Data Exchange between Network Monitoring Tools

Ronald van der Pol, Freek Dijkstra

SARA Computing and Network Services
Science Park 121, Amsterdam, The Netherlands
(rvdp|Freek.Dijkstra)@sara.nl

Abstract. Network engineers use different tools to manage and control networks. While different tools use overlapping datasets, the data exchange between tools never had much priority. The creation of lightpaths, dedicated network circuits, created the need for both multi-layer and multi-domain monitoring tools. Interdomain communication is still largely done by e-mail, but with the creation of multi-domain lightpaths, this is no longer feasible. Both standards and tools are required for the exchange of monitoring information between network domains. This article gives an overview of the current standards and tools that are under development. The Dutch supercomputing centre SARA is developing and deploying tools based on the network description language (NDL) and the network markup language (NML). The RDF-based NDL has proven to be a useful basis for an information model, although many architectural details still need to ironed out before the end-to-end monitoring of lightpaths can be fully automated.

1 Introduction

Customers demand reliable networks, so the adage of many network engineers is 'prevent, not react'. By keeping a close watch on the behaviour of their network, many problems can be spotted in advance, before they turn into an incident that affects the customer.

A typical network operations centre (NOC) is filled with displays were the engineers can scrutinise different aspects of their network. Fortunately, many tools exists to retrieve information from the network and display this information in a easily understandable form. Examples of these tools are Cacti, Nagios, NeDi, and ZenOSS [1,2,3,5]. Extensive lists of tools are maintained by Les Cottrell and Caida [11,18]. Most of these tools monitor uptime of hosts and applications, retrieve disk usage, CPU and network load and display the results in a web interface. This may be a good approach for monitoring services in a network, but does not work well for more advanced network monitoring.

Engineers in charge of monitoring large networks require in-depth analysis of their network. For example, a network engineer may need to know how and why traffic flows between two end-points, how this traffic changes if a link goes down and if such outage could cause congestion elsewhere in the network. Most

of the network monitoring tools operate at IP or application layer, but specific analysis of network behaviour may require information from lower layers in the OSI stack.

The need for complex network analysis has increased with the introduction of additional network services, such as lightpaths, dedicated network circuits for a single customer [12,28]. Most monitoring tools operate within a single domain, but lightpaths can cross multiple domains. In order to properly monitor and troubleshoot these circuits, network engineers require end-to-end monitoring tools. Some of the tools that are developed for such end-to-end monitoring include PerfSONAR, MonALISA and SpotLight [4,20,27].

The need for monitoring tools at lower layers or in multiple domains leads to an increase in the number of monitoring tools that are used at today's Network Operations Centres. Naturally, there is some overlap in the information that is required as input for these tools, for example the topology description. It is beneficial if the different tools use the same data format for common data, as that would make the deployment and integration of the different tools easier.

This article gives an overview of existing data formats in use for network status and perfrmance monitoring. Special emphasis is on the development of one of these data formats, the Network Markup Language (NML), and the experience with tool development.

2 Hybrid Networks

Many research networks have introduced hybrid networks in recent years. On these hybrid networks *lightpath* services are offered besides traditional internet services. Lightpaths are high speed (up to 10 Gbit/s) circuits with deterministic quality of service properties.

One of the motivations for lightpaths was the realisation that expensive routers were not needed when sending huge amounts of data between two fixed points in the network. A lightpath between the two points with fixed forwarding is a more economical solution.

Another example of lightpath usage is electronic Very Long Baseline Interferometry (e-VLBI), where several radio telescopes are connected to a centralised correlator. This setup creates one big virtual telescope of high resolution. However, the correlation process requires that there is little variance in delay. The deterministic behaviour of lightpaths guarantees this low variance in delay.

The introduction of hybrid networks by research networks has also introduced new challenges in network status and performance monitoring of these networks. Most research networks manage all layers in the network themselves these days. An integrated view from fibres up to IP routing and above is needed. Moreover, many of these lightpaths span multiple domains, which introduces additional challenges with respect to exchanging measurement and monitoring data between domains.

3 Data Formats and Protocols

This section gives a short overview of the most common data formats and protocols used by network management tools. The distinction between tools, data formats and protocols is blurry, especially for de-facto standard tools, which happen to use a certain format.

