A Distributed Topology Information System for Optical Networks Based on the Semantic Web

Jeroen van der Ham [a,b,*] Freek Dijkstra [a] Paola Grosso [a]
Ronald van der Pol [c] Andree Toonk [d] Cees de Laat [a]

[a] *System and Network Engineering Group, Universiteit van Amsterdam, Kruislaan 403, 1098 SJ Amsterdam, The Netherlands*

[b] *TNO Defence, Security and Safety, Postbus 96864, 2509 JG Den Haag, The Netherlands*

[c] *SARA, Postbus 94613, 1090 GP Amsterdam, The Netherlands*

[d] *BCNET, 7300 SFU Harbour Centre, Vancouver BC, V6B 5K3, Canada*

Abstract

There is a trend in the research networking community to provide e-Science applications with dedicated connections instead of shared links. *Lightpaths* are the way to guarantee the appropriate level of service in terms of bandwidth and latency.

NDL – Network Description Language – is a data model offering users and providers of lightpaths with a common ontology to describe topology information of hybrid optical networks. The strength of NDL is that it supports a wide range of applications, including path finding, visualization and asset management, via the definition of a common data model to exchange network descriptions. Since NDL is based on Semantic Web techniques, it is straightforward to relate NDL with application-specific ontologies. In this paper we present the current status of the NDL schemas and its use in several applications.

Key words: Network Descriptions, Semantic Web, Resource Description Framework, Hybrid Networks, Inter-Domain Pathfinding

* Corresponding author.
Email address: `vdham@science.uva.nl` (Jeroen van der Ham).

1 Introduction

Optical networking has seen a more widespread deployment in the research and education community over the recent years. In particular NRENs (national research and educational networks) have deployed more and more of the so called *hybrid network* infrastructures. A hybrid networks is a network where the same optical infrastructure provides regular IP connectivity, as well as the possibility for separate lightpaths. A motivating factor for these new architectures are e-Science applications running on the Grid, which require network services tailored to their specific computational needs. Lightpaths are the solution to these demands, given they can provide private connections with guaranteed bandwidth, latency and privacy.

These new architectures have brought a whole new set of problems and challenges, especially in management and provisioning of lightpaths. The process of providing a dedicated optical connection to the end user is far from an automated process. Even when lightpaths are confined within a single domain the whole process typically takes a few days. Users and network engineers need to interact multiple times to define the characteristics of the desired service, before the actual configuration can take place. If the desired lightpath spans *multiple* domains the problems is even more complex to solve. In this case the provisioning system needs to know the rough global topology of the networks involved. It also needs to determine the compatibility of each segment and the relevant adaptation services. Along with all the other parameters, these settings must be configured correctly on both sides of the connection, so they must clearly be communicated to all parties involved. See [1] for a discussion on the parameters involved in requesting a lightpath.

In this paper we present our latest work on the Network Description Language (NDL) which is meant to address the issues in lightpath provisioning we just described. NDL defines an interoperable language for network topology descriptions and provides a common data model to lightpath providers and users to exchange network topology information. NDL facilitates a wide range of applications, including path finding, visualisation and asset management.

The remainder of this paper is structured as follows. The remainder of this section discusses related work. In section 2 we introduce the Network Description Language. Existing applications of NDL are discussed in section 3, this is followed by section 4 where we discuss future challenges and research on combining NDL with domain abstractions, policy, and monitoring. In section 5 we present the conclusion of the paper.

1.1 Related Work

It is clear to the networking research community that the whole provisioning process needs to be improved and automated, in order to scale to multiple domains. There are already some applications which automate part of this process. For example DRAC [2] and UCLP [3] provide solutions for intra domain lightpath provisioning. However, these applications currently only extend to multiple domains in so far that these use the same software. A more generic solution can be found in GMPLS [4]. This approach has been implemented in DRAGON, Dynamic Resource Allocation via GMPLS Optical Networks [5]. DRAGON also provides wrappers around network equipment which do not support GMPLS. The IETF is currently working on GMPLS extensions to support inter-domain provisioning [6]. However GMPLS is not yet widely available, and the inter-domain provisioning is only possible when *all* network components support GMPLS, and are willing to use inter-domain GMPLS. The interoperability problem for multi-domain lightpath provisioning has been acknowledged by organizations such as IETF [6][7], OIF [8], and OGF [9].

