Handling Topology Updates in a Dynamic Tool for Support of Bandwidth on Demand Service

Christos Bouras, Ioannis Kalligeros and Kostas Stamos
Computer Technology Institute and Press "Diophantus"
Computer Engineering and Informatics Department, University of Patras
Patra, Greece
bouras@cti.gr, kallige@ceid.upatras.gr, stamos@cti.gr

***Abstract*—Automated Bandwidth Allocation across Heterogeneous Networks (AutoBAHN) is a tool under active development that supports a Bandwidth on Demand (BoD) service, intended to operate in a multi-domain environment using heterogeneous transmission technologies. The AutoBAHN system aims at providing a guaranteed capacity, connection-oriented service between two end points. Due to the distributed nature of the system and the fact that AutoBAHN has access to critical parts of the network, the importance of a robust, secure and dynamic mechanism for handling and distributing topology information cannot be underestimated. We will present how AutoBAHN manages to create an abstract representation of the physical network topology by hiding information such as IP addresses, port values, VLAN (Virtual Local Area Network), MTU (Maximum Transmission Unit), link capacities etc., how this abstract topology is merged among domains and how AutoBAHN system handles topology changes in terms of their influence on reservation processing.**

Keywords-Bandwidth on Demand; Topology Update; Topology Abstarction; Lookup Service; Quality of Service

I. INTRODUCTION

The GN3 European project is a research project funded by the European Union and Europe's National Research and Education Networks (NRENs). It is a continuation of the previous GN2 project and aims at building and supporting the next generation of the pan-European research and education network, which connects universities, institutions and other research and educational organizations around Europe and interconnects them to the rest of the Internet using high-speed backbone connections.

In the context of this project, a BoD service is being developed and the service is supported by the AutoBAHN tool. This BoD service is an end-to-end, point-to-point bidirectional connectivity service for data transport. It allows users to reserve bandwidth on demand between the participating end points. This service is offered collaboratively by GÉANT and a set of adjacent domains (NRENs or external partners) that adhere to the requirements of the service. These joint networks form a multi-domain area where the service is provided. The service offers a high security level in the sense that the carried traffic is isolated at the logical layer from other traffic.

The AutoBAHN system is capable of provisioning circuits in heterogeneous, multi-domain environments that constitute the European academic and research space and allows for both immediate and advanced circuit reservations. The overall architecture of the AutoBAHN system, its goal and the network mechanisms it employs are thoroughly presented in . AutoBAHN is responsible of managing the network elements in order to fulfill a user request. As a result, it needs to provide a mechanism for applying reservation actions on the necessary topology elements end-to-end. This paper highlights the structure of this mechanism and how AutoBAHN achieves, in spite its distributed nature, to have a total knowledge of the topologies of all NRENs that participate in this BoD service.

The rest of the paper is structured as follows: Section 2 presents the general architecture of the AutoBAHN system. In Section 3, we analyze how the topology handling mechanism works, while Section 4 presents in more detail how AutoBAHN processes topology changes and how they are propagated among the AutoBAHN instances. Sections 5 and 6 conclude the paper and present future fields of study.

II. AUTOBAHN ARCHITECTURE

The AutoBAHN system contains the Inter-Domain Manager (IDM), a module responsible for inter-domain operations of circuit reservations on behalf of a domain. This includes inter-domain communication, resource negotiations with adjacent domains, request handling and topology advertisements.

Furthermore, in order to build a real end-to-end circuit, the Domain Manager (DM) is another module that manages intra-domain resources. The IDM has an interface to the local DM (Idm2Dm), which undertakes all intra-domain functions (abstraction of the topology towards the IDM, scheduling and pre-reserving resources, monitoring etc.). This southbound interface of the IDM is the part of the AutoBAHN system that needs to be tailored to the domain-specific conditions.

In each domain, the data plane is controlled by the DM module using a range of techniques, including interfaces to the Network Management System (NMS), signaling protocols or direct communication to network elements. As part of AutoBAHN, a dedicated and independent Technology Proxy (TP) module allows the support of a range of technologies and vendors according to domain and global requirements.