The overview follows the overall flow of data in monitoring frameworks, starting with measurements or retrieval of data from active network elements, to storage of the data. The next section will describe the usage and exchange of the data by and between tools.

3.1 Data Retrieval

Figure 1 shows how passive and/or active measurement points are placed in the network, which retrieve status and performance information from the network. This information is displayed in a comprehensive format to the user.

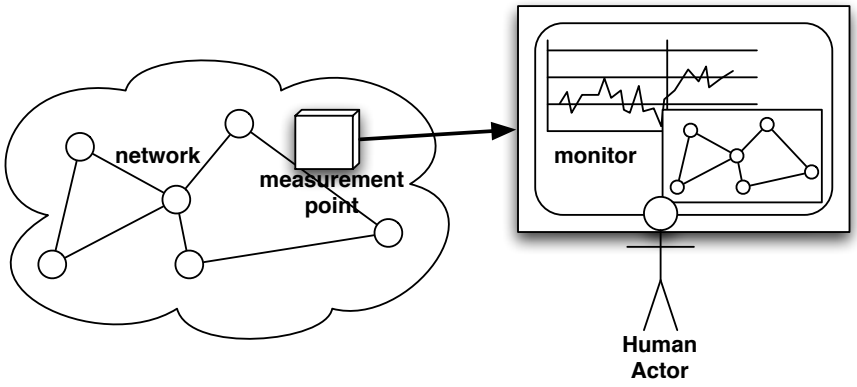

Fig. 1. A network monitor ideally absorbs all status and performance information from the network and displays it in a comprehensive format to the user.

The gathering of information can be passive or active. Passive gathering is done by retrieving information stored in network elements. Active gathering is done by sending data probes through the network, e.g. ICMP echo requests to retrieve information about RTT. Active state information can further be devided in intrusive and non-intrusive monitoring. The gathered information can contain topology information (for topology discovery), configuration information as well as volatile data (e.g. current bandwidth usage, CPU usage and and error counters).

SNMP, TL1, and CLI One of the most common protocols to retrieve information from a network device is SNMP. SNMP can also be used to configure network devices. The structure of SNMP is governed by Management Information Bases (MIBs) [29], which describe what each data entry represents.

Transaction Language 1 (TL1) is programmatic interface over TCP to retrieve data from device and configure devices. It is popular for optical devices and telecom equipment.

Ethernet switches and Routers are commonly administered through a command line interface (CLI). A CLI is not a programmatic interface, and is thus not an ideal way to use in conjunction with automated scripts.

NETCONF While SNMP is still in widespread use, the IETF considered it outdated. The NETCONF working group was chartered to create a replacement protocol, based on XML. This replacement is the NETCONF Configuration protocol [14].

Similar to how MIBs define the data structures for SNMP, data structure for NETCONF are defined using the Document Schema Definition Language (DSDL). The Netmod working group is chartered to draft the translation mechanism between DSDL schemas and NETCONF XML.

Polling Control Plane One method, which is particularly useful for automated topology discovery, it to poll control plane information. For example, an OSPF listener in a network will get a reasonable topology overview. For Generalised Multi-protocol Label Switching (GMPLS), the OSPF messages contain traffic engineering extensions, for additional information about layers and switching capabilities of devices [19,16].

Other protocols which lean itself for topology discovery are the spanning tree protocol for Ethernet and to a certain extend the ARP table for IP networks.

OSPF and Spanning tree distribute (some) topology information across a network. If these protocols are not used, it is possible to retrieve topology information using neighbour discovery protocols, such the Link Layer Discovery Protocol (LLDP) for Ethernet [6], Cisco Discovery Protocol (CDP), GMPLS' Link Management Protocol (LMP), or by using section traces in SDH.

Active Measurements The Internet Control Message Protocol (ICMP) is the foundation for active, but non-intrusive tools, such as ping and traceroute, on the IP layer.

The IEEE standardised IEEE 802.1ag [8], Connectivity Fault Management, in September 2007. This Ethernet extension provides features to detect link failures and ping and traceroute like capabilities at the Ethernet layer.

3.2 Data Formats

We have already mentioned MIB for SNMP and Netmod XML schemas for NETCONF. While both are primarily intended to specify the protocol, the data

structure can also be used as a basis for storing the data in a database. This section lists a few more data format examples.

