Network topology descriptions in optical hybrid networks

Status of This Document
This document provides information to the Grid community regarding the context for the work in the OGF NML-WG. Distribution is unlimited.

Contributors to this document:
Paola Grosso, UvA
Anand Patil, DANTE
Pascale Primet, INRIA
Aurélien Ceyden, ENS-Lyon
Jason Zurawski, Internet2
Aaron Brown, Internet2
Martin Swany, University of Delaware
Freek Dijkstra, SARA
Jeroen van der Ham, UvA

Abstract

The NML-WG goal is to define a schema for describing network topologies. This schema is to be used in various ways, including:

- lightpath provisioning applications to exchange topology information intra and inter domain;
- description of networks for reporting performance metrics.

This document constitutes Deliverable 1 of the working group. It provides a detailed overview of the framework in which the working group operates, detailing the already existing topology schemas and providing the basis for the integration of the various projects.

Contents

1. Introduction

As e-Science applications have become more distributed in nature and can take advantage of the various Research and Education (R&E) computing Grids, they have also started to have more specific requests on the performance and services offered by the underlying networks. If data moves from one location to another, from one Grid cluster to another, the time the transfer takes (the *throughput)*, the variation in arrival time at the destination (the *jitter*), or the time for a packet to get to the destination (the *delay*), all become critical components to be accounted for in the computational model of the application.
To provide applications with guaranteed, reliable and reproducible network services the R&E networks around the world have started to offer *lightpaths* to end users. Lightpaths are dedicated circuits in the network of which the application is the sole user. The user consequently has control on the traffic and has no competition for network resources. This results in a more reliable network service.
The trend is clear looking at the network architecture designs in many of the R&E networks in the last years. These networks have moved to a so-called *hybrid* model: routed IP services for traditional network use are complemented by lightpath services. Just to name a few we can think of the SURFnet6 network in the Netherlands with its Optical Private Network service, or the European network GÉANT2 with its Premium IP service and point-to-point circuits or the Internet2 network in the USA with the HOPI project.

Lightpaths are often called *lambdas* as they are provisioned for the end user as dedicated wavelengths in the DWDM network. Analogous to computing Grids, the goal is to have *lambda Grids* in which network connections are created ad-hoc and reliably when needed. To achieve this, there are two important aspects that still need a solution and are being addressed by the research community: a true dynamicity in the setup of paths and the extension of the path beyond a single domain to a true multi-domain setup.
If the creation of new circuits takes too much time and requires manual intervention, applications cannot interactively and in real time make use of them. Lighpaths need to be consistently provisioned across network domains so that applications can effectively communicate end to end. Both issues are tackled by the provisioning systems under development in all the R&E networks.

Interoperability of provisioning systems requires communication of network information among them. Network topologies, network capabilities, scheduling information, and much more, need to travel between domains. This working group focuses on the definition of a standardized model for network information exchange to be used in lambda Grids for the setup of lightpaths.
In the following sections we will give an overview of the existing schemata for topology description in use within the international research community. This list is not intended to be exhaustive, but sets a ground for the definitions of the necessary elements in a standard Network

Markup Language.

2. NDL – Network Description Language

NDL[1] is an information model developed by researchers at the University of Amsterdam to describe (computer) networks. NDL comprises of a series of RDF schemas that categorize information for network topologies, network technology layers, network device configurations, capabilities, and network topology aggregations.
The main use-cases so far have been generation of network maps, lightweight offline path finding and more recently multi-layer path finding, and network topology information exchange.
NDL has been used primarily in the research community in the Netherlands: UvA, SARA and SURFnet. It also has been applied to the GLIF Optical Lightpath Exchanges (see: http://www.glif.is).

2.1 RDF and Semantic Web

Two of the main differences of NDL over other topology schemata are its use of RDF – Resource Description Framework – as syntax and the grounding of the model in the Semantic Web framework.
The extensive reasons for the RDF choice instead of XML are explained in a document available in the NML-WG project web site (ref: http://forge.ogf.org/sf/go/doc14257?nav=1). To roughly summarize this document we can say that NDL chooses RDF because:

- It allows easier exchange of information between independent domains,
- It is easily extendible and it allows integration of independent data models developed in other fields, by other researchers.

Several tools that consume RDF data are publicly available and make the use of this syntax straightforward.

