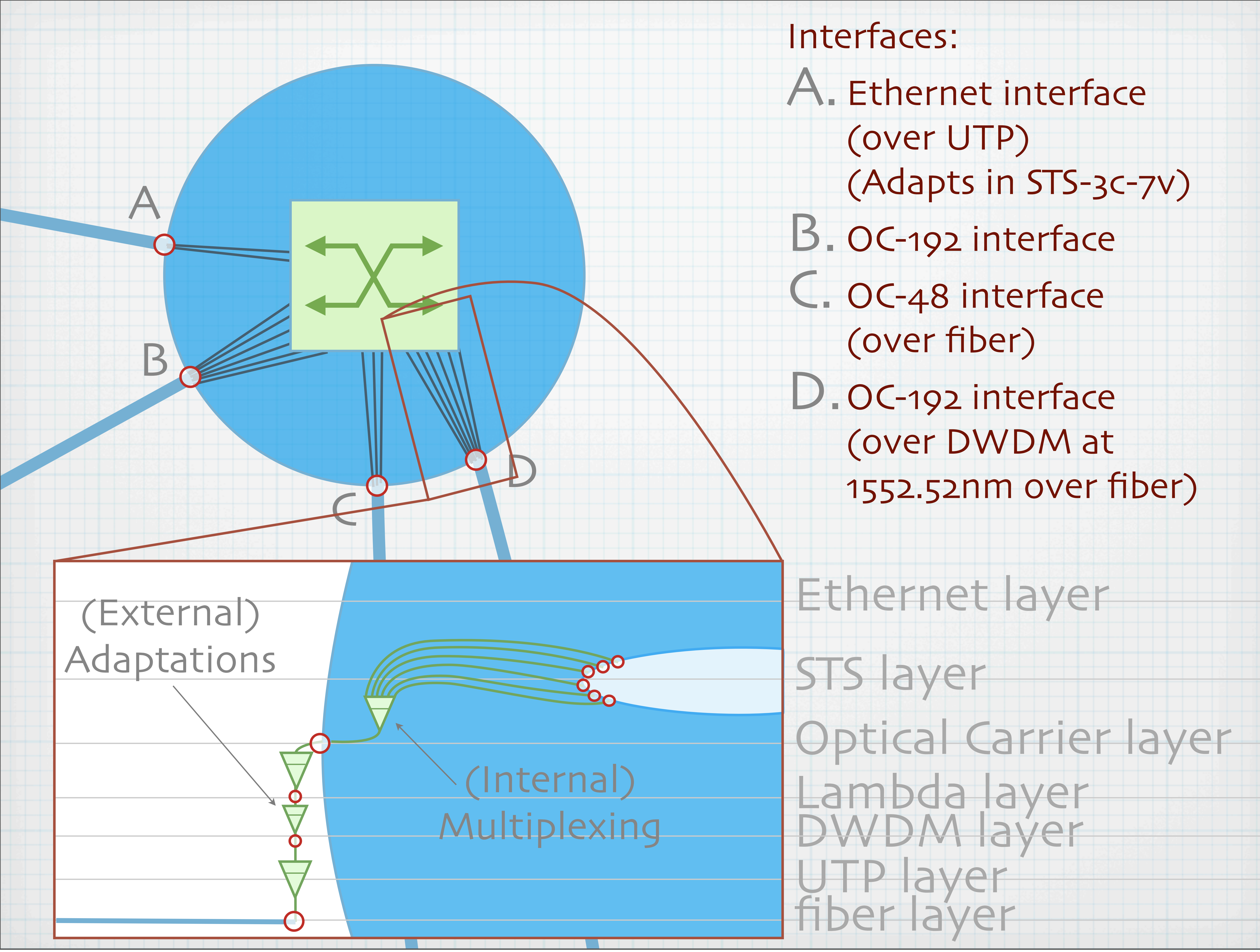
Channel Identifier

switchedTo → Interface

A subnetwork connection

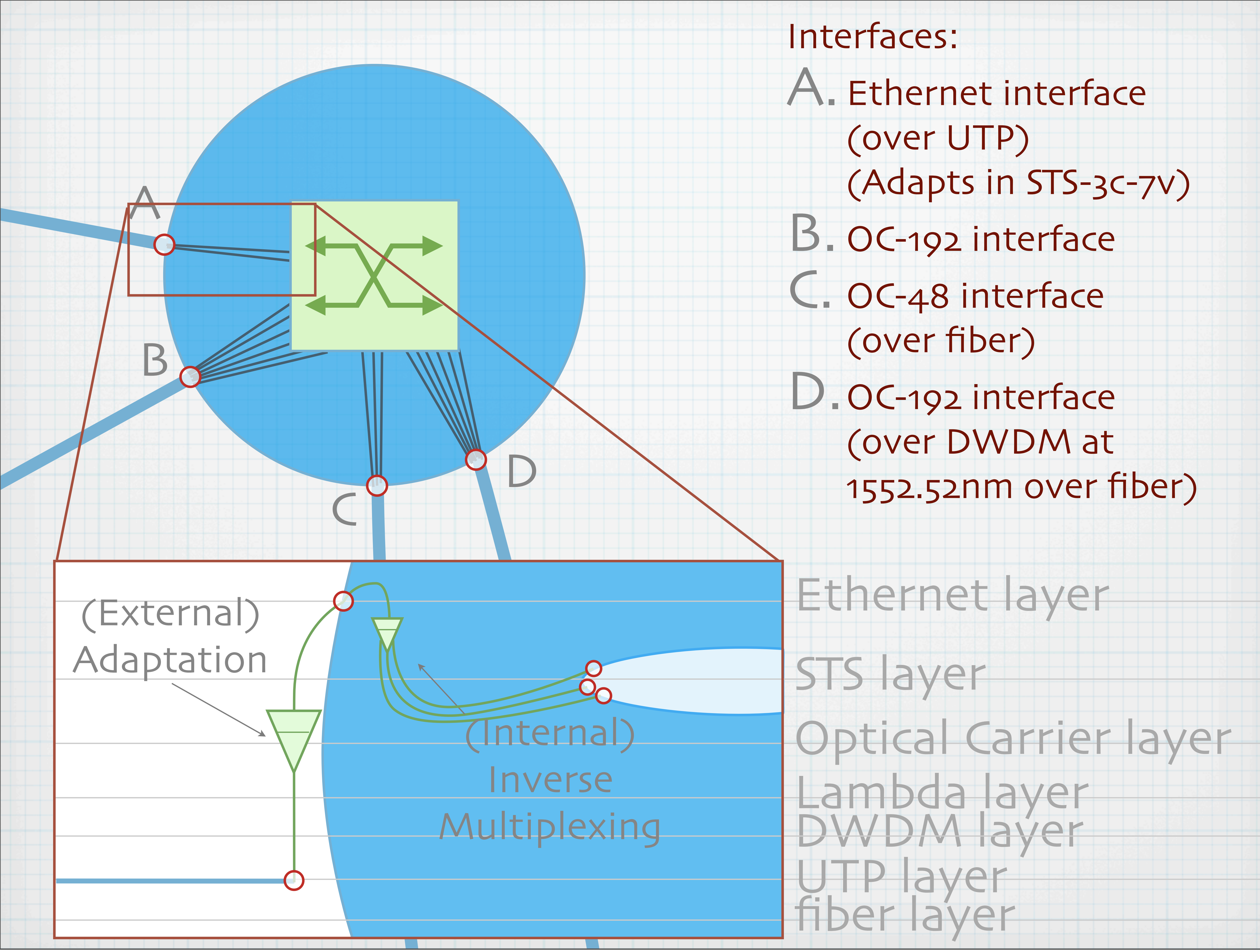
dinsdag 18 september 2007

mapping device -> subnetwork
Explain switching and swapping!



