

Using Virtualization to Provide Interdomain QoS-enabled Routing

Fábio L. Verdi, Maurício F. Magalhães

Department of Computer Engineering and Industrial Automation (DCA)

School of Electrical and Computer Engineering (FEEC)

State University of Campinas (Unicamp)

Email: {verdi,mauricio}@dca.fee.unicamp.br

Edmundo Madeira

Institute of Computing (IC)

State University of Campinas (Unicamp)

Email: edmundo@ic.unicamp.br

Annikki Welin

Ericsson Research-Sweden

Email: annikki.welin@ericsson.com

Abstract—Today, the most important aspect related with the Internet architecture is its ossification representing the difficulties to introduce evolutions in the architecture as a way to meet the demands posed by the new requirements as mobility, security, heterogeneity, etc. In this paper we discuss how the network virtualization can be used to support the interdomain QoS-enabled routing. We present the Virtual Topology Service (VTS), a new approach to provide interdomain services taking into account QoS and Traffic Engineering (TE) constraints. We advocate in favor of a service layer that offers new mechanisms for interdomain routing without affecting the underlying Internet infrastructure. The VTS abstracts the physical network details of each Autonomous System (AS) and is totally integrated with BGP. Two models to obtain VTs were defined, the Push Model and the Pull Model. The latter one uses the Internet hierarchy to get more alternative routes towards a destination. We will show how the VTS and other services such as the end-to-end negotiation service work together to provide a complete mechanism for provisioning of interdomain QoS-enabled routes in IP networks. Preliminary evaluation results are also presented.

Index Terms—Network Virtualization, Interdomain Routing, QoS, Virtual Topologies, Web services.

I. INTRODUCTION

The most widespread form found in the literature to introduce evolutions in the current Internet has been through the use of overlay networks. Several proposals have been published presenting, in general, a specific solution for a specific problem. Basic questions concerning the overlay network approaches are, for example, node identification, routing, security, etc. More generic approaches have being

proposed based on the design of an infrastructure to support the creation of virtual or overlay networks.

Currently, there are two main views about the future of the Internet architecture: 1) a monolithic view based on a single protocol (IP) required in each network element; and 2) a pluralist view that considers IP as being only one component among others of an overall system called Internet. From this last view, the evolving architecture can be interpreted as a union of the various overlay networks and protocols. The central question is how to support the coexistence of a massive number of multiple overlay networks and what for. These systems have tools supporting the configuration of network resources as, for example, links, routers, etc. Several names have been adopted to this approach: meta-networks, virtual testbed [1], [2], Articulated Private Networks [3], etc.

A possible proposal is to consider a combined approach based on a pure architecture for the high-speed core and a more pluralist architecture closer to the edge [4]. We believe that the role played currently by the Internet Service Providers (ISPs) does not stimulate this hybrid approach because they serve two roles: managing their network infrastructure and providing (arguably limited) services to end users [5]. The coupling of both roles blocks the deployment of new protocols and architectures. We understand that in the future it will be necessary the presence of two separate types of providers: 1) the provider responsible by the network substrate, called Infrastructure Provider, that owns and maintains the network equipment (e.g., routers and links) and 2) the provider, called Service Provider, which establishes agreements with one or more infrastructure providers allowing the access and sharing of these router and link resources. The service provider is responsible by the protocol deployment and by the end-to-end services.

In this paper we present a starting point towards the network virtualization. Specifically, we discuss the Virtual Topology (VT) approach as a proposal for interdomain

This paper is based on "The Virtual Topology Service: A Mechanism for QoS-enabled Interdomain Routing" by F. L. Verdi, E. Madeira, M. Magalhães and A. Welin, which appeared in the 6th IEEE International Workshop on IP Operations & Management (IPOM 06), LNCS-Springer-Verlag, Vol. 4268, Dublin, Ireland, October 2006. © 2006 Springer.

This work was supported by Ericsson Brazil, Fapesp, CAPES and CNPq.