2.2 NDL schemas

The Network Description Language (NDL) is a modular set of schemata, defining an ontology to describe computer networks:

- The topology schema describes devices, interfaces and connections between them on a single layer;
- The layer schema describes generic properties of network technologies, and the relation between network layers;
- The capability schema describes device capabilities;
- The domain schema describes administrative domains, services within a domain, and how to give an aggregated view of the network in a domain;
- The physical schema describes the physical aspects of network elements.

2.2.1 NDL topology schema

The classes and properties in the topology schema (Fig. 1) describe the topology of a hybrid network, without detailed information on the technical aspects of the connections and their operating layer. The idea is that through this lightweight schema we can provide an easy toolset for basic information exchange and path finding.

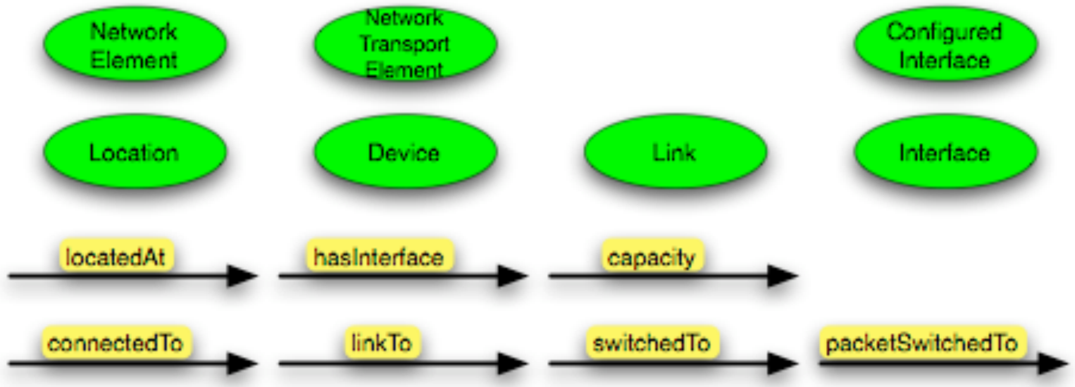

Figure 1 - NDL topology schema

The *linkTo* property corresponds to a link connection or edge, while the *connectedTo* property corresponds to a network connection or a path. linkTo and connectedTo describe external connections, between two devices.
The *switchedTo* and *packetSwitchedTo* properties define internal connections within a device: the configuration of a device.
The immediate applications of the topology schema are visualization of network maps and input to path finding systems.

2.2.2 NDL layer schema

The topology schema defines network topologies on a single layer. The NDL layer schema allows applications to describe multi-layer networks, like hybrid networks.
The NDL layer schema is based on a formal model, which uses ITU-T G.805[2] functional elements and the concept of labels as described in GMPLS[3].

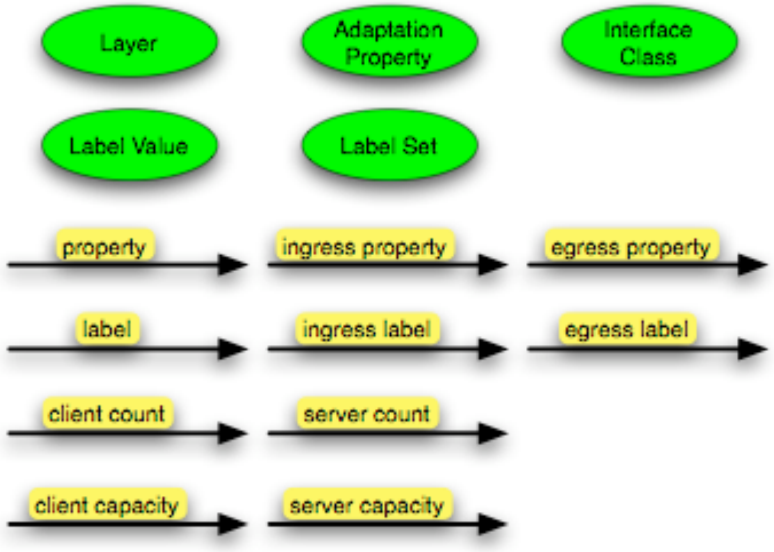

Figure 2 - NDL layer schema

A Layer is a specific encoding in network connection; most Layers have an associated Label Set that defines which channels are used to make switching decision in a device. For example, the label on the wavelength division multiplexing (WDM) layer is a wavelength.