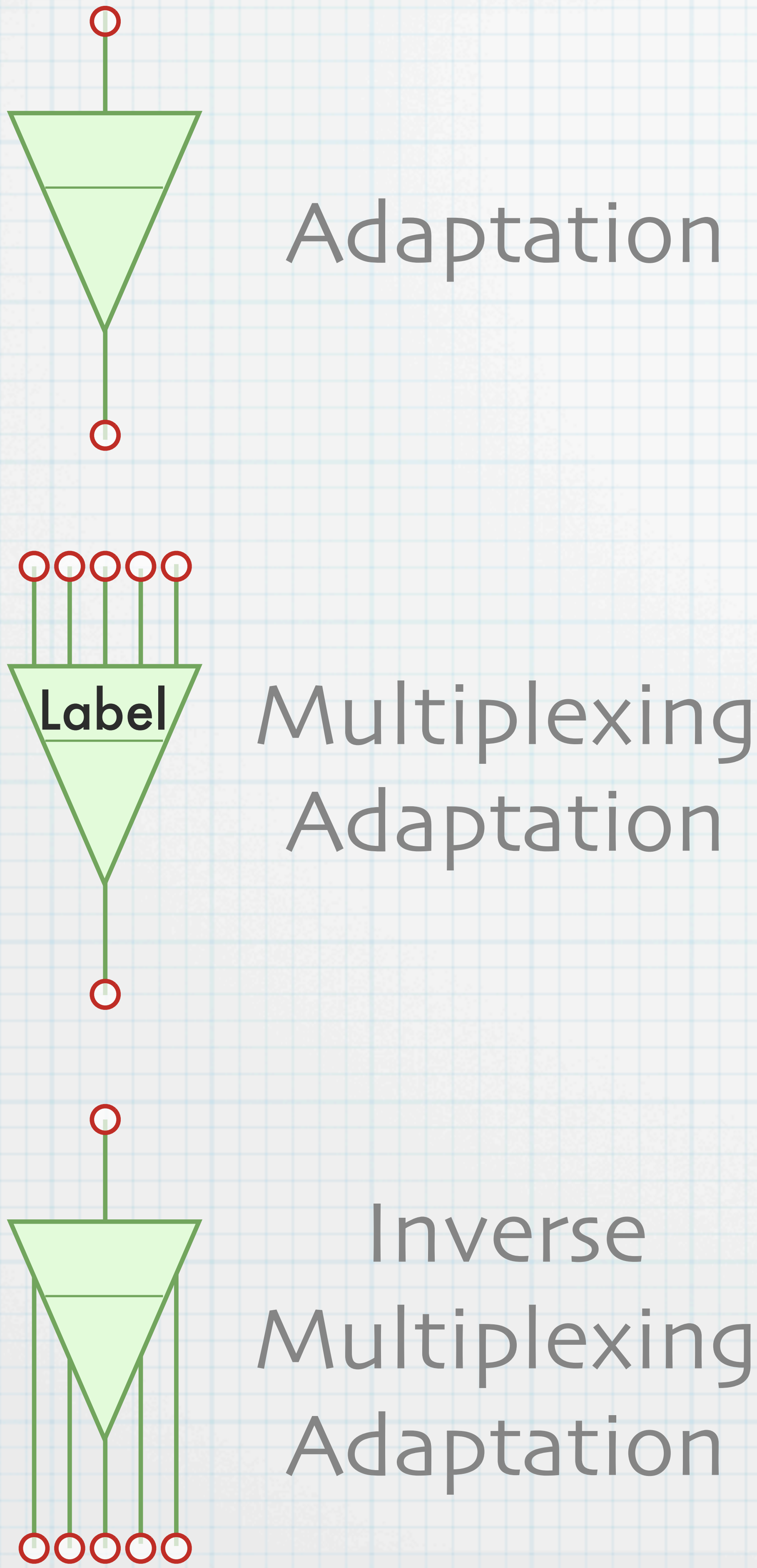
dinsdag 18 september 2007

Examples of mapping interface -> functional elements (connection points and adaptation functions)



dinsdag 18 september 2007

Examples of mapping interface -> functional elements (connection points and adaptation functions)



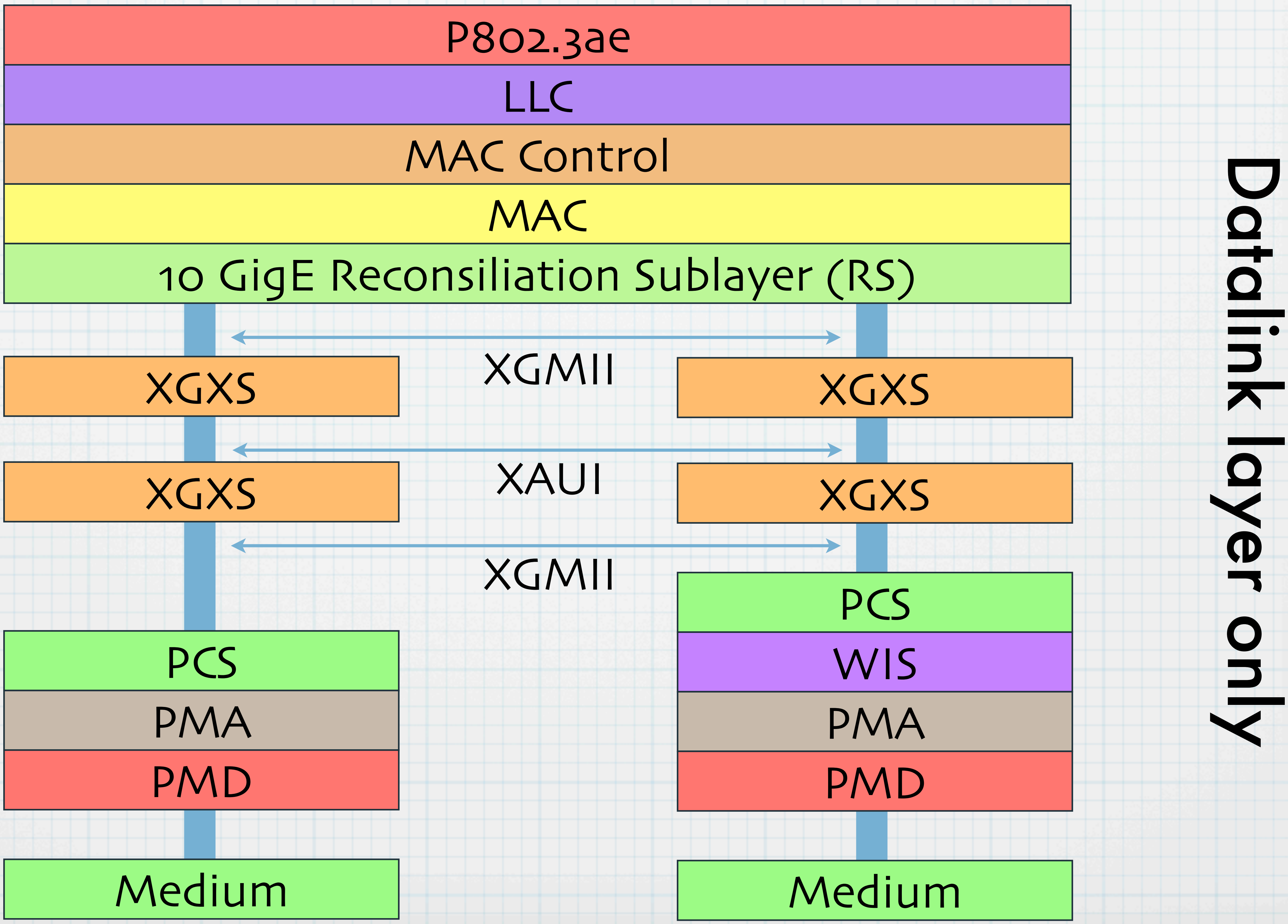
Adaptation

rdf:ID	→ URI
clientLayer	→ Layer
serverLayer	→ Layer
labels	→ LabelSet
allowed/available labels	
clientLayerCount	→ integer
>1 for multiplexing.	
serverLayerCount	→ integer
1 by default. >1 for inverse multiplexing.	
clientCapacity	→ float
provided max. capacity in Bytes/s to the client layer.	
serverCapacity	→ float
required min. capacity in Bytes/s per channel from the client layer.	
serverPropertyValues	→ PropertyValues

dinsdag 18 september 2007

Example of adaptation: Ethernet over UTP, or Ethernet over Fiber.
Example of multiplexing: different data streams, each in a separate wavelength.
Example of inverse multiplexing: Ethernet in multiple STS timeslots.