 ISBN: 978-1-61208-279-0

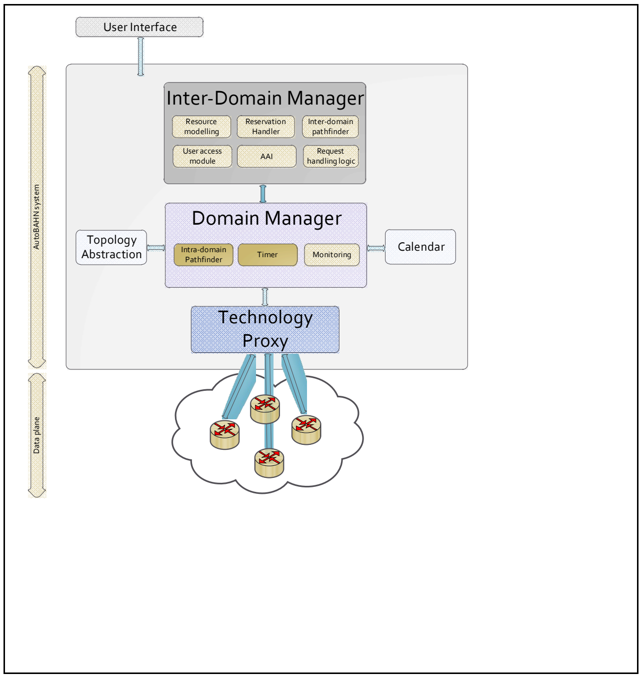

Figure 1. Basic architecture of AutoBAHN

The local NMS or service provisioning system, monitoring infrastructure, administration policies and security, may need to be adjusted for each networking domain making each TP implementation and configuration unique. However, the design of the DM has been optimized to support modular deployment and leverage the management infrastructure already deployed in any domain.

The above set of modules is deployed in each domain (NREN) that participates in the BoD service. A web based graphical environment (WebGUI) is used as a centralized portal for user access to the whole set of deployed instances.

III. Topology Handling

A. *Storage*

AutoBAHN uses cNIS , which is another service in the context of GN3 project in order to store and retrieve network topology elements.

The aim of the cNIS is to provide a unified repository for all relevant network information of a single administrative domain. cNIS was expected to be the "single point of storage", but in fact it is more than just a database. Apart from the internal functionality required for populating, validating and updating the database, it is equipped with modules for analyzing the topology data and presenting the data in a client-specified format (graphical, tabular or even XML (eXtensible Markup Language) for external applications). It can be used either as a component to build higher-level services and applications, or as a standalone repository of network topology data supporting network engineers in their daily administration work.

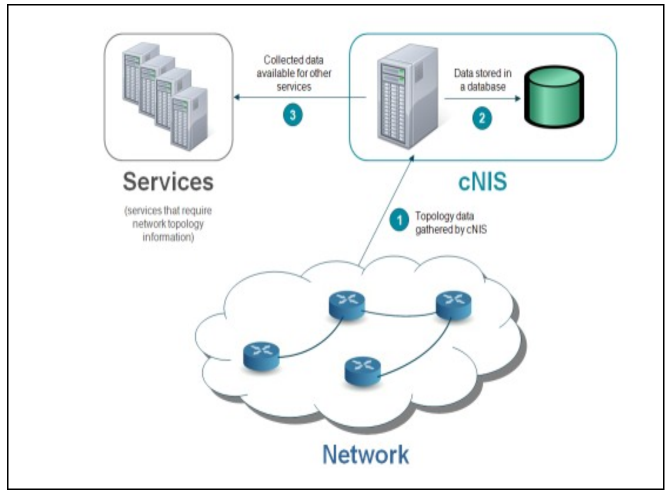

Figure 2. Basic usage of cNIS

As presented in Figure 2, cNIS implements three high-level functions, each covered by specific tools:

- Network topology data collection (automated or manual).
- Topology data management (keeping data consistent and up-to-date).
- Data sharing (interacting with external applications).

AutoBAHN retrieves the topology using a dedicated web service interface, stores it in its local database and creates an abstract representation which is later sent to Lookup Service (LS). LS is a common registry space for all AutoBAHN instances in order to exchange topology information. This procedure is thoroughly presented below.