Round Robin Database (RRD) The round robin database (RRD) is the defacto standard to store time-based data and is frequently used to store network performance data. RRD files contain a binary format developed by Tobi Oetiker for his Multi Router Traffic Grapher (MRTG), and is now in use by countless other tools [21,22]. The `rrdtool` command line tool allows easy parsing, storing and conversion of the data. `Rrdtool` can import and export data from and to XML format.

The main advantage of the RRD format is it's compactness, as well as the mature tools that are available to read and visualise the data.

RRD files only contain little meta data, such a short description what was measured, the duration and interval. It does not provide an ontology for meta data about the measurements to describe exactly what was measured where and how. Such meta data are required if different RRD data source are to be merged or automatic detection of anomalies is to be done.

Network Measurement Schemata The Network Measurement working group (NM-WG) in the Open Grid Foum (OGF) was chartered to identify network metrics that are useful to grid applications. In 2004, the group defined a nomenclature for network characteristics, distinguishing between actual, measured and perceived data [17]. A subsequent standard defined an XML schema for data and meta data for monitoring measurements [23]. This schema is used by existing perfSONAR tools.

Network Markup Language and predecessors Network topology information is used by multiple applications, like path finding and monitoring. For path finding this information is augmented with capability and current usage information, whereas monitoring is augmented with configuration information and status information.

The Network Markup Language working group (NML-WG) in the Open Grid Foum (OGF) standardises topology information. Preceding standards on which the NML build include the network description language (NDL) developed by the University of Amsterdam, and the common Network Information System (cNIS) developed by DANTE and PSNC [25,30].

These schemata have in common that they are intended to be used by a plethora of applications, and are therefor very generic. In particular, the schemata intend to be technology agnostic, while at the same time allowing applications to specify technology details in the schema.

Common Information Model The Common Information Model is an ongoing effort by the Distributed Management Task Force (DMTF) to define "management information for systems, networks, applications and services" [9]. Of

particular interest are the CIM network schemata [7], which includes configuration classes for Ethernet, MPLS and BGP up to the description of the physical dimensions of network equipment.

CIM is particular useful for data centres to describe access networks. It is less suitable for core networks. For example, it does not provide descriptions of SDH or WDM layers.

Custom Solutions In practice, most network operators will use a combination of readily available tools, proprietary tools provided by a vendor and custom made solutions.

Especially in research networks, the custom-made tools and data formats can be a large portion of the set of all tools.

4 Exchange of Data

The previous section gave an overview of data formats and protocols in use by monitoring tools. An astute reader will have observed that all protocols discussed so far cover the exchange of data from network devices to a tool, but not between tools.

This section will argue that there is a need for exchange of data between tools and domains, and discuss various approaches and functions.

4.1 Information Sharing Between Tools

So far, we treated the network as a closed entity, with a tool that retrieves state information from the network and displays this to the user. See figure 1. However, the state of a network does not give a full overview of all available information. A network state includes topology information, configuration information and capability information. Missing is information about future reservations, planned work, client data, known problems and incidents, et cetera. Such information is required for path finding and monitoring.

One option is to replicate the information for each tool (path finding tool, monitoring tool, incident tracking, etc.). The risk is of course that information in one tools gets outdated, so it is imperative that there is some sort update mechanism to exchange updated information between tools.

Ideally, there is only one authoritative data source for each type of data, and this data is distributed to other tools which require this data.

4.2 Configuration Database

A configuration management database (CMDB) is a database which contains state information about the network.

There are two ways to treat data in the CMDB: either the CMDB is authoritative, or the actually observed network configuration is authoritative. These

two views represent two distinct styles of network management. If the CMDB data is authoritative, the state is pushed to the network. In the actual network configuration is authoritative, the network state is pulled from the network and stored in the database.

An exponent of network push are programmable networks where not only the state but also the whole behaviour can be pushed to a network device.

A consequence of making the CMDB authoritative is that it demands a more advanced policy description in the CMDB, while the other way around results in more monitoring requirements.

4.3 Interdomain Information Exchange

We argued that information needs to be shared between tools within a single domain. Some information also needs to be shared between domains.

Lightpaths across domains pose additional challenges to path finding and monitoring. The current best practice in interdomain path finding and status monitoring is by using e-mail. Network engineers send mail describing topoplogy and status information to each other.