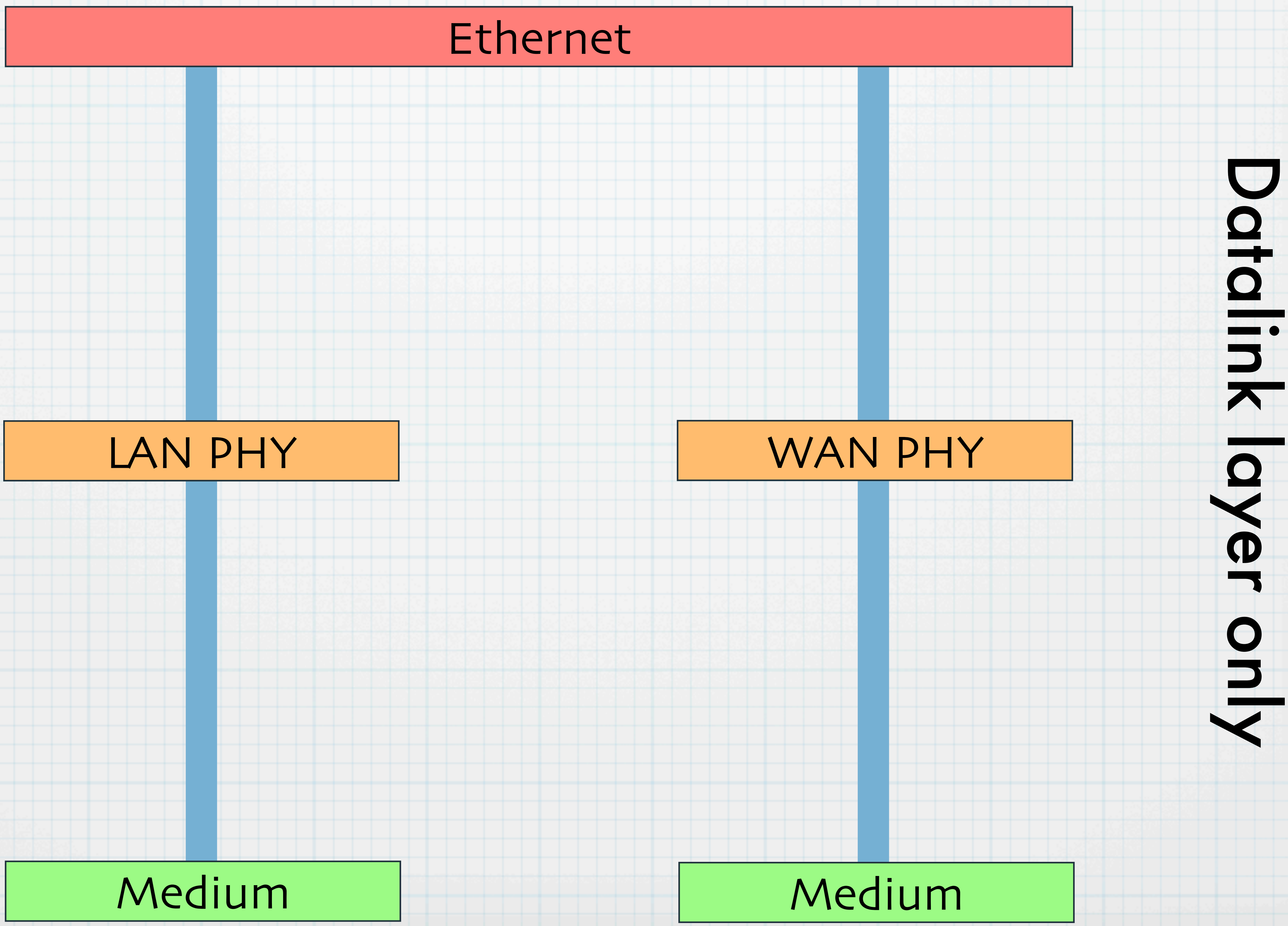
Techology Specific Schemas



dinsdag 18 september 2007

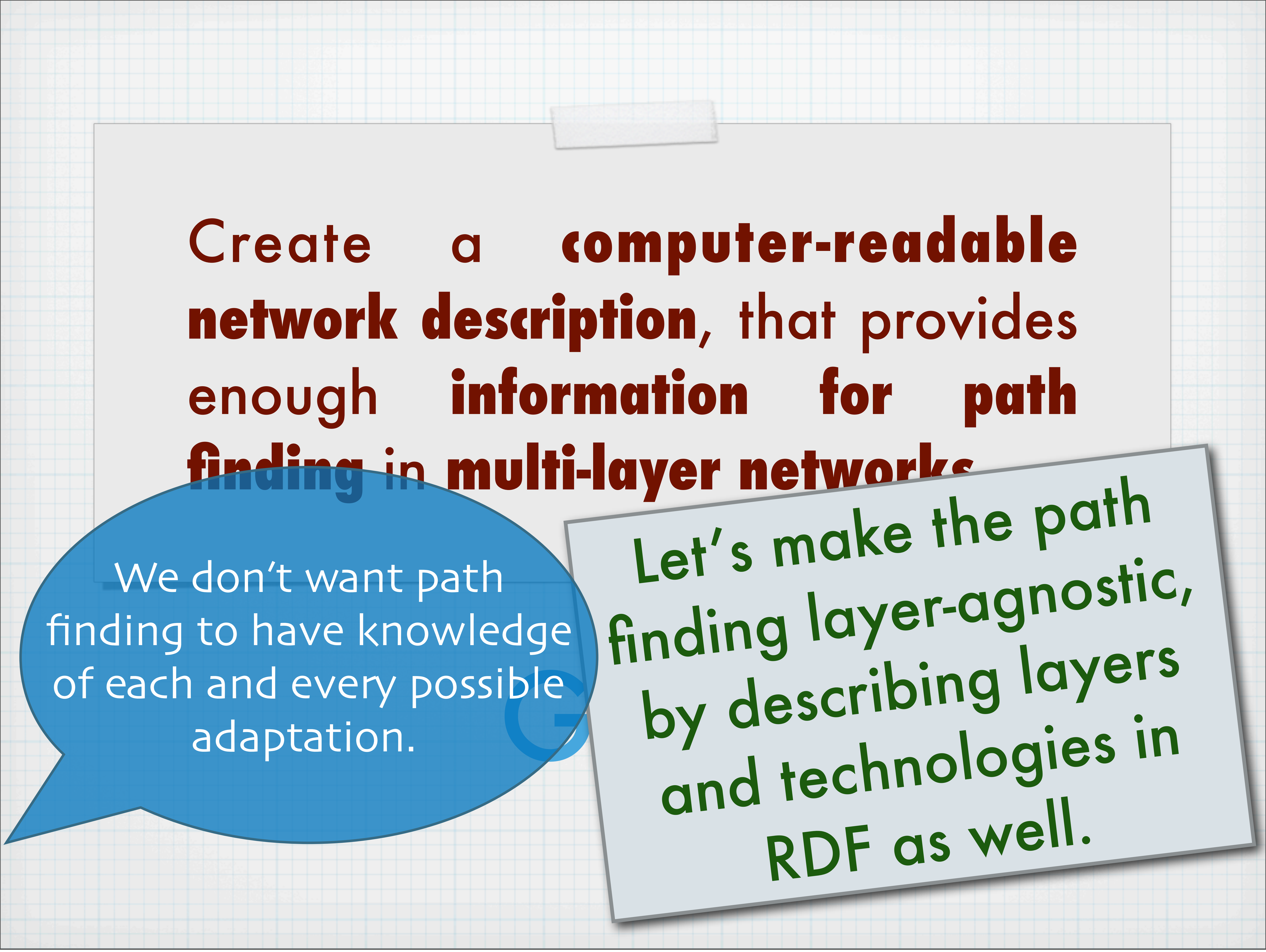
Reason to separate technology schemas from other schemas: we want to allow changes to the schema.
It is not clear how much simplification we allow in the schema. Here is an example of a model with lots of details. Left: LAN PHY, Right: WAN PHY Ethernet

Techology Specific Schemas



dinsdag 18 september 2007

More simple description of same layers.
Reason to for me describe networks: describe incompatibilities for path finding, so I don’t need more info. Perhaps someone else may.



dinsdag 18 september 2007

So we completed task 1, the modelling. Next up: step 2: syntax.

