The Network Markup Language (NML)

A Standardized Network Topology Abstraction for Inter-domain and Cross-layer Network Applications

Authors

Jeroen van der Ham <vdham@uva.nl>, University of Amsterdam
Freek Dijkstra <Freek.Dijkstra@sara.nl>, SARA
Roman Łapacz <romradz@man.poznan.pl>, PSNC
Aaron Brown <aaron@internet2.edu>, Internet2

Keywords

Network topology; network description; standard; multi-domain; multi-layer; topology exchange

Abstract

Introduction

Most network applications (provisioning, monitoring, visualisation) require knowledge of the network topology in one way or the other. Unfortunately, most applications only retain this knowledge within the application and do not share it with other applications. As a result, there can be conflicting topology instances, and it is hard to correlate data contained in these different applications. For example, if a user requests a dynamically-allocated circuit with one system, it is non-trivial to monitor this circuit in another system if the two systems describe the circuit in different ways.

In the last decade, research networks have started to open their network services to their users and to external parties. To interoperate, these control systems need to exchange topology descriptions in a standardized way. This is a change from the traditional exchange of network description as drawings by network engineers. A few network topology descriptions have been proposed (e.g. cNIS[CNIS], NDL[NDL], VxDL[VxDL] and perfSONAR[PS]), and their authors agreed to combine these efforts into a single network description standard under guidance of the Open Grid Forum (OGF)[OGF], which has become the Network Markup Language (NML)[NML].

Requirements

Initially the OGF NML Working Group set requirements for the terminology definitions for network elements, designing information structures and identifying the relations between them. The group derived from the experience with other topology description languages (e.g. ITU-T G.800 [ITU-

T.G.800], the perfSONAR schema [PS]) and the emerging needs of new advanced network services. The following is the set of requirements set for the NML specification:

- Network Infrastructure Agnostic – the NML schema must be not depend on specific network infrastructure.
- Encoding Agnostic – the NML schema must be able to be easily transformed into different wire encodings (e.g. XML, OWL[OWL], etc.).
- Extensible – the NML schema must be extensible to support new network infrastructure types as well as the needs of different applications.
- Concise – the NML abstraction should represent core network definitions (e.g. node, port, link) enough to model the basic network infrastructure. Complex structures should be provided as extensions.
- Scalable – NML must be able to deal with heterogeneous, dynamically growing networks.
- Multi-layer and multi-domain – NML must fit into the applications running in the multi-domain networks, aware of multi-layer structure.

These requirements make NML a good choice for applications such as network monitoring tools or network provisioning systems.

Schema Explanation

The NML Base schema describes an information model for computer networks. It is called Base because it contains only the basic topology definitions. In order to address the needs of certain implementations it may be extended to embrace layer-specific or technology-specific details.
The schema consists of classes, attributes, relations, and parameters. Classes represent types of objects, relations represent the relations between classes, attributes represent properties of classes. All those elements belonging to the Base are grouped in a single namespace: `http://schemas.ogf.org/nml/2012/10/base#`.

Figure 1 shows an overview of all the classes in the NML schema in a UML class diagram. There are four types of classes:

1. Abstract Classes (superclasses) with common attributes and parameters (Network Object, Service, Group),
2. Network Elements and Functionalities (Node, Port, Link, Switching Service, Adaptation Service, Deadaptation Service) that inherit from abstract classes,
3. Groups of elements and functionalities (Topology, Port Group, Link Group, Bidirectional Port, Bidirectional Link) that also inherit from abstract classes,
4. Supporting Classes that include additional set of information (Location, Lifetime), ordering feature (Ordered List) or distinguishing feature(Label, Label Group).