One of the problems with multi-domain monitoring of lightpaths is that failures in one domain will trigger alarms in other domains, while it is unclear in which domain the failure originated. End-to-end lightpath monitoring is required for engineers to pinpoint in which domain the real outage happened.

The same type of problems occur in path finding, where engineers need to know topology and capability information from other domains.

4.4 Existing Approaches

E-mail is still the prevalent method of information exchange between domains. Recently some domains augmented this by experimental monitoring services, such as a website with the current state information in a domain. While this approach still requires human interpretation of the data, it already speeds up the processes, as information is available 24x7 and engineers no longer need to wait for an e-mail response.

A next step in this process is the deployment of programmable interfaces. Webservices is a commonly used and suitable technology for this these interfaces. Before it is deployed, standard data models need to be developed for the information exchange. The Network Service Interface working group (NSI-WG) in the Open Grid Forum is chartered to develop such an interface.

Projects like Phosphorous [15] demonstrated the use of webservices for lightpath provisioning, and expressed interest in extending this to topology exchange webservices. So far, the Phosphorous architecture used a central topology database, but this will not scale. A decentral approach, such as deployed on the Internet is required here as well.

The Network Description Language (NDL) is based on the resource description framework (RDF). By using the `seeAlso` property, it is possible that each

domain publishes its own topology information, while the data sources are still linked together [26].

An unforeseen practical disadvantage of the use of RDF is that no practical implementations exist to combine RDF with webservices. A practical solution to this problem may be a generic network descriptions provided by the Network Markup Language workgroup (NML-WG), combined with more explicit service descriptions to point to neighbouring domains.

5 Tools at SARA

Currently, existing commercial and open source network management tools all have their restrictions. In order to investigate the status and performance monitoring challenges described in the previous sections several prototype tools were developed at SARA. These tools are used by the NOCs of SURFnet6 (the Dutch national research network) and NetherLight (SURFnet's optical exchange point in Amsterdam). The experiences of the NOCs with the tools is used to further enhance the functionality.

5.1 TL1 Toolkit

Many optical devices that are used in hybrid networks have little to no SNMP support. Instead they use TL1, Transaction Language 1. This is a CLI like interface to configure devices and to retrieve information from the devices. In order to make it easy to retrieve information from the network with scripts, the TL1 Toolkit [24] was developed. The TL1 Toolkit is a Perl module that can be used by Perl scripts. It offers the possibility to easily retrieve data from all devices in batch transfers.

The TL1 Toolkit is used to retrieve data of what is currently configured on the network and to retrieve operational status information from the network. All this data is stored in a database. Currently it is in the form of relational database tables, but work is going on to look at standardised configuration data formats, like NETMOD.

5.2 Topology Discovery

TL1 Toolkit scripts are used to automatically discover the topology of the network. This is done by retrieving section trace and adjacency information from all devices. This information is used to extract information about which device ports are connected to each other. The topology of the network is made available to tools by generating an NDL file. NML will be used as soon as it is standardised.

In SURFnet6 Nortel OME6500 SDH equipment is used. SDH has section trace information in the SDH headers. This 16 byte field is used to transmit a string to the adjacent device, which can check it against a configured string. We use this feature to decode port information about the adjacent device in the section trace. By doing this on all the links of SURFnet6 a list of connected

interface can be built. Combining all that information generates a complete topology of the network.

The DWDM layer of SURFnet6 consists of Nortel CPL equipment. On the ingress and egress of CPL add/drop interfaces, adjacency information can be configured. We use this feature to configure information about which OME6500 interface is connected to the CPL ingress/egress. This interface will use a particular colour on the CPL network. With the TL1 Toolkit we can retrieve the exact path of the colour through the CPL network. This returns all CPL devices in between. By doing this for all CPL paths information can be built about which OME6500 links are going through the same CPL links. These correspond to fibre ducts.

5.3 Path Finding

SURFnet6 is a hybrid network consisting of over two hundred Points Of Presence. Finding paths through the network for lightpaths cannot be done manually because of the large amount of data, dependencies and complexity, especially in the case of protected paths.