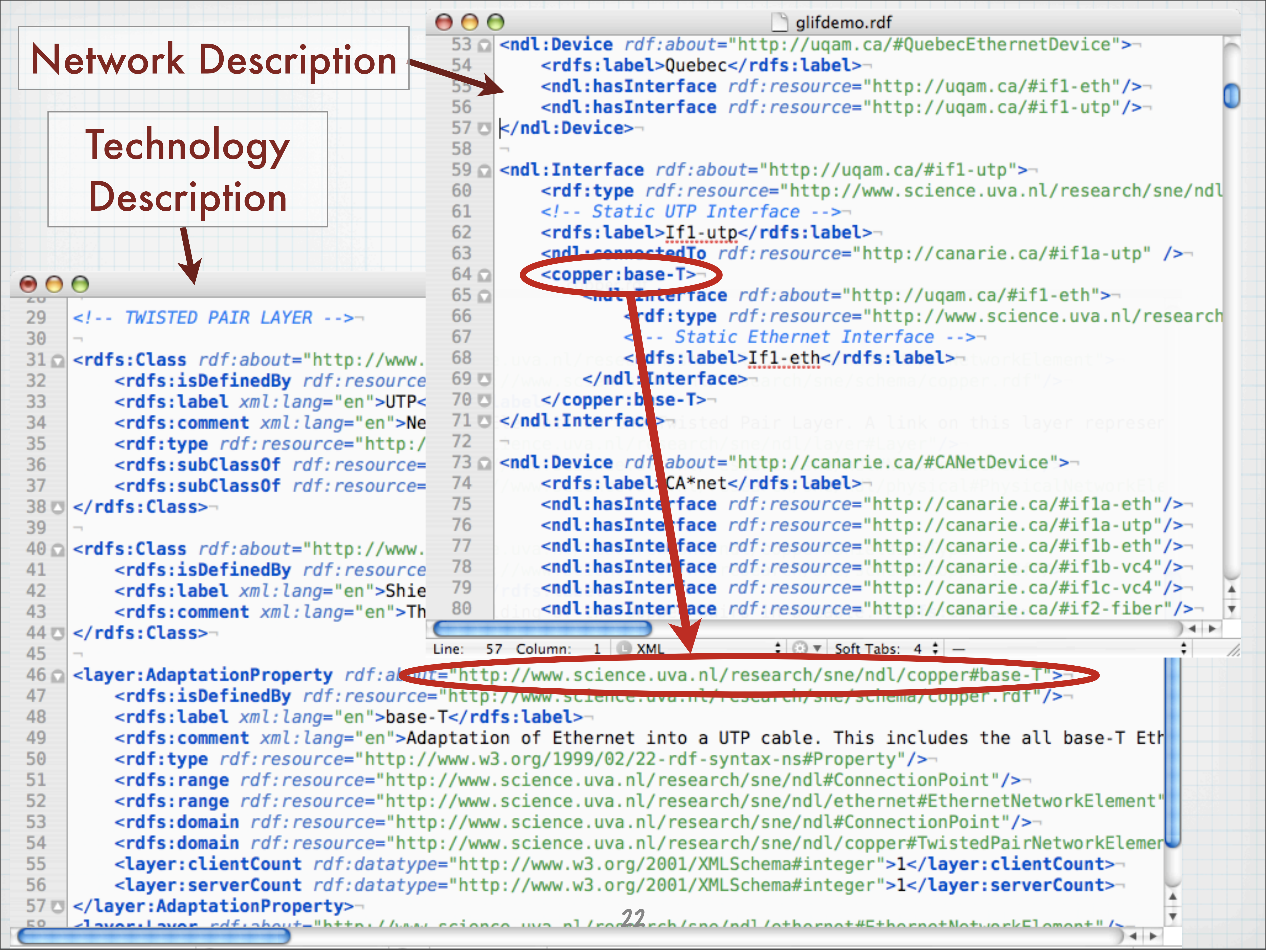
Different Subtopics

- Topology
  (existing NDL schema)
- Layer specification
  Definition of different Layers
  Layer, LabelType, Adaptation
- Device capabilities
  Configurable Interfaces,
  switching & swapping capability.
- Device configuration
  Internal connections, available
  labels (e.g. free VC-4 channels)
- Domain abstraction ...
- Path Requests ...

- IP schema
- Ethernet schema
- ATM schema
- SONET/SDH schema
- WDM schema
- Physical layer schema
- Fiber bundle schema

dinsdag 18 september 2007

Each subtopic got it's own schema. We have 4 basic schemas (not mentioned: physical properties, re-use CIM).
In addition, we have 6 layer-specific schema.
Capability: needed for path finding; Configuration: needed for fault isolation.



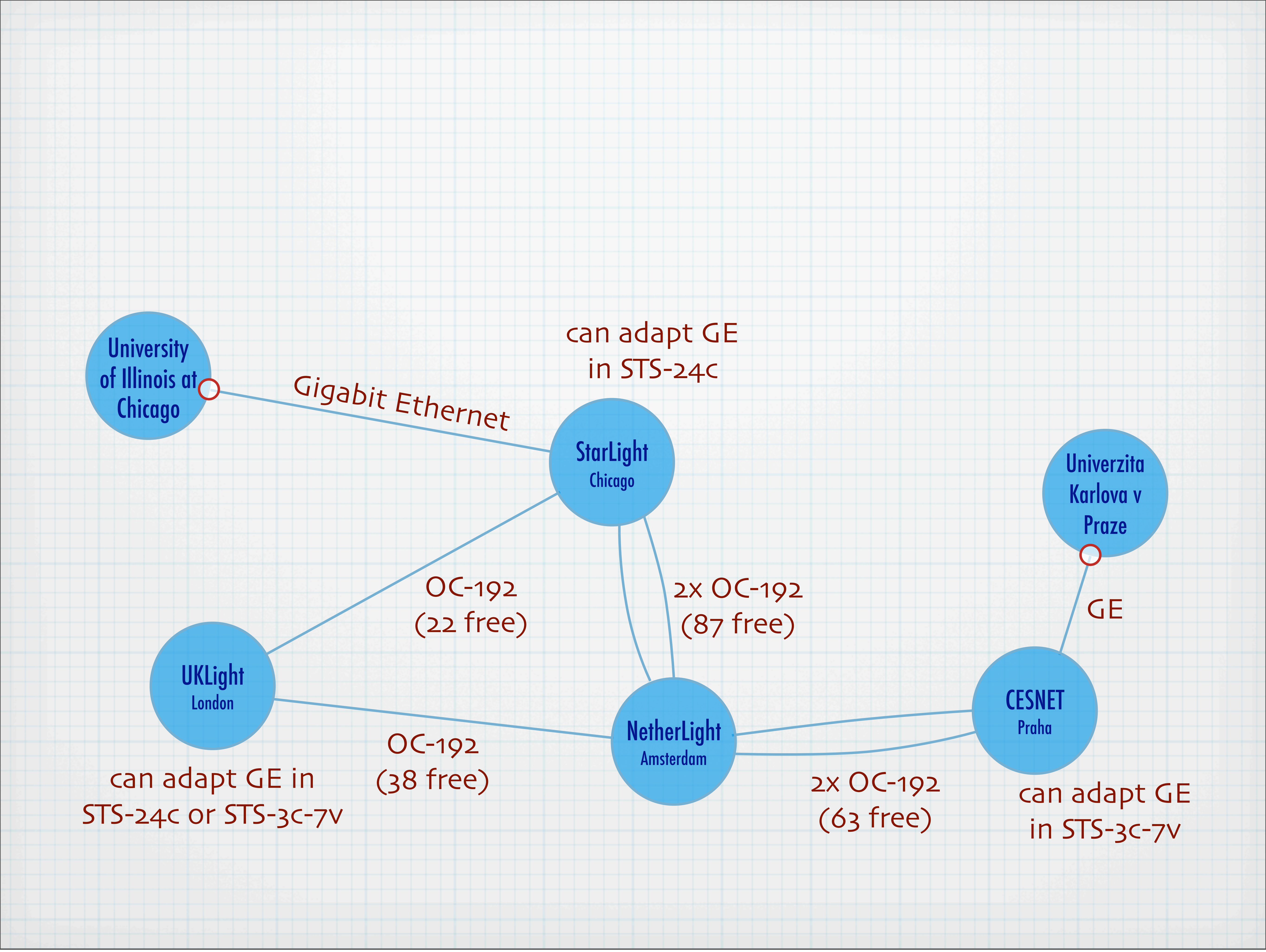
dinsdag 18 september 2007

Every domain publishes its own data. RDF allows domains to link to each other. Publication can be (semi-)static, or using webservice.