QoS-enabled routing. For a long time the networking research community has been trying to tackle with the interdomain provisioning of services. QoS is not considered in the original Internet architecture. Only a best-effort packet delivery service is available, but there is value in enhancing the network to meet application requirements [6]. While there are several solutions for TE and QoS within a single domain, the provisioning of services involving more than one domain is still a challenge. Currently, there is no way for end-users to choose the route in the domain-level. Today, the decision of which route the packets of a given flow will follow is taken basically by the BGP protocol that considers the business relationships between each pair of domain. Due to this scenario, the interdomain selection of routes does not take into account the diversity of paths among domains as a solution for load balancing and traffic engineering.

Giving more power of decision for customers will foster competition among Internet Service Providers (ISPs) imposing a different economic discipline and offering better services at lower prices for clients. It is a consensus that monopoly is not consumer-driven but provider-driven. If end-users can choose the domain route observing prices and the quality of the service, ISPs will have to face a competitive pressure to drive the deployment of good and new services to attract clients.

Although BGP is the current “de facto” interdomain routing protocol, it is becoming the main drawback for Internet Service Providers (ISPs). BGP advertises only reachability among domains by announcing network prefixes to its neighboring domains. While protecting internal details of domains is a requirement in commercial relationships, a certain degree of information could be opened without affecting the local strategy of each domain. Such information might include an abstract cost representing the current physical state of the network. Another known problem of BGP is related to slow convergence behavior. Interdomain routes can take up to 15 minutes to fail-over in the worst case [7]. This is not acceptable for mission-critical applications.

In this work we present a proposal for interdomain provisioning of QoS-enabled routing in IP networks. We assume that every single domain is capable of offering QoS towards some destination network prefixes. Each domain is responsible for implementing the network-enabled QoS by using, for example, DiffServ or Multiprotocol Label Switching (MPLS) technologies. Each domain is represented by a VT that gathers the traffic parameters to cross the domain in terms of bandwidth, latency, jitter, etc. The Virtual Topology Service (VTS) is responsible for getting the VT of each domain in a route in order to give the source domain more information related to QoS towards a given destination. The End-to-End Negotiation Service (E2ENS) will then negotiate the contracts with the chosen domains to reserve resources. The VTS implements two models for obtaining the virtual topologies from other domains. The first one is the Push Model in which every domain advertises its virtual topologies to

other domains. The second model is the Pull Model in which the virtual topologies are obtained on demand by the domains.

We elaborate our architecture to work in the management plane acting as a service layer for other domains and customers. This service layer offers specific services such as e2e interdomain provisioning of connections and interdomain provisioning of Virtual Private Networks (VPNs). The management plane abstracts the underlying details on how the provisioning of connections is performed by each network provider. This idea allows to have a service layer (a.k.a. service provider) over the network provider (infrastructure provider). In this work we propose to implement the service layer by using the Web services technology [8].

The architecture presented in this paper has already been used for provisioning of interdomain services in optical networks. In [10], [11] the architecture and the services are detailed. In [12] the evaluation of the architecture in terms of time and bandwidth consumption to establish interdomain optical connections was done. In this paper, we detail how the architecture can be easily used to provide interdomain routing with QoS in IP networks. The main purpose of this paper is to present and validate the integration between the service layer and BGP as well as to compare the two defined models (Push and Pull).

Our approach does not preclude the Internet as it is today neither does it exclude BGP policies that define the rules on how the network traffic must enter and exit in each domain. On the contrary, we propose a service layer that facilitates the interactions between providers by using Web services keeping all the legacy Internet infrastructure. Instead of competing with BGP, our architecture can be seen as a complementary tool for BGP running on the management plane.

This paper is organized as follows: next Section shortly presents some related work and their limitations for interdomain QoS routing. Section III details the VTS. Section IV is dedicated to show how the VTS was implemented and integrated with BGP routing. Finally, Section V concludes the paper.

II. RELATED WORK

The most recent approach related to interdomain QoS routing is the MESCAL approach [13]. The MESCAL project has defined the concept of local classes and extended classes. Extended classes are created by combining local classes with other extended classes from external domains. After defining and engineering the classes, the architecture has a function that advertises the QoS capabilities to customers and peers. The authors claim that a variety of advertising mechanisms can be used. However, they do not discuss how the classes will be announced. They assume an abstract relationship called *peering* that is general and implies the existence of some type of customer-provider interaction.