B. *Topology Abstraction*

As it is already mentioned, topology elements and network resources are parts of the AutoBAHN system, dealing with reservation requests. There are three types of topology information:

- Physical network topology outside of the AutoBAHN system which is stored in cNIS.
- Technology-specific topology (DM level)
- Abstract topology (IDM level)

Due to the nature of the BoD service that AutoBAHN has to support, which is comprised of multiple heterogeneous and administratively separate domains, it is not possible, both for practical and for policy reasons, to handle global service topology in a flat manner and share topology information across the service. For example, one domain's administrators may not be willing to share details about their topology to everyone else participating in the BoD service. Therefore, detailed and technology-specific topology information is only kept per-domain, at the local cNIS instance. This topology is then abstracted before shared with the rest of the domains participating in the BoD service. This abstraction also enhances the heterogeneity of

 ISBN: 978-1-61208-279-0

the service since different network technologies can be represented in a unified manner and exchange information about the elements without taking into consideration how each network is implemented. Some of the supporting network technologies are Ethernet and SDH (Synchronous Digital Hierarchy) with the ability to easily extend current implementations to support also other technologies. The abstracted topology contains no technology-specific information such as actual interface names, VLANs, MTU sizes etc. It looks like a generic graph, uses ad hoc identifiers for network elements and it is shared among all domains. Each domain merges its own abstracted topology with the abstracted topology views advertised by the other domains, and this procedure converges with all domains having a shared view of the global abstracted topology.

The abstraction process that may be used by AutoBAHN is generally replaceable. The current abstraction algorithm creates one node for each edge node in the actual topology and then builds a virtual mesh network between them. The reason for choosing this algorithm is that it matches closely the 2-step way that AutoBAHN processes reservation requests: it deals with inter-domain links and domain edge nodes during the initial inter-domain pathfinding, and later each domain is responsible for finding a suitable intra-domain path. Inter-domain pathfinding operates on the globally shared abstracted topology, whereas intra-domain pathfinding is done separately by each domain and operates on the fully detailed local topology information.

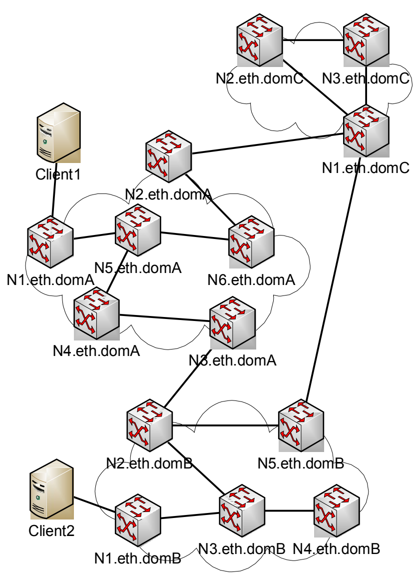

Figure 3. Technology-specific labels for topology components

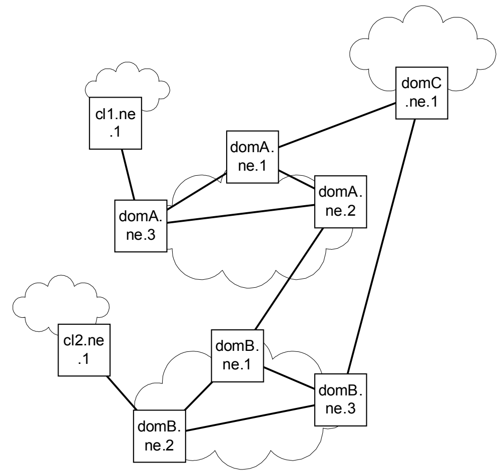

Figure 4. Abstracted topology view with global scope identifiers

Figure 3 shows a sample topology with domain-scope identifiers assigned to nodes and ports. The topology also contains information about interfaces, technical data including the half/full duplex attribute, MTU, capacity, and so on. Each domain only has local view of the technology-specific topology in Figure 3, for example domain C only knows about elements N1.eth.domC, N2.eth.domC, N3.eth.domC and their connections. The abstracted topology (that corresponds to the actual topology shown in Figure 3) is shown in Figure 4.