Python NDL Toolkit

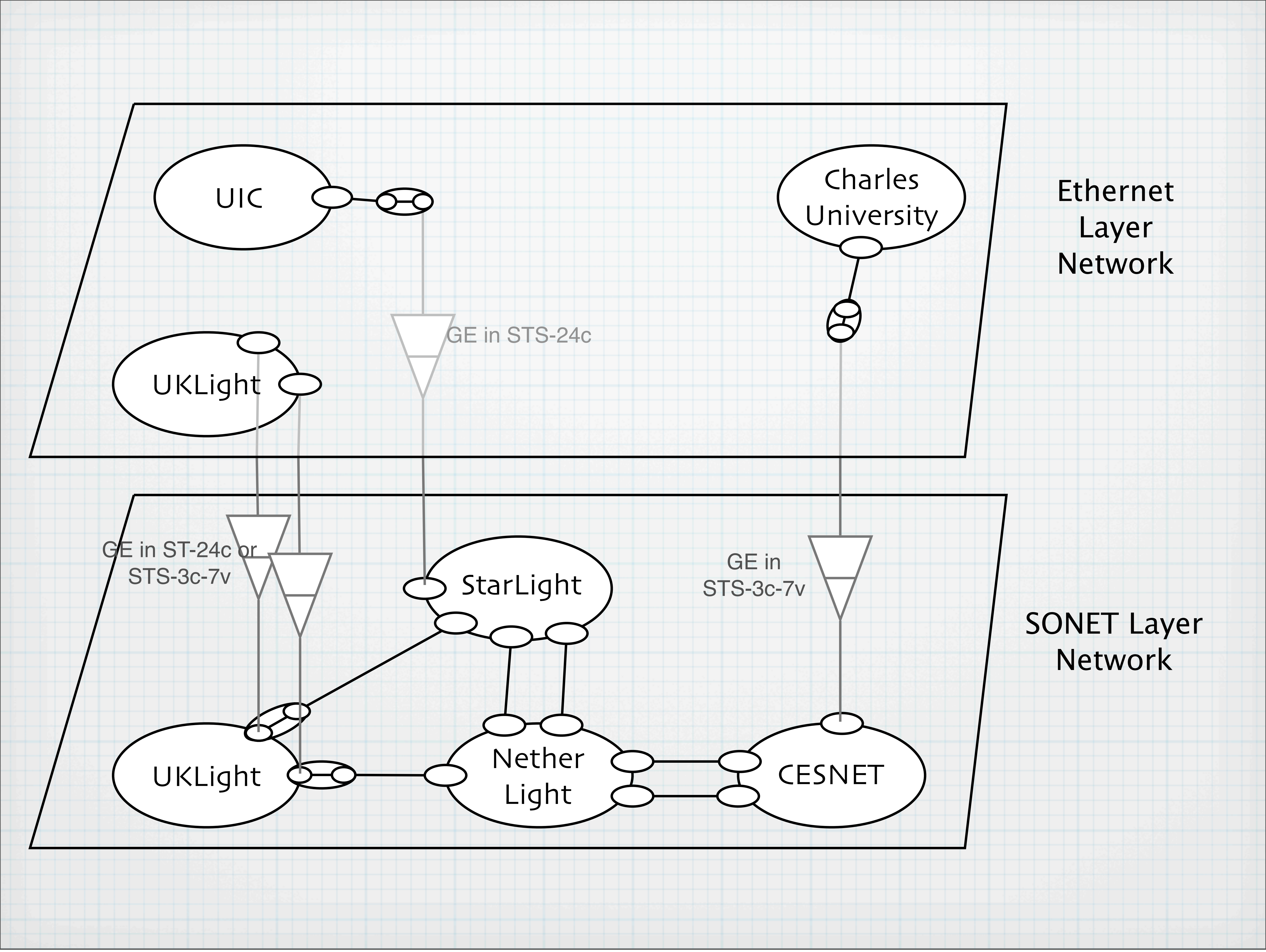
- Input of network and technology descriptions in NDL/RDF
  - Partial RDF schema parser
- Input of network descriptions from devices (TL1, CLI or OSPF)
  - TL1, CLI, OSPF parsers
    (Similar to SARA Perl TL1 toolkit, only in Python)
- Output to NDL/RDF
- Output to DOT (Graphviz)
- Path Walk (existing connections)
- Path Find (available connections)

Now we can make network descriptions. (Step 2 is done).
Next up, step 3: Make software to do something useful with it.
Available for download. (BSD-license)



dinsdag 18 september 2007

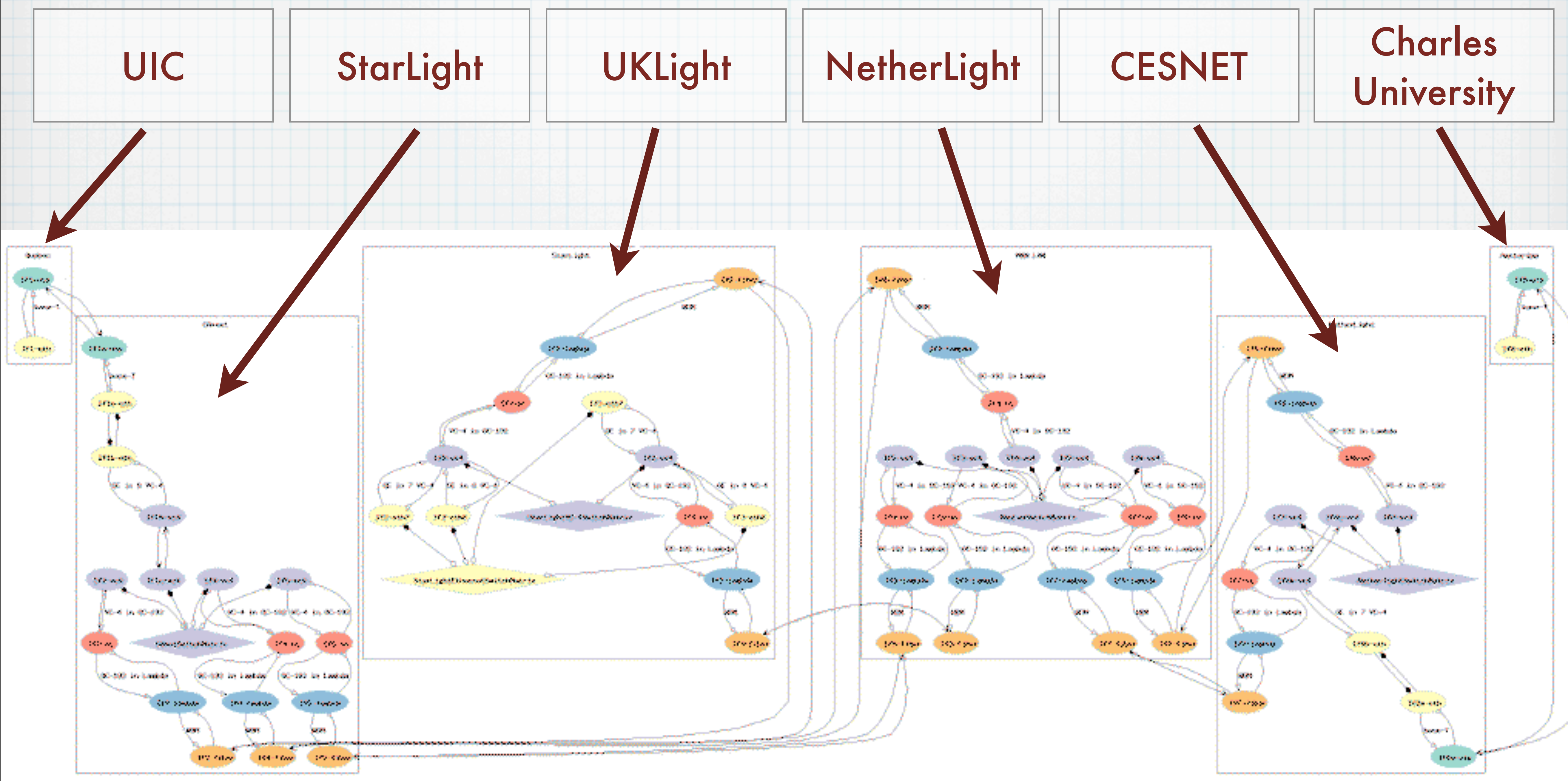
This is our network. It's an example based on a practical problem in the GLIF about 2004. Hybrid: different technologies/layers. Different adaptations between layers. Not all available. Question for the network engineers in the audience: please find the shortest working path from University of Illinois at Chicago to Charles University in Prague.