2 Network Description Language

The Network Description Language (NDL) is a modular set of schemas, defining an ontology to describe computer networks. In the next section we explain why we have chosen RDF over XML. This is followed by a section where we introduce the different NDL schemas.

2.1 Semantic Web and RDF

NDL is based on semantic web technologies, and in particular uses the resource descriptions framework (RDF) [10] to describe the schemas. For an explanation of RDF and its relation to the semantic web, see [11]. From the outset it was clear that an open interoperable format was required to create network descriptions. The RDF syntax suited our needs best, below we list four advantages of RDF over other formats:

Unique Identification Objects in RDF are identified by a URI. This is an advantage in multi-domain environments, since it makes it easy to clearly and uniquely define network elements in requests.

Flexible Graph Structure The relations between network elements can lead to cycles in the relation-graph (e.g. when combining multiplexing and inverse multiplexing adaptation functions). RDF extends the tree structure of XML with reference pointers so that it is able to deal with cycles.

Distributed Descriptions In order to describe inter-domain connections, the interrelation of different (administrative) network domains must be described. Each domain must be able to independently publish its own network information and point to other network domains. The RDF `seeAlso` predicate provides an elegant solution for this problem.

Extendable The network schema should be easily extendable. That is, allow the users to not only publish the information they care about, but also allow them to mix it with other schemas, both current (e.g. geographic information or organizational information in `geo` and `vcard`), but also future schemas, either direct extensions to NDL or non-directly related schemas.

2.2 NDL schemas

NDL classes and properties are organized in five modular schemas [12]:

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

The following three sections give a more thorough overview of the topology schema, the layer schema and the domain schema.

2.2.1 NDL topology schema

The classes and properties in the topology schema 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.

In figure 1 we see the topology classes and properties. A *Device* represents a physical or an abstract network element. In the topology schema we use the classes *Interface* and *Link* to create connections between devices. There are four properties to do this: *connectedTo* and *linkTo*, *switchedTo* and *packetSwitchedTo*.

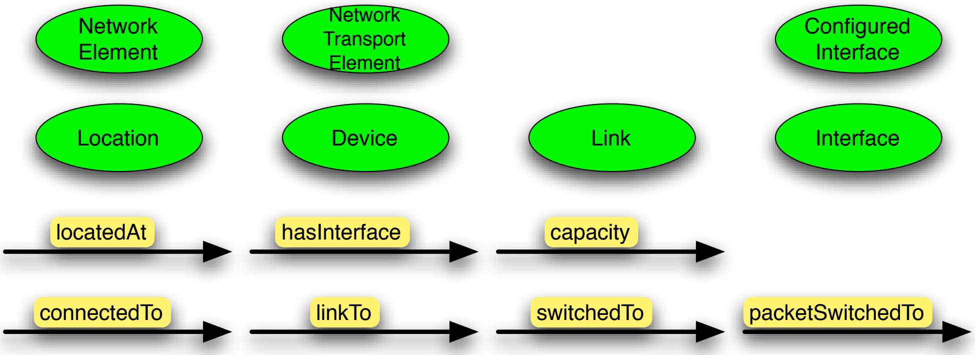


Figure 1. Classes and predicates in the NDL topology schema

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. A more extensive definition of the different classes and predicates can be found in the NDL schemata itself [12].

The current topology schema (part of NDL version 2) is based on the initial NDL schema (NDL version 1) we created. An example of a network description with NDL can be found in our earlier work [13].