Each Interface instance operates at a certain Layer. When data from one layer needs to be encapsulated in another layer we use Adaptation. The client (layer) and server (layer) refer to the Layers before and after the Adaptation.
This layer schema does not define actual adaptation functions, but instead provides a common vocabulary to describe technologies, layers and the relation between layers. We make use of the layer schema in a tool for path finding across multiple layers.

2.2.3 NDL domain schema

The NDL domain schema (Fig. 3) defines administrative domains and the services offered by a domain. It allows network operators to provide an aggregated view of their domain to neighboring domains, rather than the full topology.

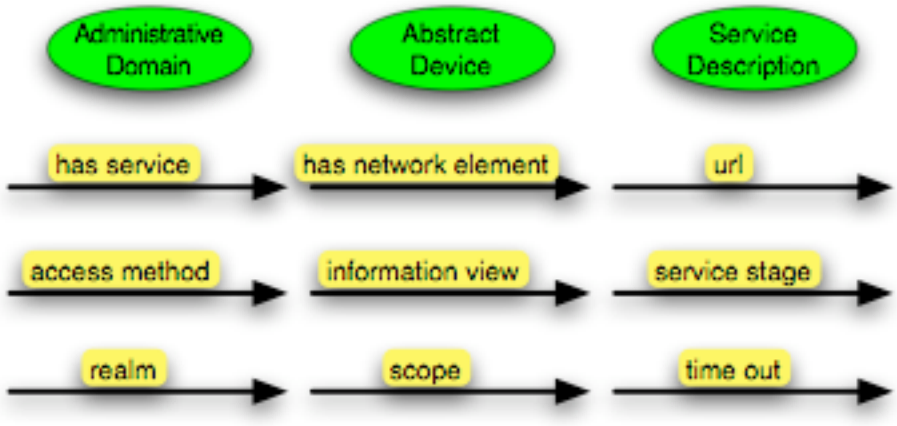

Figure 3 - NDL domain schema

An important concept in the domain schema is that of Service Descriptions. Service descriptions allow domains to point applications to the (web)services they offer.

The idea is that domains publish static information in NDL, and provide a webservice for dynamic information or more confidential data, like reservation requests. Furthermore, different domains will have different opinions on what is ``static'' and ``non-sensitive''.

3. perfSONAR

PerfSONAR[8] is a software infrastructure designed to perform and otherwise aid in the collection of network performance data. This data, once exposed, can be used in solving end-to-end performance problems on paths crossing several networks. The suite of services offered by perfSONAR forms an intermediate layer designed to sit between data collection tools and analysis tools, including visualization, that Network Operations Centre (NOC) and Performance Enhancement and Response Team (PERT) staff rely upon. Using well-defined protocols it is possible to exchange both performance information such as network measurements as well as information about the design of the network, such as topology, through specially purposed software services.

The perfSONAR protocols are based on the work of the Network Measurement Working Group (NM-WG)[9] to define an XML representation of the various forms of network measurement. A key concept of this work has been the distillation of information collected from performance measurement into two fungible units: metadata and data. The metadata encompasses the static, or similar information of a series of measurements. The data section pertains to the dynamic, or time dependent portions of a measurement. An example of this natural split is a series of latency observations performed by the Ping measurement tool. The static information for the series of measurements would be the two hosts involved as well as any parameters used to start the tool. The actual latency measurements, as well as the times they were observed, would make up the data portion.

In addition to representing measurements with this flexible XML format, perfSONAR has applied this principal to the message exchange used between services as well as the representation of topological entities. The later work is used extensively in the DICE[10] Control Plane Community as well as with the Internet2 Dynamic Circuit Network (the successor to HOPI) to represent, describe, and control network functionality through web services interfaces. A specific perfSONAR service, the Topology Service (TS), is used to accept topological information via registration from other services and infrastructures, and is able to answer queries regarding its internal contents. All information is stored in a native XML format and related technologies such as XPath and XQuery allow for straightforward discovery of information.