Although the project idea is very interesting, it depends on the extension of BGP, what, in our point of view, is a long term process of standardization without guarantees of becoming a standard. The authors advocate in favor of a QoS-enhanced BGP (q-bgp) that will be used to convey QoS-related information between ASes. Since the QoS information is exchanged by q-BGP, the route selection process needs to be modified to take into account new QoS attributes. This is, to the best of our knowledge, difficult to be put into practice due to the changing that will be necessary in every border router across the Internet. Are the companies, ISPs and vendors willing to change their interdomain process of route selection? These challenges have limited the solutions that depends on BGP extensions.

Among other works that propose to extend BGP, we can cite [14], [15] that suggest an additional attribute called QOS_NLRI as an extension of the BGP UPDATE message. The work presented in [15] has proposed statistical QoS metrics to achieve satisfactory routing optimality. However, none of these proposals was put into practice in real scenarios.

The Virtual Multi-Homing proposal (VMH) [16] discusses a new framework to achieve source-based path selection and improve interdomain routing. It is also a complementary approach to the current Internet routing. The main feature of VMH is to create an overlay network by which packets can be forwarded and the diversity of interdomain routes can be explored. Each domain that belongs to the overlay network has one or more Multi-Homing Servers (MHSs) that cooperate to each other to form a Multi-Homing Overlay Network. These MHSes are responsible for forwarding packets that are sent in the overlay network using IP Tunneling or similar methods. A new inter-AS relationship called Virtual Peering is proposed. It is a remote virtual connection between two remote ASes. A virtual BGP peering session is established between these two ASes. The solution can potentially improve the interdomain service quality by sending duplicate copies of packets using the overlay network and the physical connections (BGP routing) at the same time. When a packet arrives in a MHS, it verifies if the destination was reached. If the target domain is the local one, then the MHS sends the packet using the normal IP forwarding path. If the destination is not the local domain, then the local MHS sends the packet to the next MHS in the overlay interdomain route.

The idea of having an overlay network over the normal IP network is gaining attention and some proposals have used it to support interdomain routing. The VMH approach briefly described above uses the overlay as a solution to overcome BGP limitations. The approach discussed by the VMH model allows an MHS to contact a remote MHS using a virtual peering to send packets. Our idea is the same when a given domain, after choosing an interdomain route, contacts remote domains to negotiate the feasibility of the route. In our approach, the packets are sent over the virtual topology as if it was an overlay

network.

A very recent paper [17] has also used the *Pull Model* as a mechanism to obtain extra BGP routes besides the default ones. Such work also defines a bilateral negotiation between ASes in order to negotiate alternative routes. The mentioned work presents similarities with the VT approach discussed in this paper, however, it does not focus in QoS neither does it mention virtual topologies. Its focus is on finding different interdomain paths.

Although there are many proposals for interdomain QoS routing, there is nothing related to the use of virtual topology in this type of scenario. In our point of view, the virtual topology approach outlines novel models on how the interaction between domains will be performed.

III. THE VIRTUAL TOPOLOGY SERVICE

The Virtual Topology (VT) concept represents the QoS features of each domain towards a destination domain. A given domain may have several different VTs that are advertised following specific rules such as the variation of the amount of traffic during the day, services being offered (VoIP, video conference, VPNs being established and so forth) and the availability of the resources in terms of bandwidth, latency, jitter and packet loss rate. The VT is formed by a set of virtual links that map the current link state of the domain without showing internal details of the physical network topology. Fig. 1 illustrates the domain 1's VT and its QoS values towards the downstream domain 2.

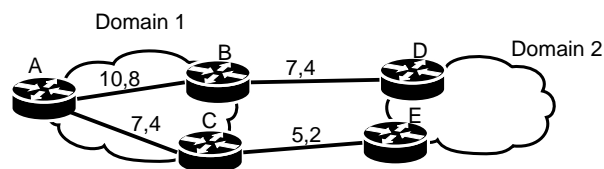


Figure 1. The Virtual Topology concept.

In this example, domain 1 has two egress routers towards domain 2. Each internal virtual link as well as each external virtual link can be implemented in different ways. One egress router can be a DiffServ router whilst the other can be an MPLS router. Each virtual link gathers QoS traffic parameters that reflect the current state of the domain. In Fig. 1, the virtual link A-B has, for example, 10 Mbps of available bandwidth and a latency of 8 ms. The virtual link A-C has 7 Mbps of available bandwidth and a latency of 4 ms. Then, going through the virtual link A-B to reach domain 2, the available bandwidth would be 7 Mbps (the lowest value is used) having a latency of 12 ms. If the virtual link A-C is chosen, the available bandwidth would be 5 Mbps having a latency of 6 ms. If Domain 1 advertises its virtual topology to Domain 2 at this moment, Domain 2 would see the virtual topology and the values as shown in Fig. 1.