The immediate applications of the topology schema are visualisation of network maps and input to path finding systems. All applications that are discussed in section 3 rely on the topology schema to describe and exchange data.

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 [14], which uses ITU-T G.805 functional elements [15] and the concept of labels as described in GMPLS [16].

Figure 2 shows the classes and properties in this 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*.

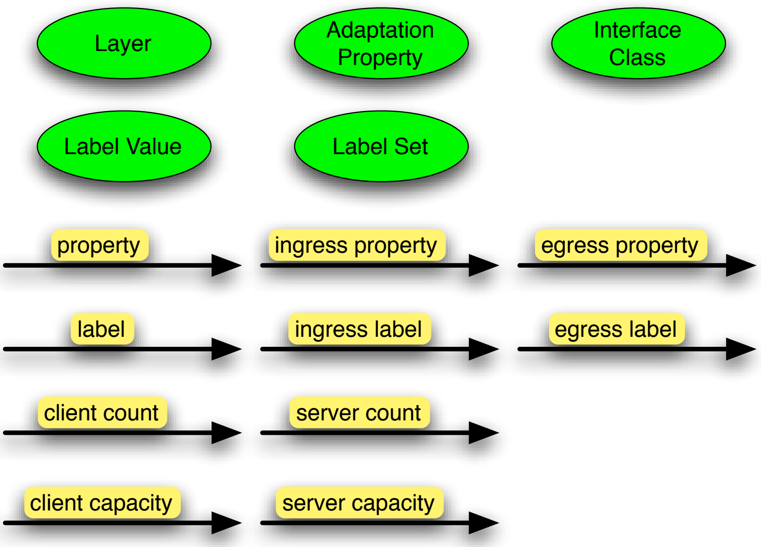


Figure 2. Classes and predicates in the NDL layer schema

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 defines administrative domains and the services offered by a domain. It allows network operators to provide an abstracted view of their domain to neighboring domain, rather than the full topology. Figure 3 shows the classes and properties in this schema.

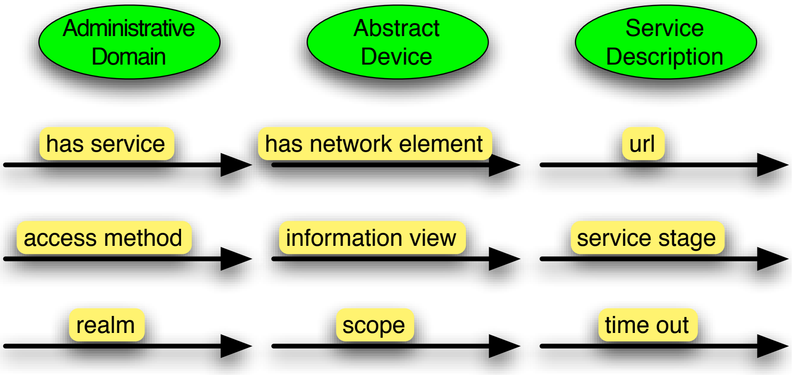


Figure 3. Classes and predicates in the NDL domain schema

The current schema only describes administrative domains, not owner domains [17]. This was done to keep the schema as simple as possible.

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.

We expect domains to publish static information in NDL, while providing a webservice for dynamic information or more confidential data, like reservation requests. Furthermore, we expect that different domains will have different opinions on what is “static” and “non-sensitive”.

The domain schema plays an important role in our future research, see section 4.

3 Existing Applications of NDL

NDL provides a powerful language to solve many of the operational issues that operators and users face in hybrid networks. It allows the automatic creation of network maps; it facilitates path finding algorithms used by reservation and network management systems; it enhances the interoperability and the exchange of information between different administrative domains.

In a previous publication [18] we showed the results of visualization tools based on NDL using GraphViz [19]; there we also described some applications under development. In this section we describe our progress on those applications.