dinsdag 18 september 2007

1. model

Path Find Algorithm



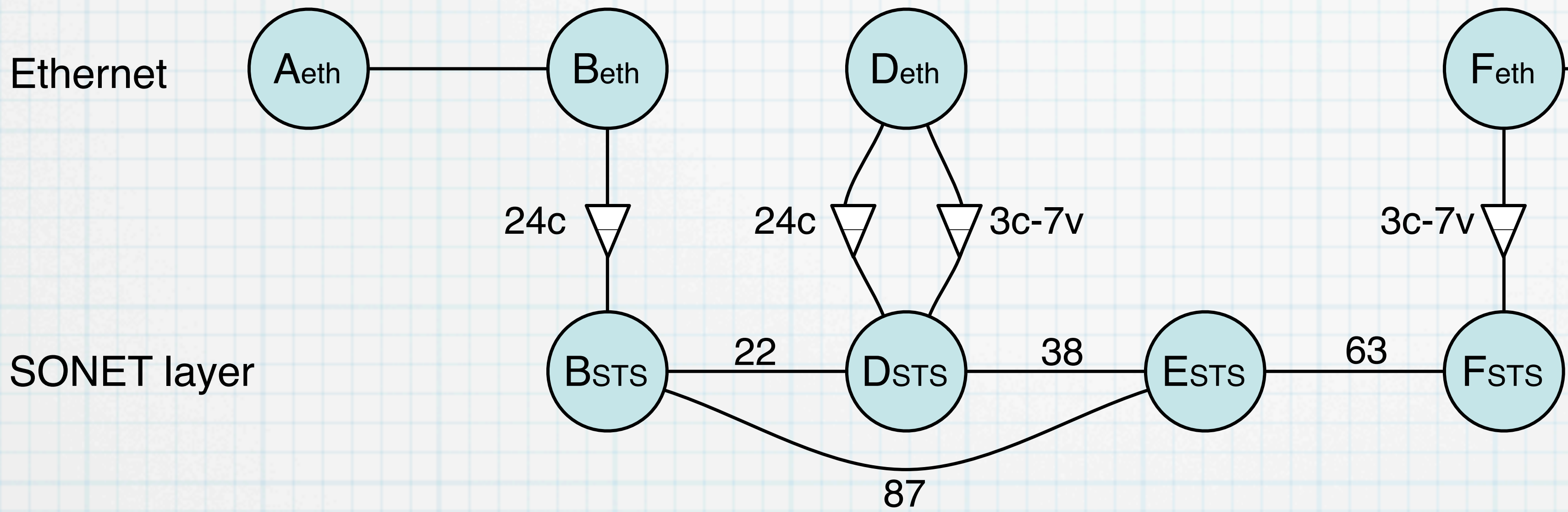
dinsdag 18 september 2007

Graph of logical interfaces (each colour is a different layer).
Diamond shapes are switch matrices (subnetworks in G.805).
Labels are adaptations functions. Labels are not represented in this visualization.

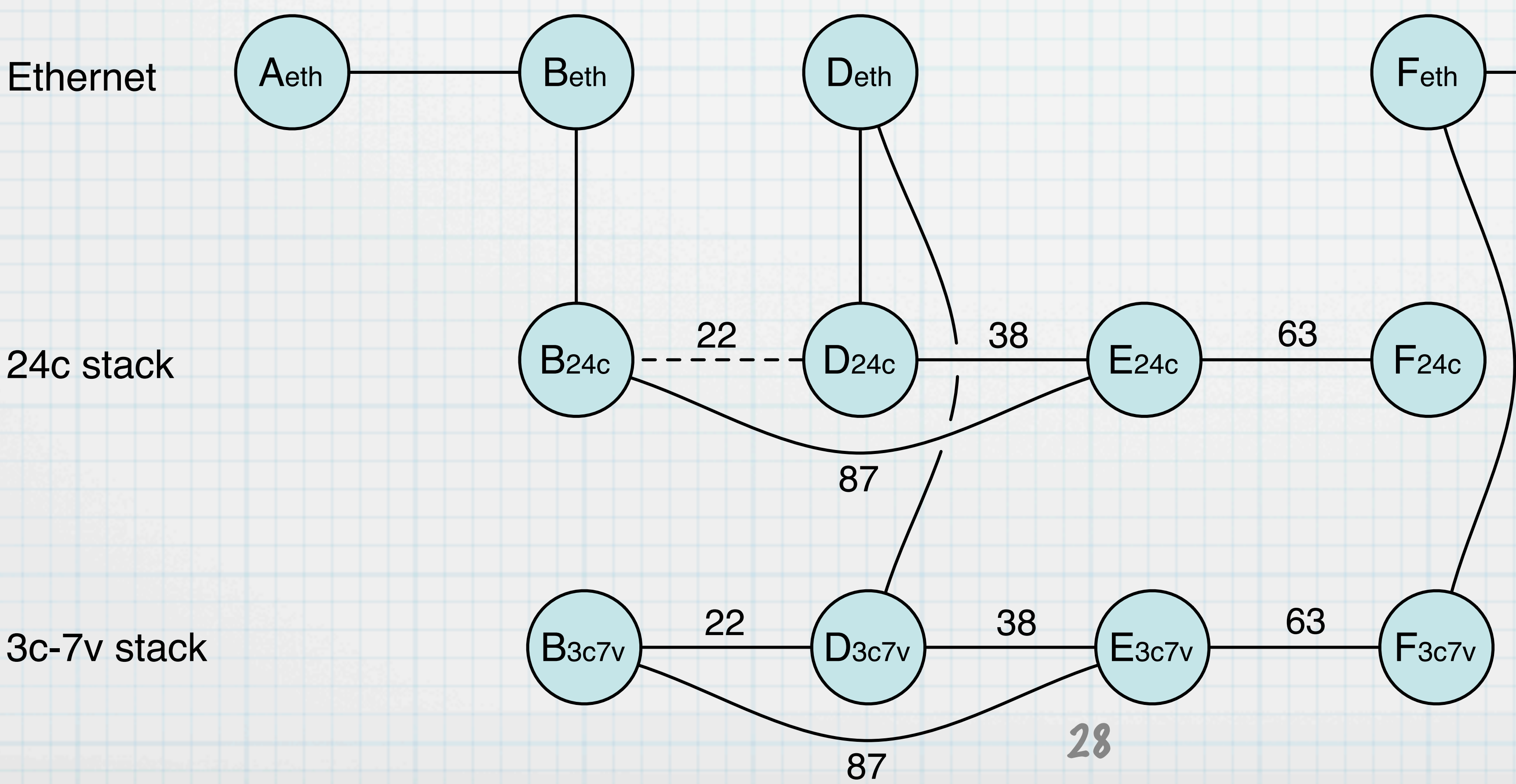
```
% pathfind.py PFAvailable
Using breadth first search algorithm to find a path from Quebec if1 to Amsterdam if8


             Starting point --> UIC          if1   Ethernet
                    Link to --> StarLight    if1   Ethernet
     Adaptation GE in 24 STS --> StarLight    if1   Ethernet over STS
        Through SONET switch --> StarLight    if4   Ethernet over STS
    Adaptation STS in OC-192 --> StarLight    if4   Ethernet over STS over OC-192
                    Link to --> NetherLight  if4   Ethernet over STS over OC-192
 De-adaptation STS in OC-192 --> NetherLight  if4   Ethernet over STS
        Through SONET switch --> NetherLight  if3   Ethernet over STS
    Adaptation STS in OC-192 --> NetherLight  if3   Ethernet over STS over OC-192
                    Link to --> UKLight      if3   Ethernet over STS over OC-192
 De-adaptation STS in OC-192 --> UKLight      if3   Ethernet over STS
  De-adaptation GE in 24 STS --> UKLight      if3   Ethernet
     Through Ethernet switch --> UKLight      if2   Ethernet
     Adaptation GE in 21 STS --> UKLight      if2   Ethernet over STS
    Adaptation STS in OC-192 --> UKLight      if2   Ethernet over STS over OC-192
                    Link to --> StarLight    if2   Ethernet over STS over OC-192
 De-adaptation STS in OC-192 --> StarLight    if2   Ethernet over STS
        Through SONET switch --> StarLight    if4   Ethernet over STS
    Adaptation STS in OC-192 --> StarLight    if4   Ethernet over STS over OC-192
                    Link to --> NetherLight  if4   Ethernet over STS over OC-192
 De-adaptation STS in OC-192 --> NetherLight  if4   Ethernet over STS
        Through SONET switch --> NetherLight  if6   Ethernet over STS
    Adaptation STS in OC-192 --> NetherLight  if6   Ethernet over STS over OC-192
                    Link to --> CESNET       if6   Ethernet over STS over OC-192
 De-adaptation STS in OC-192 --> CESNET       if6   Ethernet over STS
        Through SONET switch --> CESNET       if8   Ethernet over STS
  De-adaptation GE in 21 STS --> CESNET       if8   Ethernet
                    Link to --> CUni         if8   Ethernet
```