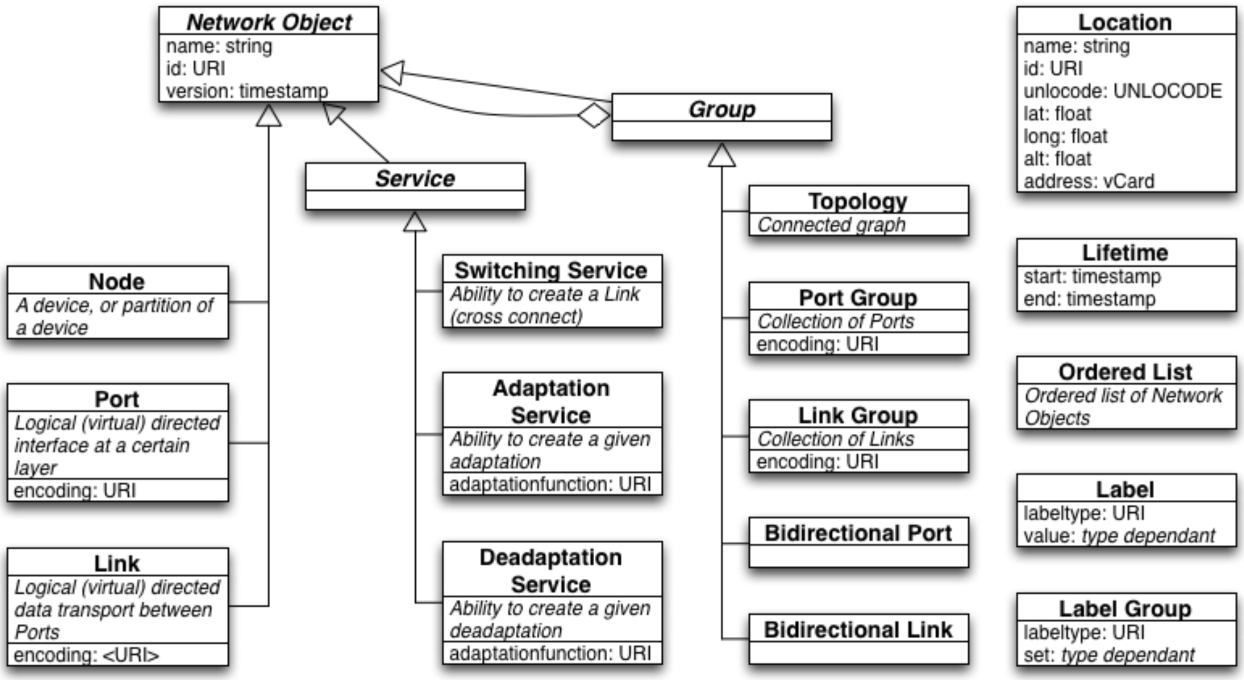

Figure 1 A UML class diagram of the classes in the NML schema and their hierarchy

NML objects may be related to other NML objects through relations. These relations allow for the description of the rich set of relationships that exist between network elements. The following list is an example of relations that have been defined for Node:

- existsDuring (to one or more Lifetimes)
- hasInboundPort (to one or more Ports or Port Groups)
- hasOutboundPort (to one or more Ports or Port Groups)
- hasService (to one or more Services)
- implementedBy (to one or more Nodes)
- isAlias (to one or more Nodes)
- locatedAt (to one Location)

All NML objects should have identifiers (id attributes), which must be a unique URI. Implementations that publish a network topology description with instance identifiers may adhere to the syntax of Global Network Identifiers as defined in [URN-OGF-NETWORK], which ensures global uniqueness and the easy recognition of Network Object instances. An example of an identifier applying to the defined rules is as follows:

```
urn:ogf:network:example.net:2012:local_string_1234
```

The Network Markup Language, currently, has two normative syntaxes. There is an XML syntax, defined using an XML Schema, and an OWL RDF/XML syntax, defined in an OWL schema. The OWL syntax is aimed at Semantic Web-oriented applications, while the XML syntax is suitable for any application.