Therefore, a planning tool for SURFnet6 was developed. It uses the NDL file to get information about the topology of the network. The NDL file is converted to a mathematical graph so that shortest path algorithms can be used. In this case a constraint-based shortest path algorithm is used. The constraints that are used in SURFnet6 are the amount of free timeslots on a link, weights preferring cheaper links on the network instead of more expensive ones, shared risk link groups (e.g. prevent the primary and backup path from going through the same fibre duct), etc.

Besides the NDL topology information, the configuration data in the database is used. The configuration data is used to decide what part of the resources (e.g. timeslots) is already in use.

The Dijkstra shortest path algorithm is used to find unprotected paths in SURFnet6 [13]. The Suurballe algorithm is used to find protected paths in the network [10]. Work is going on to find protected path considering shared risk link group constraints. This is an NP-complete problem, but by trying to make use of the characteristics of SURFnet6 we hope to reduce the complexity of the algorithm enough to complete the path finding in reasonable time.

5.4 Future Work

Most of the tools that we developed and which are in production use are still mostly intra-domain, even though multi-domain usage have been demonstrated repeatedly. We found a clear distinction between what is technically feasible and organisational feasible. At this moment, most neighbouring domains value the automated information disclosure, but they only use the human readable part, while the programmatic interfaces are not yet used.

There clearly is an interest in a programmatic interface, as can be seen by initiatives such as the Network Service Interface (NSI-WG) in the Open Grid

Forum and the Generic Network Interface specifications (GNI) task force in the GLIF. However, the initial investment to use such an interface is still high while there are only few tools available.

Our goal is to simultaneously provide user interfaces and programmatic interfaces for our services, to at least make the investment for neighbouring domains as low as possible. At the same time, the value of standardisation is crucial, as commonly agreed protocols and data formats severely lowers the initial investment to deploy tools, and allow collaborative tool development.

We found that tools that aid rather than replace human procedures were more easily adapted by network engineers. A positive side effect of the tools was that it helped us shape existing procedures. For example, we found some internal inconsistencies in existing databases, and the tools allowed use to automatically detect this, and improve the source data. The quality of the source data proved to be crucial for the deployment of tools. However, since a large part of the source data is still manually generated (e.g. geographic markers in topology data, customer information and trouble tickets), it is imperative that the procedures to keep such information accurate are essential to tool deployment.

6 Conclusion

Network management systems and tools play an important role in network operations. There are many different programs and tools available, both commercial and open source. They all use their own data formats for storing configuration, topology and performance data. As a consequence there are many different tools for retrieving and storing data in various data formats. This makes exchanging information difficult.

This article explains why it is more efficient to have a few standardised data formats that can be used by different tools. Fortunately, some of these standardised formats are emerging, either as de-facto standards or as the results of formal standardisation bodies.

The tools that were built by SARA and that use these standards give useful insight in what data formats are needed and how to make the best use of them. By using the tools in daily operations useful feedback to the standardisation processes can be given.

References

1. Cacti. Available from: `http://www.cacti.net/`.
2. Nagios. Available from: `http://www.nagios.org/`.
3. Nedi. Available from: `http://www.nedi.ch/`.
4. perfSONAR. Available from: `http://www.perfsonar.net/`.
5. Zenoss. Available from: `http://www.zenoss.com/`.
6. Station and media access control connectivity discovery. IEEE Standard 802.1AB, IEEE, May 2005. Available from: `http://www.ieee802.org/1/pages/802.1ab.html`.