Result of path finding algorithm

Path Finding



Breadth first search algorithm (by F. Dijkstra, UvA)



Modified E.W. Dijkstra (by F. Kuipers, TU Delft)

Demo & Papers

See downstairs or

http://ndl.uva.netherlight.nl/

Basics Concepts

- Adaptation stacks
- Switch matrix
- Switching and swapping
- Labels
- Multiplexing (potential interfaces)

The Schema

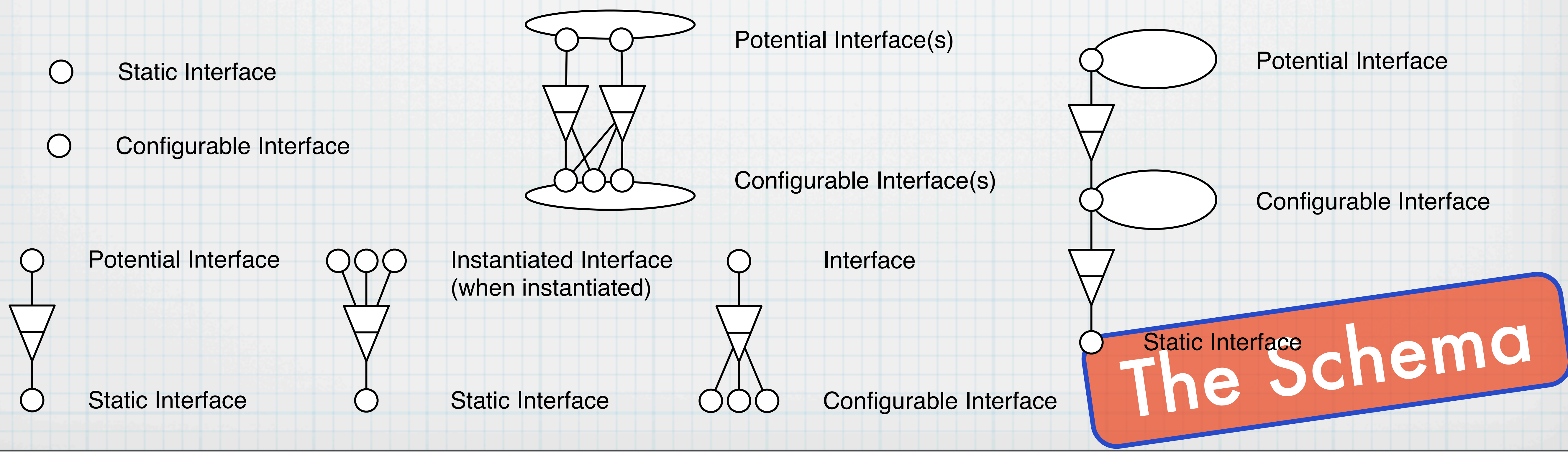
Advanced Concepts

- Optional vs. compulsory labels
- Ingress/egress label (packet switching)
- Internal labels (Untagged Ethernet)
- Internal adaptation stack
- Inverse multiplexing (> 1 server layer)
- Multicast switching
- Broadcast switching

The Schema

Interfaces

- ## Static Interface
  Fixed interface. Can not be changed in any way.
  laser at 1310 nm
- ## Configurable Interface
  Interface always exists, but can still be configured.
  tunable laser
- ## Potential Interface
  Abstract interface. 0, 1 or many of these interface can be configured.
  "It is possible to create Tagged Ethernet channels"
- ## Instantiated Interface
  Instantiation of a Potential Interface.
  Configured timeslot on VC-4 layer.

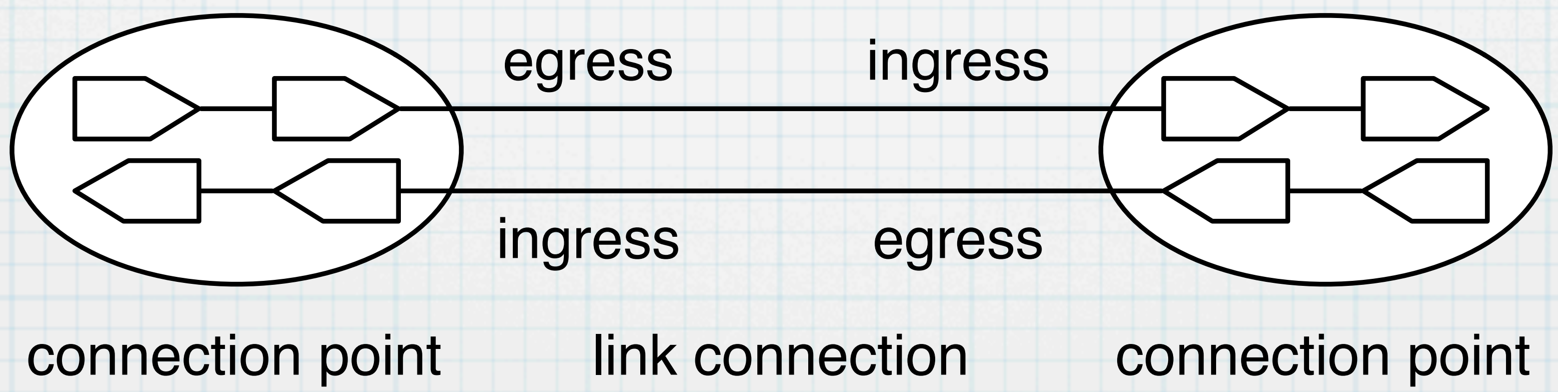


Semantic Challenges

X linkTo Y, but not **Y linkTo X** means:

- a unidirectional link
- only X is configured, Y is not (but X would accept data from Y).

Which of the connection points below is configured (admin up/link up), and is there a fiber?



Challenges

Semantic Challenges

What does a **Potential** or **Available** configuration mean:

- Is it technically possible? Possible without breaking other connections ?
  If so, what does "breaking" mean? What if I reconfigure the other switch connection? Is that broken?
- Is it administratively possible?

We distinguish between actual (is configured/static), potential and available

Challenges

Semantic Challenges

If a layer has a label, does it have to exists for an actual Interface?

- The Ethernet label is the VLAN (IEE 802.1Q) label.
- It is only embedded in the data itself for Ethernet over Ethernet (Tagged Ethernet).
- For untagged Ethernet, it is used for switching within a switch matrix
- An untagged channel can have different "label" at each end.

Our solution: we use the "empty label" as concept, but still sometimes it MUST be empty, sometimes it MUST NOT be empty

We only use the IEEE 802.1Q label as the actual label (in the GMPLS sense), and the VLAN tag as an "internal label", for switching only.