3.1 Lightpath Planning in SURFnet6

SURFnet6 is the Dutch national research and education network. SURFnet6 is a hybrid network, offering both IP services and lightpath services. We have written a tool for planning new lightpaths based on NDL. In this application we use the TL1-Toolkit [20] to automatically generate an NDL topology description of the SURFnet6 network. Additionally, a network state database holds the cross-connect information for each network element in the network. That is, information about currently provisioned lightpaths. This enables the application to determine the amount of time-slots still available on each interface.

The user can use a web interface to query for a lightpath between two endpoints of the SURFnet6 network. The user first selects two endpoints from a list of the available endpoints. The next step is to specify some properties for this new lightpath, such as a name, the capacity and whether this should be a protected or unprotected path. Consequently, we use the information from the NDL file to construct a graph of the network. Using the network state database, we prune this graph by taking out sections that do not have enough bandwidth available. We then apply the Dijkstra algorithm using the current load of the network and metrics as constraints. If the user requests a protected path, we further prune the graph by taking out the network elements and interfaces used by the first path. In this modified graph the Dijkstra algorithm is run for a second time to find a backup path.

We are currently working on extending this application to support shared risk links groups in order to create better protected paths.

3.2 SuperComputing 2006 demonstration

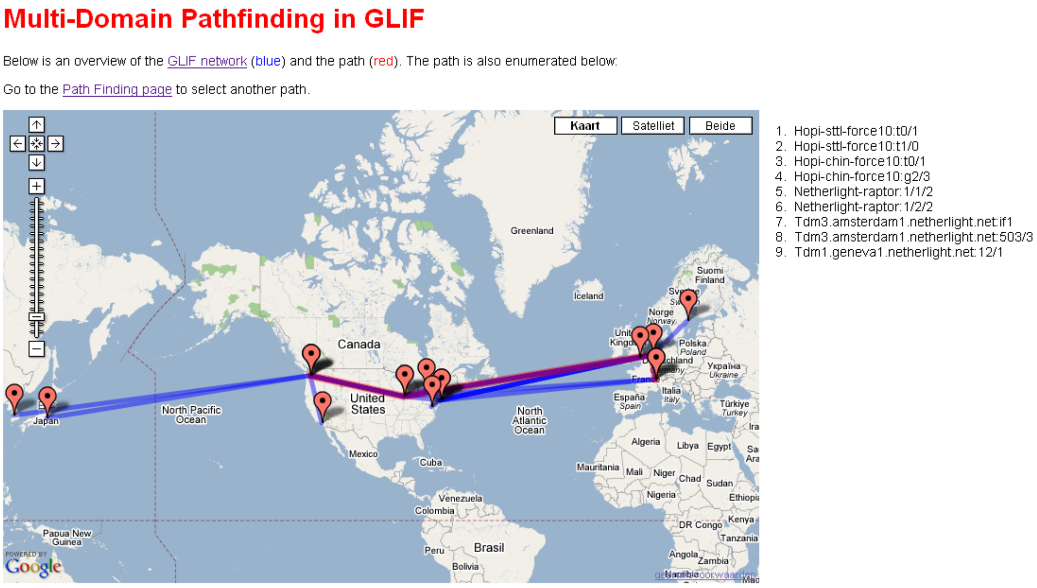
At SuperComputing 2006 we demonstrated path finding in the GLIF infrastructure [21] [1]. This infrastructure consists of interconnected GOLEs (GLIF Open Lightpath Exchanges). To set up a path through the GLIF infrastructure, you have to know which resources are available in each domain, and then find a path through the available resources.

For this demonstration most of the GOLEs provided a description of their network in NDL format. A key feature is that each description was published independently of the others, allowing GOLEs to stay in charge of the data they publish. The descriptions defined the physical resources in the GOLEs and the links to other GOLEs. To correlate the inter-domain connections, each description used the URI of the endpoint in the other GOLE. This URI, together with an RDF `seeAlso` property pointing to the description of the other GOLE. Using these links, a web of descriptions is formed, allowing applications to crawl all available information, getting a global view of the network.