The following two examples show a Node definition in each of the two formats, XML and OWL repectively.

```
<nml:Node id="urn:ogf:network:example.net:2012:nodeA">
  <nml:name>Node_A</nml:name>
  <nml:Location idRef="urn:ogf:network:example.net:2012:redcity"/>
  <nml:Relation type="http://schemas.ogf.org/nml/2012/10/base/hasOutboundPort">
    <nml:Port idRef="urn:ogf:network:example.net:2012:nodeA:port_X:out"/>
    <nml:Port idRef="urn:ogf:network:example.net:2012:nodeA:port_Y:out"/>
  </nml:Relation>
  <nml:Relation type="http://schemas.ogf.org/nml/2012/10/base/hasInboundPort">
    <nml:Port idRef="urn:ogf:network:example.net:2012:nodeA:port_X:in"/>
    <nml:Port idRef="urn:ogf:network:example.net:2012:nodeA:port_Y:in"/>
  </nml:Relation>
</nml:Node>
```

```
<nml:Node rdf:about="urn:ogf:network:example.net:2012:nodeA">
  <nml:name>Node_A</nml:name>
  <nml:locatedAt rdf:resource="urn:ogf:network:example.net:2012:redcity"/>
  <nml:hasOutboundPort rdf:resource="urn:ogf:network:example.net:2012:nodeA:port_X:out"/>
  <nml:hasOutboundPort rdf:resource="urn:ogf:network:example.net:2012:nodeA:port_Y:out"/>
  <nml:hasInboundPort rdf:resource="urn:ogf:network:example.net:2012:nodeA:port_X:in"/>
  <nml:hasInboundPort rdf:resource="urn:ogf:network:example.net:2012:nodeA:port_Y:in"/>
</nml:Node>
```

Standardization at the OGF

The Open Grid Forum (OGF) is a standardisation organisation with a standardisation process based on the IETF. The OGF was formed in 2006 from the merger of the Global Grid Forum and the Enterprise Grid Alliance. The Network Markup Language WG was initiated in 2007. The group published a context document [GFD.165] in 2010 and has been working on the schema document since. The current NML Schema document is in its final stage, and currently being prepared to go into the final public comments period in 2012, after which it can become a Proposed Recommendation early 2013.

A Use Case for NML – Supporting Network Provisioning

One important application of NML is supporting inter-domain circuit provisioning. The OGF' Network Services Interface (NSI) working group has standardized the Connection Services interface, which allows network provisioning software to interact inter-domain on lightpath requests. Version 1.0 of the NSI-CS was demonstrated at SC11, where an ad hoc topology representation mechanism was used. At SC12[SC12], NSI version 2.0 was demonstrated. As can be seen in Figure 2, the demonstration used several different implementations distributed over a dozen different networks globally.

The NSI 2.0 uses a topology representation in NML format. The NSI has extended this with extra information regarding the service plane information, for example the relation of a Network Service Agent and the network domain, and its URL.