3.1 perfSONAR Topology Schema

Originally designed to handle the topological elements of measurement (i.e. network interfaces, application layer endpoints, etc.), the perfSONAR topology schema has undergone significant restructuring to allow the description of general networks, including hybrid networks. The schema is designed around a handful of general-purpose elements: domains, nodes, ports, links, services, networks and paths. These elements are then used to represent any network entity or service. XML namespaces are used to allow for more specific versions of the basic elements to be defined. This allows for technology-specific versions of the elements (e.g. an Ethernet link or an HTTP Web Service) to be created as the need arises without requiring modification of the base classes. These technology-specific namespaces are defined in a basic object-oriented manner allowing applications, such as path-finding applications, to be written in a general manner, but still work with the network elements no matter what namespace they use.

New technologies can be written by adding a new namespace containing versions of the above elements with the technology specific elements added. Each element of this schema constitutes an entity (e.g. endpoints, links, circuits, paths) and relationships are created by linking identifying elements.

3.1.1 Topology Elements

Domains – A domain corresponds to an administrative domain and is used to divide the entities in a topology into administrative groupings. In a normalized topology schema, all the nodes, paths, networks and bidirectional links for a given domain will be defined inside the domain element for that domain.

Nodes – A node element is used to describe network entities like hosts, network devices, abstract concepts like “sites” or even an abstracted view of a domain. These can be thought of as a typical node in a graph context. In a normalized schema, the node elements will all be defined inside the domain element.

Links – A link element can be used to describe a connection between two nodes. This could correspond to a physical connection like an Ethernet link or could be a more abstract link like a VPN or a TCP connection. There are two general types of link elements: unidirectional and bidirectional. In a normalized topology, unidirectional links are defined inside the interface that can write to them. Bidirectional links, however, are defined inside the domain in which they exist.

Ports – A port element is used to describe a connection point between a node and a link. These types of elements could be used to describe Ethernet interfaces, listening TCP sockets or any other entity that could read or write to a link.

Services – Service elements are used to describe the services offered by a network element. In a normalized schema, these elements will be defined in the 'node' element of the network entity from which they are offered.

Path – Path elements are a list that may contain links, ports, nodes or domains, which describe a path through a topology. These elements are used to describe the path a multi-domain circuit is using.

Networks – Network elements can be used to describe a group relation between a variety of network elements. A network element could, for example, describe all the disparate network elements in a VLAN.

4. GÉANT2

GÉANT2 [4] is the pan-European research and education network connecting the European National Research and Education Networks (NRENs). The GN2 project within which the network is being operated also includes an integrated research program [5] and development of advanced services [6] for network users. Among others, these activities include network monitoring, network security, bandwidth on demand, cross border fibers, authentication and authorization infrastructure, multi-domain premium IP service etc.

4.1 Common Network Information Service (cNIS)

Several activities within the GN2 project as well as external clients, for example Grid middleware, require network information. These activities were developing, populating and storing their own unique data store according to their respective requirements. This would lead to duplication of effort, increased load on network elements and replication of data as well as potential inconsistent views of the same underlying network.

The aim of the cNIS is to provide an interface to a unified repository of all relevant network information about a single domain's network infrastructure for client applications. There will be one instance of cNIS per domain, with data being collected and stored separately for each domain. The collection of cNIS instances work together so that a client application can receive whatever network information it requires, even if this spans several domains. Each NREN is expected to operate an instance of cNIS themselves.

The success of cNIS depends upon the efficiency of automatic data collection, population and verification. A plug-in architecture for this is currently under development.

4.2 cNIS Use Cases

The design of cNIS was driven by the use cases and requirements of the client applications. Different client applications have different, and often conflicting, requirements. For example some clients wanted to know the physical topology of the network down to interface cards and patch cables, whereas others only wanted abstract connectivity information.

Some design considerations that influenced the schema are as follows:

- ability to store past, present and future topology data

- ability to store technology specific data
- ability to convert technology specific data to different levels of abstractions
- ability to extend to other technologies
- ability to define allowed relationships between elements
- ability to store dynamic information like LSPs, VPNs, VLANs, circuits etc
- ability to link information from multiple domains using lookup services
- ability to search for specific topology elements
- ability to find paths between end points

4.3 cNIS Schema

The cNIS schema is an RDBMS schema rather than an XML schema. The background, scope, design, and data model of the schema is described in a GN2 deliverable [7].