During SuperComputing 2006 we demonstrated an application which gathered all the NDL files from the different GOLEs. Using a web interface, a user can select two endpoints from a list, which is generated from the gathered NDL information. After the two endpoints are selected, the application applied the Dijkstra algorithm to find the shortest path between the two endpoints. The resulting path is displayed in the web browser as a highlighted path through the network graphically presented using using Google Maps. A list of hops is also provided next to the map. Figure 4 shows the example output for a path between Seattle and Geneva.

With this demonstration we show how NDL can be used for inter-domain resource discovery and path finding. This application is mostly a proof of concept, because there are still more challenges with regard to inter-domain

[1] GLIF stands for Global Lambda Integrated Facility, which is a virtual organization of international research networks, research consortia and institutions.



Multi-Domain Pathfinding in GLIF

Below is an overview of the GLIF network (blue) and the path (red). The path is also enumerated below:

Go to the Path Finding page to select another path.

1. Hopi-sttl-force10:t0/1
2. Hopi-sttl-force10:t1/0
3. Hopi-chin-force10:t0/1
4. Hopi-chin-force10:g2/3
5. Netherlight-raptor:1/1/2
6. Netherlight-raptor:1/2/2
7. Tdm3.amsterdam1.netherlight.net:if1
8. Tdm3.amsterdam1.netherlight.net:503/3
9. Tdm1.geneva1.netherlight.net:12/1

Figure 4. Path finding in GLIF, presentented in Google Maps.

path finding. Issues such as policy and authorisation need to be addressed, as well as information regarding utilisation. The NDL domain schema can help here by providing pointers to relevant information services on policy, utilisation, or reservations. This is part of future implementations.

3.3 Lightpath Monitoring in NetherLight

NDL can play an important role in lightpath monitoring as well. We currently use NDL for lightpath monitoring in NetherLight, which is one of the larger GOLEs in the GLIF infrastructure.

To monitor the lightpaths, we use NDL to specify their topology details, and actively query the network elements involved. The output is stored in a network state database with alarm and configuration information. This enables us to correlate the configuration data with the alarm information and determine whether a specific lightpath is up or down. If a failure is detected somewhere in the lightpath route, this will be clearly indicated using a visualisation of the lightpath. This application is available online, see [22].

4 Future Directions

In this section we provide our ongoing research and future plans. In section 4.1 we present ideas on domain abstractions, followed by a discussion on policy and other path constraints in section 4.2. Then in section 4.3 we present ongoing work on improving lightpath monitoring across domain boundaries. Finally in section 4.4 we describe how we think NDL enables future architectures.

4.1 Domain Abstractions

In inter-domain scenarios it is often not desirable to exchange complete topology information. Domains often do not wish to disclose their internal topology, and it also does not scale to have a full view of the global optical network. NDL is currently capable of describing abstracted topologies, but these have to be constructed manually. The next natural step that we want to take is to make generation of abstracted topologies automatic. This will make it easier for domain administrators, so that they do not have to maintain two separate versions of the domain topology.

There are two strategies for doing domain abstractions [23], collapsing to a single point, or to edge points. The first abstraction method collapses the whole domain to a virtual domain node. The second method presents the domain as a collection of edge nodes, combined with the connectivity between them. The advantage of the first abstraction method is that it is simple to create. However, the disadvantage of this high abstraction level is that it is not possible to provide details of internal links, such as bandwidth, delay, or protection details. While the high abstraction level is simple to create, it is complex in use when mapping paths through the abstracted connection to actual paths through the network.

For both abstraction methods it is possible to include multi-layer information in the abstracted description. In the edge nodes configuration, each edge node maps directly to the multi-layer information of the actual edge node. If relevant, adaptations of the internal connection of the edge nodes can also be provided as properties of that interface.