All domains have full knowledge of the global abstracted topology, so each domain is aware of Figure 4 in its entirety.

Abstract topology consists of domains, nodes, ports and links. Abstract identifiers for these elements are generated based on a 1-way hash of actual network elements in cases where corresponding actual network elements exist (for example for edge nodes) or artificial names in cases where corresponding actual network elements may not exist (for example for virtual mesh links in the abstracted topology).

The flow of the topology information within a single domain managed by AutoBAHN is the following; at the very beginning the domain's physical topology must be inserted into the technology-specific topology part by a system administrator. This topology includes a very detailed view of nodes, ports and links, also related to the resource usage and reservation mechanism. The DM operations are based on this technology specific topology view. However, as the IDM should not be aware of the technical details of the intra-domain topology and also the time issues within the domain, it must be provided with a less detailed, public

 ISBN: 978-1-61208-279-0

view, which can be also distributed to other domains. Once the DM has prepared the inter-domain abstract topology view, the latter is forwarded to the IDM module. It may be the whole topology during the system initialization phase, or it may be in the form of topology updates in the case of topology changes. In the second case, the IDM is responsible to forward changes that may affect one or more reservations. If a path is affected by a topology change (for example, by network failure), the reservation must be redirected to another path or must be aborted, depending on the resilience parameter specified by the reservation owner.

The IDM forwards the abstract topology view to the Lookup Service in order to have it advertised to AutoBAHN systems in neighboring domains. The advertisement mechanism also updates the local abstract inter-domain topology view. It contains not only the local view of the domain, but the entire inter-domain topology of all domains operating under a system, as it is retrieved by the LS. The abstract topology part of the system is the main source of information for the inter-domain pathfinding process.

C. Distribution

Since AutoBAHN is considered a distributed system because multiple instances are deployed across Europe, there is a need for those instances to be aware of the total network topology and not only the topology of the NREN that they actually belong. In order to fulfill this need, it was decided to have a centralized point of reference for all instances that could be used to exchange topology information and advertise their presence to other instances.

This is done with the utilization of Lookup Service which is also part of another GN3 project and more specifically perfSONAR . The Lookup Service acts as a service directory, where services (in our case AutoBAHN instances) can advertise themselves (provide their lookup information) and requestors are able to find any service they need. A more detailed view of Lookup Service's architecture and its purpose is described in .

Whenever an AutoBAHN instance starts, first of all, it writes an entry in LS with the following main attributes;

- Domain Name: unique identifier, e.g., GRNET, GEANT, PIONIER etc.
- URL: This is the web service endpoint of the domain's IDM interface through which other instances can use in order to communicate with one another

The second step is to query the LS to find what other instances have created a record. In that way, AutoBAHN knows what other instances are operating in the "neighborhood", stores this information in the local database and then communicates directly with them (if it is needed) through the interdomain (Idm2Idm) interface.

The procedure of adding new topology to LS is the following:

1. Check if LS contains existing abstract topology:

a. If yes, fetch it from LS and merge the elements that belong to other domains with the local elements. The final result first is stored to the local database and then is being written from scratch to LS.

b. If no, then the local domain's abstract topology is being sent to the LS for other domains to fetch it.

2. Update the timestamp of the entry.

IV. Topology Updates

As was described in the previous section, AutoBAHN creates an abstract representation of the topology. This is done by applying specific rules regarding the kind of the network element and a unique identifier that is created by information of this element such as IP address, name, domain where it belongs etc. The benefit of generating the identifier this way is that identifier generation is deterministic and constant among subsequent executions of the abstraction algorithm, even if the topology has changed significantly. In other words, the abstract identifier is kind of a signature for the network element that remains constant as long as the element is present in the topology.