The schema does not make any reference to the standard OSI 7 layer model as technologies like Ethernet and MPLS do not map cleanly onto an OSI layer and techniques such as tunnelling may lead to a situation where a lower-layer protocol runs on top of a higher-layer protocol, leading to confusion.

The schema can be divided into:

- Common schema
- Technology specific schemas
- Extensions

4.3.1 Common Schema

The common schema includes generic entities (such as node, interface and link). This means that the model does not contain any direct or hard-coded relationships between technology specific tables, and so the schema remains flexible and open to extensions for any other technologies. The common schema adds a time dimension to the data store.

4.3.2 Technology Specific Schemas

The technology specific schemas describe individual technologies. Currently SDH, Ethernet and IP technology schemas are defined with MPLS draft under development.

4.3.3 Extensions

Some requirements from activities like end-to-end monitoring of cross border fibers, traversing multiple administrative domains, do not fit easily into the common schema or technology specific schemas. For such cases the schema design is extensible to accommodate application specific requirements.

4.4 cNIS Interfaces

cNIS provides two different types of interfaces:

- A management interface for system users to perform various administrative tasks and visualization.
- An operational interface for external applications.

Some of the client applications had already defined interfaces for retrieving information from cNIS, for example PerfSONAR uses NMWG schema as described in section 3. To ensure a smooth transition for the applications, cNIS decided to honor all these individual interfaces. Ideally these bespoke interfaces will be temporary, such that as cNIS and its clients evolve all applications will be able to work towards one common unified interface or in other words a standard schema.

4.5 cNIS Future Work

For the remainder of GN2, the project will investigate potential extensions to the scope of the cNIS to possibly include the storage, retrieval and analysis of real-time networking data, advanced visualization of database contents and storage of multi-domain information, as well as support for a “trouble ticketing system” and multi-domain circuit provisioning.

GN2 supports NML-WGs effort for defining a standard schema and will actively promote its use by the cNIS client applications.

5. Topology descriptions in Grid'5000

5.1 Grid’5000 overview

Grid’5000, is a 5000 CPUs nation-wide Grid infrastructure for research in Grid computing. Grid’5000 is designed to provide a scientific tool for computer scientists similar to the large-scale instruments used by physicists, astronomers, and biologists. Grid'5000 serves as an experimental testbed for research in Grid Computing at all levels: from Application, Programming Environment, Middleware, Operating Systems, Network Protocols. It is a research tool featured with deep reconfiguration, control and monitoring capabilities designed for studying large scale distributed systems and for complementing theoretical models and simulators.

Up to 17 French laboratories are involved and 9 sites are hosting one or more clusters of about 500 to 1000 cores each (see Fig. 4) A dedicated private optical networking infrastructure, provided by RENATER, the French NREN and composed of 10Gb/s lambdas is interconnecting the Grid’5000 sites. Two international interconnections are also available: one with DAS-3 in the Netherlands and one with Naregi in Japan.

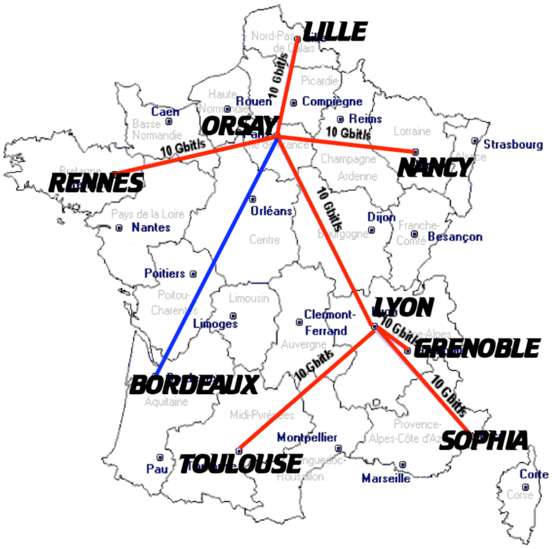

Figure 4 - Grid'5000 topology

5.2 Grid'5000 usage

The main goal of Grid'5000 is to provide users with the ability to deploy their own operating system on the resources they reserve for a limited number of hours.
Grid'5000 exposes two tools to implement these features: OAR, a reservation tool, and Kadeploy, an environment deployment system. OAR offers an accurate reservation capability (CPU/Core/Switch reservation) and integrates the Kadeploy system. With Kadeploy, each user can make his own environment and have a total control on the reserved resources.