In the single node configuration this becomes more complex. The single node is a virtual representation of the whole domain and does not correlate directly to an actual node. The adaptation capabilities of this virtual node then have to be a combination of the edge nodes. This can lead to complex situations if the adaptation capabilities of different edge nodes do not match.

Both abstraction strategies can be combined with a meta abstraction level as proposed in [6], where the abstraction of topologies is built using a hierarchy of domains. The abstraction is then performed in a tree like manner, where the top of the tree has a complete, but very abstracted view of the global topology.

To the best of our knowledge there has been no work on comparing the performance of different abstraction strategies, and it has not been used in practice yet.

4.2 Policy and Other Path Constraints

In scenarios with multiple domain connectivity, publishing a policy for inter-domain connections is inevitable. NDL can be used to provide details about the topology of the domain. Together with the topology it is also possible to include information about the policy of the domain. We believe that it is not feasible to include the policy itself, but rather that a pointer is given to a (web)service, where more information about the policy can be requested.

Note also that pathfinding in inter-domain scenarios is a multiple constraint problem; The following factors all play a role in determining the best possible connection:

topology Which paths there are between the two endpoints,
technology Which paths are possible, given the technological constraints,
policy The policies of *all* domains involved must be checked, and whether you agree with them,
availability Whether the user is entitled to use the path, and if so, whether the path is available at the requested time,
price The use of a section of the path will come with a price tag,
protection The required level of protection for the lightpath.

Multiple constraint based pathfinding is known to be NP-complete. There are several heuristic algorithms available [24], but these require information about the topology, and the constraints. Using NDL we aim to provide as much information as possible, so that based on the situation, heuristic algorithms can select the relevant constraints and perform pathfinding.

4.3 Lightpath Monitoring

In section 3.3 we have shown how we use NDL to visualize the topology of a lightpath correlated with its fault information. The same principle can also be applied to inter-domain lightpath monitoring. Currently, if an inter-domain lightpath goes down, it takes a long time to isolate the exact cause of the problem. The primary cause of this delay is that each NOC has to be contacted, requesting information about the relevant lightpath. Getting response from each NOC can take some time, because of miscommunication, there is no way to uniquely identify a lightpath. To make matters worse, inter-domain lightpaths usually span multiple time zones.

We are currently thinking about what is required to enable inter-domain monitoring. Great efforts have already been made to make NDL files available of all the individual GOLEs in GLIF. Using `seeAlso` pointers, it is possible to traverse the domains. If domains also provide references to services where information about alarms on links and interfaces can be requested, then distributed monitoring becomes a possibility. Each domain can use the linked NDL files to visualize a lightpath and query the webservices of each domain involved for alarm information. We are currently working to implement such a distributed monitoring system in GLIF.

4.4 Beyond lightpaths

An emerging application using inter-domain lightpaths is the distribution of high-quality digital cinema and video content. The CineGrid initiative aims:

> To build an interdisciplinary community that is focused on the research, development, and demonstration of networked collaborative tools to enable the production, use and exchange of very-high-quality digital media over photonic networks [25].

Lightpaths provide the appropriate level of service to efficiently deliver digital media, and as we illustrated in the preceding sections of this paper, NDL provides the network resources information basis for the provisioning of these connections. What we believe to be the next step is the definition of ontologies that together cover the whole end-to-end infrastructure: from the actual content being distributed to the Storage elements holding the data to the CPUs rendering the images and the (tiled) display or projector to visualize it. Figure 5 illustrates the concept of RDF representation of all these elements forming the overall infrastructure.

Our vision is that a media content locator will be able to consume the RDF descriptions of all the architectural components that form the end-to-end infrastructure. From this information it can build the optimal paths from the storage elements via the CPUs for transcoding to the visualization displays, making the inter-domain lightpath provisioning a piece in the overall orchestrated effort.