7. CIM network. Standard, Distributed Management Task Force (DMTF), August 2007. Available from: http://www.dmtf.org/standards/cim/cim_schema_v216/CIM_Network.pdf.
8. Connectivity fault management. IEEE Draft 802.1ag, IEEE, December 2007. Available from: http://www.ieee802.org/1/pages/802.1ag.html.
9. Common information model (CIM). Standard, Distributed Management Task Force (DMTF), November 2008. Available from: http://www.dmtf.org/standards/cim/.
10. Ramesh Bhandari. *Survivable Networks: Algorithms for Diverse Routing*. Kluwer Academic Publisher, 1999.
11. Les Cottrell. Network monitoring tools. Available from: http://www.slac.stanford.edu/xorg/nmtf/nmtf-tools.html.
12. Cees de Laat, Erik Radius, and Steven Wallace. The rationale of the current optical networking initiatives. *Future Generation Computer Systems*, 19(6):999–1008, August 2003. Available from: http://www.sciencedirect.com/science/article/B6V06-48V83MF-5/2/d8aac1d72ec497da8c83c4a07fdfec0c, doi:10.1016/S0167-739X(03)00077-3.
13. Edsger W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.
14. R. Enns. NETCONF Configuration Protocol. RFC 4741 (Proposed Standard), December 2006. Available from: http://www.ietf.org/rfc/rfc4741.txt.
15. Sergi Figuerola, Nicola Ciulli, Marc De Leenheer, Yuri Demchenko, Wolfgang Ziegler, and Artur Binczewski. Phosphorus: Single-step on-demand services across multi-domain networks for e-science. In *European Conference and Exhibition on Optical Communication (ECOC) 2007*, Berlin, Germany, September 2007. Available from: http://www.ist-phosphorus.eu/files/publications/PHOSPHORUS_Single-step_on-demand_services_across_APOC_07.pdf.
16. K. Kompella and Y. Rekhter. OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS). RFC 4203 (Proposed Standard), October 2005. Available from: http://www.ietf.org/rfc/rfc4203.txt.
17. Bruce Lowekamp, Brian Tierney, Les Cottrell, Richard Hughes-Jones, Thilo Kielmann, and Martin Swany. A hierarchy of network performance characteristics for grid applications and services. OGF Grid Final Documents 23, Open Grid Forum, May 2004. Available from: http://www.gridforum.org/documents/GFD.55.pdf.
18. Alex Ma. CAIDA tools. Available from: http://www.caida.org/tools/.
19. E. Mannie. Generalized Multi-Protocol Label Switching (GMPLS) Architecture. RFC 3945 (Proposed Standard), October 2004. Available from: http://www.ietf.org/rfc/rfc3945.txt.
20. Harvey B. Newman, I.C. Legrand, P.Galvez, R. Voicu, and C. Cirstoiu. MonALISA: A distributed monitoring service architecture. CHEP 2004, La Jola, CA, United States, March 2003. Available from: http://monalisa.caltech.edu/documentation/MOET001.pdf.
21. Tobias Oetiker. MRTG: The multi router traffic grapher. Available from: http://oss.oetiker.ch/mrtg/.
22. Tobias Oetiker. Projects using rrdtool. Available from: http://oss.oetiker.ch/rrdtool/rrdworld/.
23. Martin Swany. An extensible schema for network measurement and performance data. Draft, Open Grid Forum, February 2008. Available from: https://forge.gridforum.org/sf/go/doc15119.
24. Andree Toonk and Ronald van der Pol. Tl1 toolkit. Available from: http://nrg.sara.nl/TL1-Toolkit/.

25. Jeroen van der Ham and Freek Dijkstra. Network description language homepage. Available from: http://www.science.uva.nl/research/sne/ndl/.
26. Jeroen van der Ham, Freek Dijkstra, Paola Grosso, Ronald van der Pol, Andree Toonk, and Cees de Laat. A distributed topology information system for optical networks based on the semantic web. *Journal of Optical Switching and Networking*, 5(2-3):85–93, June 2008. Available from: http://staff.science.uva.nl/~vdham/research/publications/0703-ApplicationsOfNDL.pdf, doi:10.1016/j.osn.2008.01.006.
27. Ronald van der Pol and Andree Toonk. Lightpath planning and monitoring. In *eChallenges Conference 2007*, The Hague, The Netherlands, October 2007. Available from: http://nrg.sara.nl/publications/E-Challenges-v1.4.pdf.
28. Ronald van der Pol and Andree Toonk. Lightpath planning and monitoring in surfnet6 and netherlight. In *TERENA Network Conference 2007*, Lynby, Denmark, May 2007. Available from: http://nrg.sara.nl/publications/LightpathPlanningAndMonitoring.pdf.
29. S. Waldbusser. Remote Network Monitoring Management Information Base Version 2. RFC 4502 (Draft Standard), May 2006. Available from: http://www.ietf.org/rfc/rfc4502.txt.
30. Marcin Wolski, Stanislaw Osinski, Paweł Gruszczynski, Maciej Labedzki, Anand Patil, and Ian Thomson. common network information service schema specification. Deliverable DS3.13.1, GEANT, April 2007. Available from: http://www.geant2.net/upload/pdf/GN2-07-045v4-DS3-13-1_common_Network_Information_Service_Schema_Specification.pdf.