5.3 Network monitoring tools and their usage

In terms of monitoring tools, users have access to the RENATER Monitoring tool, the Grid'5000 infrastructure description, Nagios tool, Ganglia and SFlow (only at Lyon).
The RENATER Monitoring tool provides a view of all connections on the Grid'5000 backbone. With it, the users can see the availability and the usage of the global links.
The Grid'5000 infrastructure description gives all information about the different Grid'5000 platforms. It shows, graphically, on all sites their internal connections, that means, what kind of network hardware is used and what are the different links between them.
The Nagios tool is used to warn the administrators about connectivity and services problems.
Ganglia provides users the bandwidth usage per node and other information, not necessarily network specific, such as the disk space available, the CPU usage and so on.
The Sflow tool is an experimental monitoring tool, which provides a sampling of the network traffic.

We can distinguish three types of actors in Grid’5000: 1) researchers and experimenters, 2) application users, 3) administrators. These different communities do not require and use the same type of network monitoring tools. The researchers and experimenters and administrators are using all of them while applications users only use the provider tools and the infrastructure description. We have observed that the infrastructure description is widely required. This is why the Grid’5000 administrators have designed a two-level graphical network description of the tesbed.

5.4 Grid’5000 graphical network description

The Grid’5000 graphical network description is defined in two levels. The first level gives a general view of the topology (Fig.5), while the site view explicitly shows all the networking elements of a site (Fig.6).

These graphs inform the viewer on the long distance link capacities, but do not give the details of the internal resources in the provider network. These resources are hidden to any Grid’5000 community. For example, in Lyon and Paris, the Grid’5000 dedicated lambdas are converted in 10Gb/s Ethernet links and interconnected with a 10Gb/s switch. The Grid’5000 community does not have access to the details of these switches (backplane capacities, buffer space, equipment type, activated functions, etc.). The first-level graph only shows the service (virtual links) provided by RENATER in terms of their capacities (1Gb/s, 10Gb/s, 20Gb/s) and their medium (copper, optical). On the other hand, the second-level graphs, the site view, gives the detailed equipments information, such as: type (switch/router), vendor name, IP address. It also gives a logical description of internal resources such as: Node/Frontend name, or Node network.

This kind of infrastructure representation presents several problems: 1) the representation is static, 2) the representation is only graphical, 3) it requires an heavy synchronization and updating work.