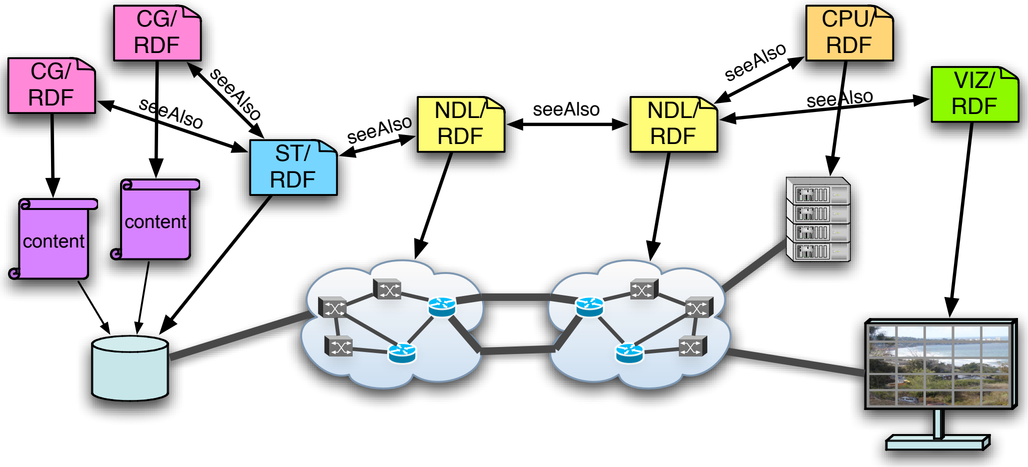


Figure 5. Using RDF to describe infrastructure

5 Conclusion

The management of hybrid networks and in particular of lightpaths is a complex task. While there are several systems for the provisioning of lightpaths intra-domain, these are currently not able to cooperate automatically to provision inter-domain lightpaths. The first step towards an interoperable multi-domain provisioning process is to create a standard vocabulary for the exchange of topology and requests.

In this paper we have introduced NDL, which aims to bridge the gap between provisioning systems. NDL provides a common vocabulary for exchange of information between providers, but also between users and providers. By utilizing the power of the Semantic Web, NDL allows providers to create a description of the global network, while each network remains responsible for his own topology description, by creating a distributed web of topology descriptions.

We have shown current applications of our data model and indicated our future research directions. In particular we believe that the definitions of abstracted network descriptions should provide enough information to facilitate the provisioning of lightpaths across domain boundaries. The open issue is what exactly 'enough' means in this context, and furthermore, how all this information can be used in inter-domain path computation. And once a lightpath has been provisioned, an interoperable inter-domain monitoring process will help to solve any problems as quickly as possible.

By using NDL, and thus RDF, future applications can more easily correlate information from different technology domains, allowing a new range of applications to use the optical network to its fullest extent.

Acknowledgments

This research was financially supported by SURFnet in the GigaPort-NG Research on Networks project. Part of this research is done under the Interactive Collaborative Information Systems (ICIS) project, supported by the Dutch Ministry of Economic Affairs, grant nr: BSIK03024.

The authors would also like to thank Franco Travostino from eBay (formerly with Nortel Networks) for sharing his vision of the Topology Knowledge Base [26], which initiated this research.

References

[1] J. Sobiesky, T. Lehman, Common service definition, Tech. rep., Mid-Atlantic Crossroads (MAX) (2005).
URL: `http://dragon.maxgigapop.net/twiki/bin/view/DRAGON/CommonServiceDefinition`

[2] Dynamic Resource Allocation Controller (DRAC).
URL: `http://www.nortel.com/drac/`

[3] User-Controlled LightPaths (UCLP).
URL: `http://www.canarie.ca/canet4/uclp/`

[4] A. Farrel, I. Bryskin, GMPLS Architecture and Applications, Morgan Kaufman, 2006.

[5] T. Lehman, J. Sobieski, B. Jabbari, DRAGON: A framework for service provisioning in heterogeneous grid networks, IEEE Communications Magazine 44 (3).