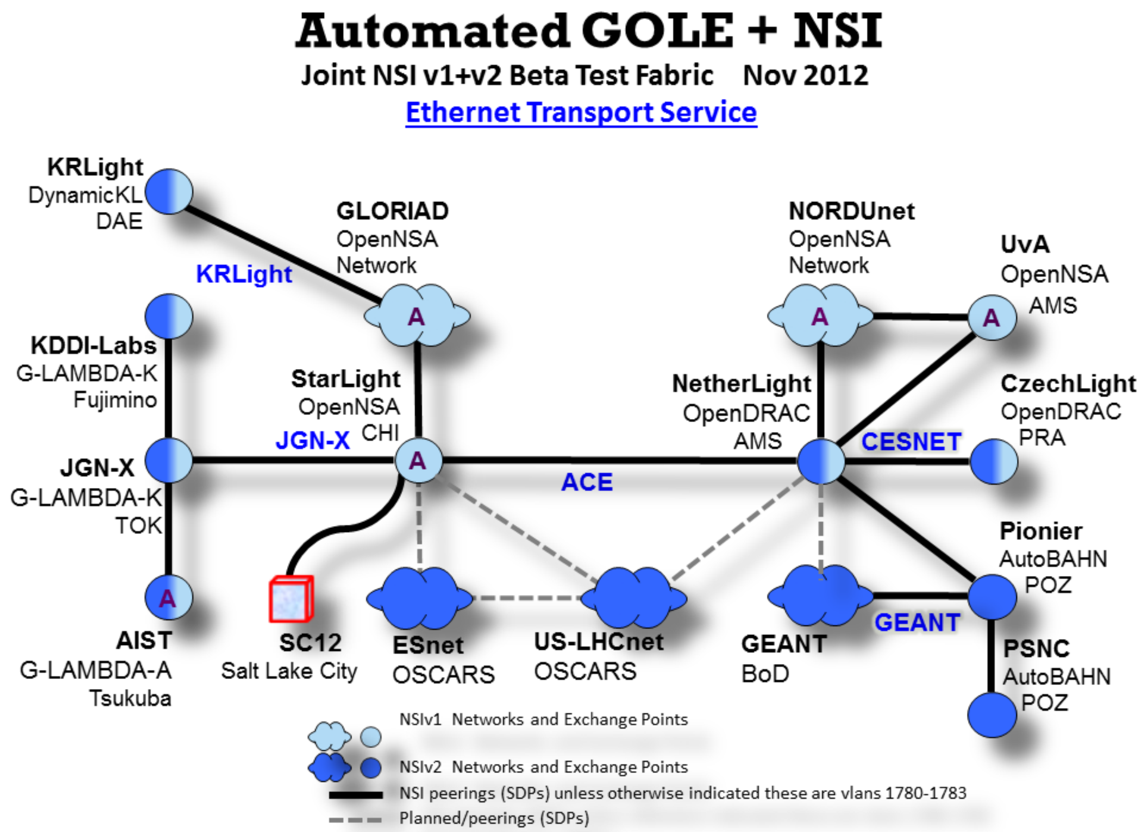

Figure 2. The topology used for the Automated GOLE NSI demonstration at SC12

Future Work

Version 1 of the NML base document is going through the last revisions as this is written. It is expected to go public comments and reach "recommendation proposed" standardisation status in the first quarter of 2013. Technology-specific extensions on Ethernet and DWDM are planned, as well as a document describing more experimental features. Further integration with NSI (for network provisioning) and PerfSONAR (for network monitoring is planned).

Acknowledgment

Parts of this work have been supported by the GN3 project and NOVI, both funded by the EC in the 7th framework programme, Grant Agreement No. 238875 and Grant No. 257867, SURFnet in the GigaPort-NG Research on Networks project, and the Dutch national program COMMIT.

References

[ITU-T.G.800] ITU-T Recommendation G.800: Unified functional architecture of transport networks, Tech. rep., International Telecommunication Union, Februari 2012, http://www.itu.int/rec/T-REC-G.800

[pS] perfSONAR web site, http://www.perfsonar.net/

[URN-OGF-NETWORK] A URN Namespace for Network Resources, F. Dijkstra, J. van der Ham, GWD-C (in public comments), https://forge.ogf.org/sf/go/doc16260

[GFD.165] Network Topology Descriptions in Hybrid Networks, P. Grosso, A. Brown, A. Cedeyn, F. Dijkstra, J. van der Ham, A. Patil, P. Primet, M. Swany, J. Zurawski, GFD.165, March 2010, http://www.ogf.org/documents/GFD.165.pdf

[CNIS] cNIS Database Documentation, https://forge.geant.net/forge/display/CNIS31DOC/Database+schema

[NDL] Network Description Language, F. Dijkstra and J. van der Ham, http://sne.science.uva.nl/ndl/

[VXDL] Lyatiss Resources, http://www.lyatiss.com/resources/

[OWL] Web Ontology Language, http://www.w3.org/TR/owl2-overview/

[OGF] Open Grid Forum, http://www.ogf.org/

[NML] Network Markup Language Working Group, http://www.ogf.org/gf/group_info/view.php?group=nml-wg

[SC12] SuperComputing 2012, Salt Lake City, http://sc12.supercomputing.org/

[NSI] Network Services Interface Working Group, http://www.ogf.org/gf/group_info/view.php?group=nsi-wg