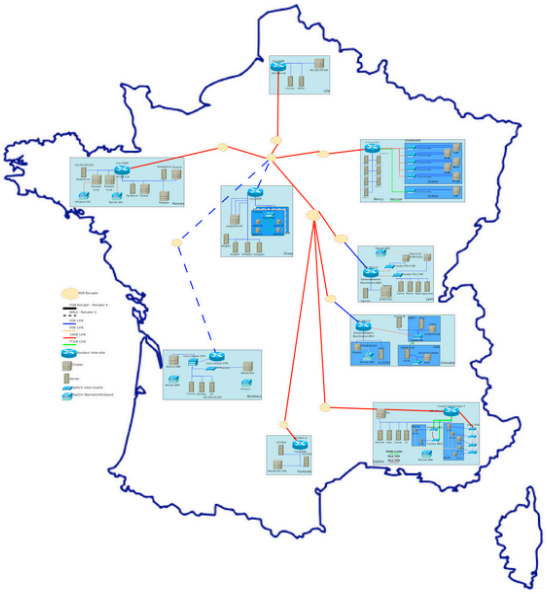

Figure 5 - Grid'5000 topology view

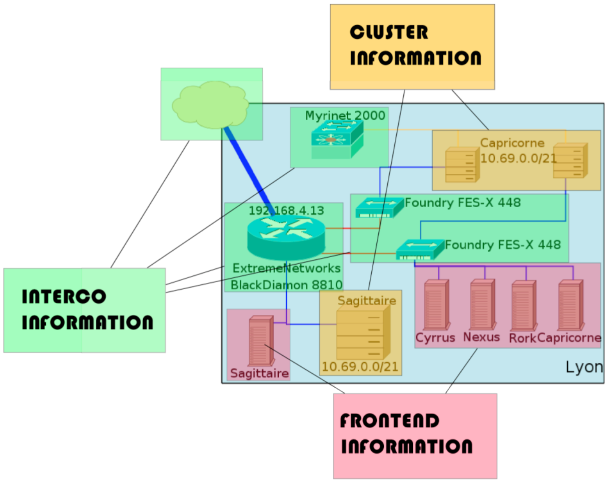

Figure 6 - Grid'5000 site view

5.4.1 Infrastructure and topologies description needs in Grid'5000

In this section, we try to summarize the needs expressed by each Grid'5000 community:
The users (researchers, developers, grid applications users) want to

- Know which devices their traffic crosses
- Know what are the theoretical weight of links (edges)
  - link capacity
  - link latency
  - orientation (unidirectional, bidirectional..)
- Know what are the links status (up or down, available bandwidth)
- Have a multiview of the network
  - Hardware view
  - Transport layer view
  - Application view (user space)

- Filter the different information

Generally, from this information they try to infer the end-to-end performance they obtain and to debug it.

The administrators need a view of all connections (compute network, admin network, etc.). They want to be able to:

- Locate the resources
- Access to the hardware representation; all information about the hardware is interesting (router vendor, nodes type)
- Identify the routing path

This shows that all types of users need both a graphical view, and also query tools.
Some application developers and researchers need an API to directly access the network description with which their network-oriented services interact: Bulk Data Transfer Scheduling Service (BDTS), Network Virtualisation and Reconfiguration Service (SRV), Virtual Cluster/Virtual Network Overlay Composer (HiperCal/HiperNet), High Speed Transport Test Service (HSTTS).
In this context, data models for network topology information exchange are highly desirable for several types of usage.

6. Toward a standard schema

We described in the preceding sections (some of) the current efforts in the definition of suitable data models for network topology information exchange. It is obvious that the convergence towards a common standard model facilitates its adoption from the various provisioning systems that in turn can more easily interoperate.

6.1 Use Cases

It has been suggested in the WG that standard network description language would be employed for:

- Path finding
- Visualization
- Asset management (inventory of network devices)
- Network measurements

In the following section we provide a more detailed description of one of these use cases and the role the standard NML will play in them.

6.2 Path finding

Finding lightpaths is the most interesting and most relevant application for the NML schema.
In general there are two ways in which inter-domain paths can be found:

- In-band, letting the network devices autonomously negotiate among themselves.
- Out-of-band, letting the provisioning system exchange topology information and perform the calculation.

The in-band approach requires network domains to let information about their internal topology cross the domain boundary autonomously and seamlessly. For this reason many providers prefer an out-of-band, or even offline approach in which the information on the network is exchanged using predefined schemas and there is more control on the data that is externally exposed. In the latter case the NML finds its best applicability.

Which data is necessary to be exposed to find a lightpath?
We need:

- Topology information – which devices connect to which devices
- Capability information – what kind of services are available between connecting interfaces

- Scheduling information – (this might be optional) – what are the available times for the requested service

7. Conclusions

This document provides the research and business context in which the work of the NML-WG takes place. We described a few of the existing network topology efforts and provided details on their implementations and use cases.

8. References

[1] NDL webpage URL: http://www.science.uva.nl/research/sne/ndl
[2] ITU-T Recommendation G.805: Generic functional architecture of transport networks, URL: http://www.itu.int/rec/T-REC-G.805/en
[3] E. Mannie, "Generalized Multi-Protocol Label Switching (GMPLS) Architecture", URL: http://www.ietf.org/rfc/rfc3945.txt
[4] GÉANT2 website URL: http://www.geant2.net
[5] GÉANT2 Joint Research Programme website URL: http://www.geant2.net/server/show/nav.753
[6] GÉANT2 Services website URL: http://www.geant2.net/server/show/nav.744
[7] GÉANT2 common Network Information Service Schema Specification URL: http://www.geant2.net/upload/pdf/GN2-07-045v4-DS3-13-1_common_Network_Information_Service_Schema_Specification.pdf
[8] perfSONAR website URL: http://www.perfsonar.net
[9] NM-WG website URL: http://nmwg.internet2.edu
[10] DICE website URL: http://www.geant2.net/server/show/conWebDoc.1308