Challenges

Logical Challenges

Give me all "switchTo" means:

Depends on:

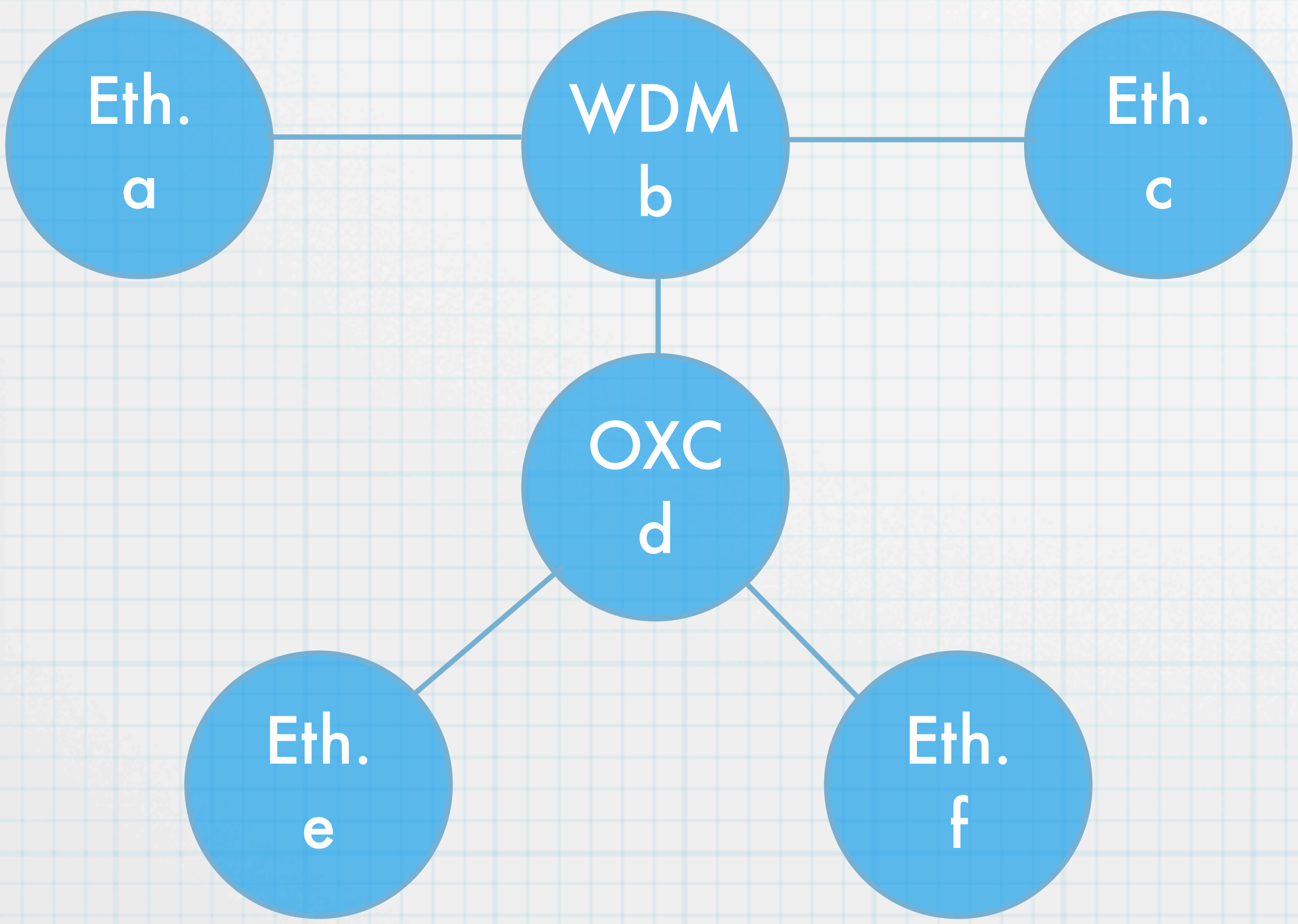
- Question: do you want Actual/Potential/Available switchTo?
- What kind of interfaces are we talking about: Static/Configurable/Potential/Instantiated
  Do we return one or two switchTo for a Potential and Instantiated interface?
- What type of switch matrix, if any: None (patch panel)/Unicast/Multicast/Broadcast
- Can the switch matrix convert between labels (switching & swapping)

Challenges

Logical Challenges

When is a switchTo (subnetwork conncetion) in use?

- We can re-use a connection at a lower layer, as long as the labels are different on higher layers (different channels).



Example:

Ethernet A -> C need to go to E and F:

- A -> E: 10 Gb/s LAN PHY
- E -> F: 10 Gb/s WAN PHY
- F -> C: 1 Gb/s

How/when to detect that this is not possible due to a conflict at OXC d?

Challenges

Logical Challenges

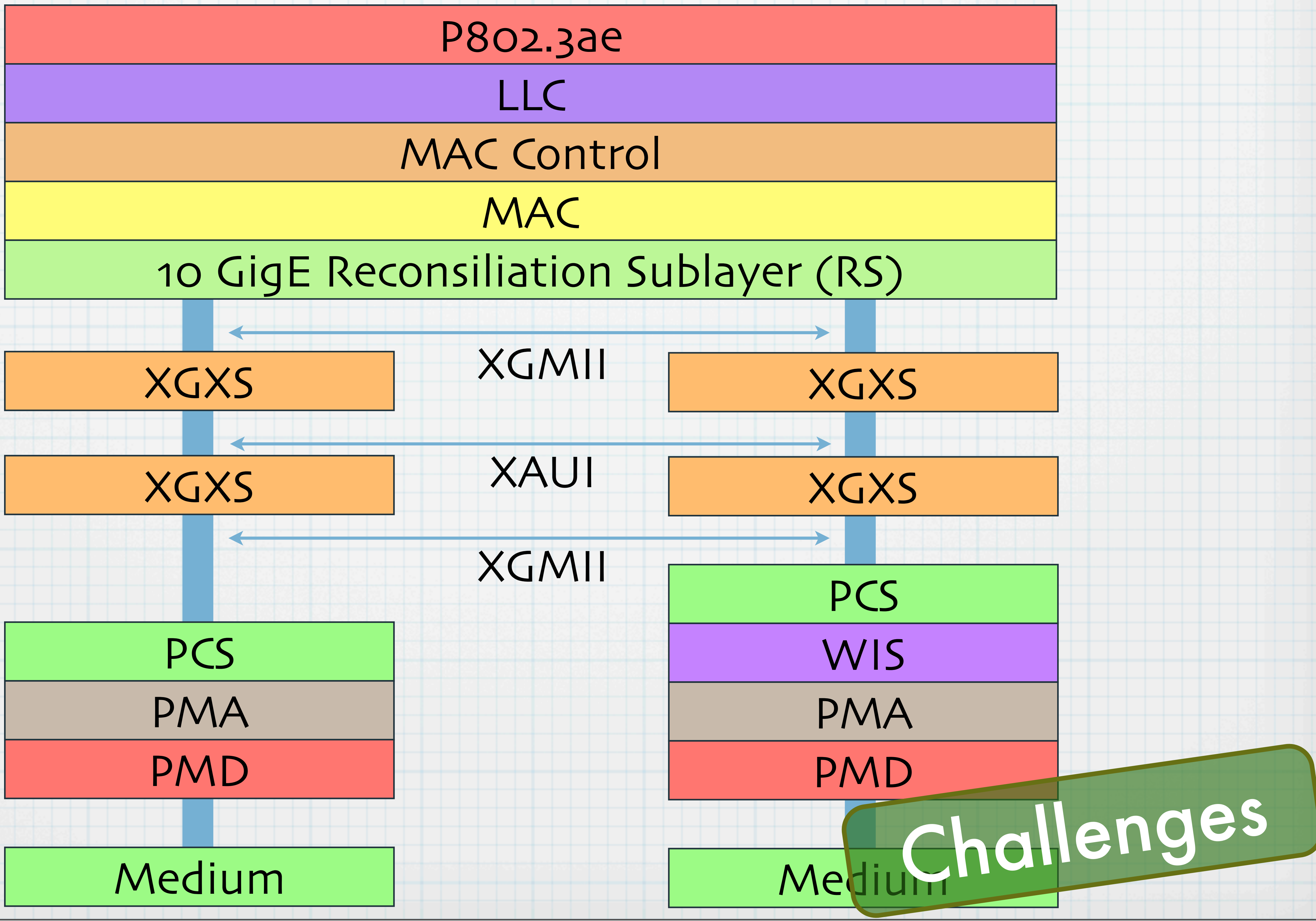
For a path, 4 channels over the same client layer are required:

- A. Must have label in set {3,4}
- B. Must have label in set {3,4}
- C. Must have label in set {3}
- D. Must have label in set {4-11}

How to detect this is not possible? If we sequentially pick a label for each channel, we may get a false negative.

Challenges

Practical Challenges



Thank You