The way the IP traffic will be carried in each domain depends solely on the engineering policies deployed within the domain. In case of hard QoS guarantees, specific MPLS LSP tunnels could be used within each

domain to force the reservation of the resources otherwise DiffServ with TE functions should ensure the QoS required in the contracts. Other wrapping and tunneling solutions, as mentioned in [17], could also be utilized.

The VT approach also allows a domain to balance the load among the egress routers towards the destination prefixes. Since the VT reflects the current state of the network capabilities, virtual links that are overloaded will be avoided by the source domain during path calculation. Below, we list some advantages of the VT approach.

- **Policy-based configuration of VTs:** Each domain can setup VTs by following local policies. These policies represent business interests and commercial relationships between domains. Also, VTs can be created taking into account specific rules. The domain administrator can define a “standard” VT to be used in normal conditions and other VTs that are used when specific conditions are detected;
- **Promotional Offering of VTs:** Domains can offer promotional VTs in a specific day of the week, for example. Administrators can setup their VTs in such a way that the idle links are used by a low price. By doing this, domains can attract more customers to use their connections then increasing the domain’s revenue;
- **Specific VTs for specific Applications:** A given domain can offer VTs that fit into customers requirements. For instance, if a customer needs to have full backup of its connections, then it can contract the virtual topology service to advertise VTs whose virtual links have some degree of protection. This type of scenario is common when very important transactions are performed such as bank transactions or any other type of financial transaction;
- **Reservation of Resources:** Resources in each virtual link can be reserved for further use. This mechanism is used to guarantee that when the resources are really needed they will be available. A typical scenario for using this is the establishment of a VPN that needs to reserve resources for being used when the VPN is set up;
- **Rapid Re-advertisement of new VTs:** As the resources of each virtual link are consumed, the domain can re-advertise other VTs in such a way that the physical resources of other routes are used. This can be done by creating new tunnels (e.g. using MPLS) forcing the usage of idle resources. Also, as time passes, a domain can grow and change its relationship with its neighboring domains. Then, the VTs can be advertised with more information depending on the new established relationship;
- **Routing is done over the VTs:** When a given domain advertises its VTs to other domains, it means that such domain is accepting to receive connections over those VTs instead of following only BGP policies. During the negotiation phase, the domains can negotiate specific attributes of connections;
- **VTs are seen as commodities:** The VTs are no longer

seen only as a set of nodes and links. They are seen as commodities that can be traded using commercial interests to motivate the “fight” for customers. Domains need to engineer their network in such a way that the VTs efficiently map the physical resources. The administrators can use complex off line algorithms to find the best set of VTs that should be advertised. This estimation could be done by using a traffic matrix and the forecasting of income and outcome flows;

- **E2E Interdomain QoS routing is done over the VTs:** While BGP does not have any type of TE metric, the VTs advertising can include information concerned to several types of TE metrics. The level and quantity of information that each domain advertises to each other are a local decision based on the relationships between them;
- **Multi-homed customers can decide which provider to use:** Based on the requirements of a flow, prices and availability of resources, customers are able to choose the best route option that fits their interests.

A. How VTs are Obtained

We have defined two models to obtain VTs: the push model and the pull model. The latter is also known as on demand model.

1) *The Push Model:* In this model, the VT advertising is done between pairs of domains. Each domain announces the VT to all its neighbors respecting commercial and economic relationships previously defined. The push model is more indicated to a regional scenario. We envisage that this regional scenario is formed by “condominiums of domains” by which a group of domains agrees on advertising virtual topologies to each other. This advertising is done in a peering model where all the domains that make part of the same condominium have the virtual topologies of other domains. These condominiums of domains could define business rules in a tentative of creating new relationships that make the interactions more customer-oriented. Fig. 2 shows a scenario with three domains advertising their VTs to each other.