[6] A. Farrel, J.-P. Vasseur, A. Ayyangar, A Framework for Inter-Domain Multiprotocol Label Switching Traffic Engineering, RFC 4726 (Informational) (Nov. 2006).
URL: `http://www.ietf.org/rfc/rfc4726.txt`

[7] F. Aslam, Z. A. Uzmi, A. Farrel, Interdomain path computation: Challenges and solutions for label switched networks, IEEE Communications Magazine 45 (10).
URL: `http://dl.comsoc.org/cocoon/comsoc/servlets/GetPublication?id=11794458`

[8] M. Lazer, H.-M. Foisel, A. D'Alessandro, V. Shukla, M. Carroll, S. Okamoto, T. Tsuritani, R. G. andThierry Marcot, Optical network control plane carrier requirements for multi domain intelligent optical networks, Tech. rep., Optical Internetworking Forum (OIF) (2006).
URL: `http://www.olddog.co.uk/oif2006_134_02.doc`

[9] Network mark-up language working group (NML-WG).
URL: http://www.ogf.org/gf/group_info/view.php?group=nml-wg

[10] Resource Description Framework (RDF).
URL: http://www.w3.org/RDF/

[11] Resource description framework (RDF) primer (2004).
URL: http://www.w3.org/TR/rdf-concepts/

[12] Network description language homepage.
URL: http://www.science.uva.nl/research/sne/ndl/

[13] J. J. van der Ham, F. Dijkstra, F. Travostino, H. M. Andree, C. T. de Laat, Using RDF to describe networks, Future Generation Computer Systems 22 (8) (2006) 862–867.
URL: http://dx.doi.org/10.1016/j.future.2006.03.022

[14] 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, Computer Networks 52, (In Press).

[15] ITU-T Recommendation G.805: Generic functional architecture of transport networks, Tech. rep., International Telecommunication Union (March 2000).
URL: http://www.itu.int/rec/T-REC-G.805/en

[16] E. Mannie, Generalized Multi-Protocol Label Switching (GMPLS) Architecture, RFC 3945 (Proposed Standard) (Oct. 2004).
URL: http://www.ietf.org/rfc/rfc3945.txt

[17] F. Dijkstra, B. van Oudenaarde, B. Andree, L. Gommans, P. Grosso, J. van der Ham, K. Koymans, C. de Laat, A terminology for control models at optical exchanges, in: Adaptive Infrastructure, Management and Security (AIMS), 2007.
URL: http://staff.science.uva.nl/~fdijkstr/publications/controlmodels.pdf

[18] J. van der Ham, P. Grosso, R. van der Pol, A. Toonk, C. de Laat, Using the network description language in optical networks, in: Tenth IFIP/IEEE Symposium on Integrated Network Management, 2007.
URL: http://staff.science.uva.nl/~vdham/research/publications/0606-UsingNDLInOpticalNetworks.pdf

[19] Graphviz – graph visualization software.
URL: http://www.graphviz.org/

[20] SARA, TL1-toolkit.
URL: http://nrg.sara.nl/TL1-Toolkit/

[21] Global Lambda Integrated Facility (GLIF).
URL: http://www.glif.is/

[22] SARA, NetherLight lighpath status overview.
URL: http://noc.netherlight.net/cgi-bin/netherlight-status.pl

[23] T. Lehman, Personal Communication (2006).
URL: http://www.glif.is/meetings/2006/controlplane/lehman-gmpls.pdf

[24] F. Kuipers, P. Van Mieghem, T. Korkmaz, M. Krunz, An overview of constraint-based path selection algorithms for QoS routing, Communications Magazine, IEEE 40 (12) (2002) 50–55.
URL: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1106159

[25] CineGrid homepage.
URL: http://www.cinegrid.org/

[26] F. Travostino, Using the semantic web to automate the operation of a hybrid internetwork, in: GridNets conference proceedings, 2005.