For example, suppose that we have a network interface (port) with ID ath2.1-multi-10gbps that belongs to GRNET domain and a client (in other words an endpoint) that connects directly to this port. The hashing algorithm based on the id creates an 8 character long representation of this element which constitutes its abstract form. The final identifier becomes GRNET.pc.h438jd7t. The middle part is a categorization flag. In the specific example it means Port Client. AutoBAHN utilizes similar flags for other types of elements such as Port Interdomain (pi), Port Virtual (pv), Node Edge (ne) etc. In that way we have a quick solution for retrieving specific type elements from the abstract topology to process them. Furthermore, since this is a deterministic procedure, each time the algorithm is applied, the result is the same. This makes handling of topology updates and historical tracking of elements much simpler.was decided in order not to have database relations between actual network topology elements and abstract identifiers which would result in difficult handling in cases where after a topology update some elements could be removed by the topology. The latter is described in more detail below.

It is expected that during a topology update, AutoBAHN will have some reservations, either in an ongoing processing stage or at a processed (and finished) stage. This posed the problem of how to properly reflect the evolution of the topology, as network elements that may have been used or may be actively used by a reservation are modified or completely removed from subsequent instances of the domain topology. In order to handle this, we have chosen a different approach depending on whether a reservation is still being processed or whether it has finished processing.

Reservations that have been processed and they have terminated in some permanent state (e.g., finished successfully, failed permanently or were canceled) are transferred to "historical" status. Historical information is persisted in the AutoBAHN database separately from present topology and reservation information, and is therefore immune to any topology changes.

Reservations that are currently being processed may be affected by the topology update, if for example their chosen path is no longer available, or if a better option has been made available. In order to handle these cases, AutoBAHN uses the following approach:

1. Check if new network topology is available in cNIS.
2. If yes, freeze currently processed reservations (with all their state) in a different persistence location.
3. Clean the existing network topology information.
4. Import new topology information from cNIS.
5. Abstract the new topology.
6. Share and merge the updated abstract topology with all other domains through the LS.
7. Restore frozen reservations and try to map them to the new topology.

Restoration of frozen reservations differs depending on the way that they are affected by the topology change, Reservations that depended on elements that were removed are currently rejected, although it is planned to implement a restoration mechanism in this case. Also reservations that are not directly affected are restored and continue being processed without any change, even if the updated topology presents opportunities for different path selection.

V. Conclusion and Future Work

AutoBAHN is being developed towards the goal of a formidable tool for supporting a bandwidth on demand service. A very important part of its success depends on its capability to dynamically adapt to changes and in particular to topology updates, which are in practice frequent and sometimes wide-ranging, without negatively affecting service availability. In this paper we have discussed how this goal has been pursued until now.

Future work includes the design and implementation of the capability to restore reservations that are affected by a topology update and their admission decision is no longer valid (if for example some of the elements on their previously selected path are no longer available). Such restoration could be to recalculate a new path from scratch based on the updated topology. Right now AutoBAHN utilizes a fallback mechanism. If the topology update results in a reservation processing failure, it provides the administrator the ability to restore AutoBAHN in its previous state right before the update. So, the administrator can take proper steps to handle this case in a manual manner, e.g., to cancel existing active reservations from the system that will conclude in failure after the update.

References

[1] "GN3 European Project," [Online]. Available: http://www.geant.net/pages/home.aspx. [retrieved: January, 2013]

[2] Mauro Campanella, Radek Krzywania, Victor Reijs, Afrodite Sevasti, Kostas Stamos, Chrysostomos Tziouvaras, and Dave Wilson, "Bandwidth on Demand Services for European Research and Education Networks," in 1st IEEE International Workshop on Bandwidth on Demand, San Francisco (USA), 2006, pp. 65-72.

[3] "GN2 Project", [Online]. Available: http://www.geant2.net/ [retrieved: January, 2013]

[4] "TERENA", [Online]. Available: http://www.terena.org/ [retrieved: January, 2013]

[5] "cNIS (Common Network Information Service)", [Online]. Available: http://www.geant.net/service/cnis/pages/home.aspx [retrieved: January, 2013]

[6] "perfSONAR", [Online]. Available: http://www.perfsonar.net/ [retrieved: January, 2013]

[7] Jason Zurawski, Jeff Boote, Eric Boyd, Maciej Glowiak, Andreas Hanemann, Martin Swany, and Szymon Trocha, "Hierarchically Federated Registration and Lookup within the perfSONAR Framework", in Proceedings of the 2007 Integrated Management Symposium, 2007, IFIP/IEEE, Munchen, Germany, May, 2007