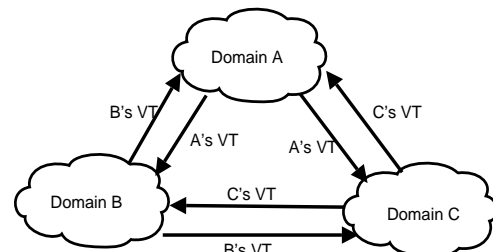


Figure 2. Advertising virtual topologies. All the domains have all the VTs (Push Model).

2) *The Pull Model:* In the pull model, domains do not advertise their topologies to the neighboring domains. The VT is obtained by each domain that wants to know the VTs of other domains. When a given AS needs to

find an e2e QoS-enabled interdomain route, it queries its BGP local table and verifies what are the possible routes to reach the destination. Then, based on these routes, the source domain can invoke each domain in the route towards the destination and gets the VT of the domains specifically for that route. Fig. 3 shows an example.

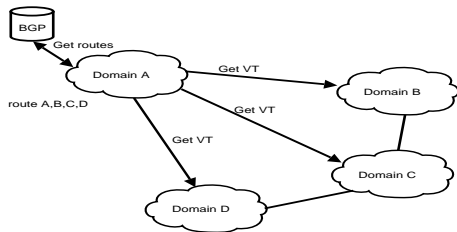


Figure 3. Getting VTs (Pull Model).

Suppose that domain A needs to reach a destination located at domain D with a certain QoS. Domain A queries its local BGP table and discovers that the best route based on BGP is through domains B, C and D. Then, domain A invokes the VTS to obtain the virtual topology of each domain. Unlike the MESCAL approach which adopts a cascaded model to perform the interactions among domains, the VTS adopts a centralized mechanism by which all the VTs are obtained in parallel. Note that there can be more than one BGP path to the destination D. As a result, the source domain can recursively query each domain in each path and find the best path towards the destination.

After obtaining all the VTs of each possible route towards domain D, the source domain A is able to use a Constraint Shortest Path (CSP) algorithm to find the best route that fits the QoS requirements. The path calculation can be done using only one attribute or more than one. For instance, if a given domain requires low latency, it can use only the latency attribute. If it desires lowering packets loss rate, it can use only the packet loss rate attribute.

However, after obtaining the VTs of each route towards a destination, the source domain can realize that there is no route that satisfies the QoS requirements. Then, as mentioned before, our architecture makes use of the Internet hierarchy to collect more alternative routes towards a given domain. It does so by going one-level uphill in the Internet hierarchy to get other not-advertised BGP routes. This is explained below.

Although very difficult to define, the Internet hierarchy is divided into 5 layers [18]: the dense core(0) with about 20 ASes, transit core (1) with about 129 ASes, outer core (2) with 897 ASes, small region ISPs (3) with about 971 ASes and the customers (4) or stubs ASes with about 8898 ASes. Based on previous studies, it was verified that the quantity of possible different paths towards a given destination increases when going uphill in the Internet hierarchy [18]. As an example, there are 2409 edges from level 3 to level 4 and 3752 from level 2 to level 3. It has been also shown that there are about 193,000 different paths from any customer AS to the dense core [7]. However, BGP only advertises the best path and

then, the quantity of possible paths depends on the multi-homing aspect of each domain. Fig. 4 illustrates this scenario.

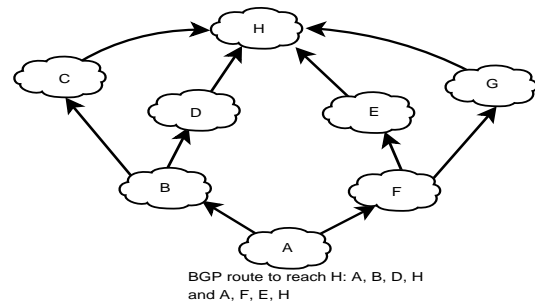


Figure 4. Getting more paths (Pull Model).

The stub domain A is multi-homed with domains B and F. It has received two BGP routes from its providers to reach prefixes at domain H. The first route is A, B, D, H and the second one is A, F, E, H. However, there are other possibilities to reach H through domains C and G that were not advertised to domain A. Then, to increase the number of paths to query for VTs, the VTS can invoke its providers and asks the other BGP routes that were not advertised. In this case, domain B would return to domain A the path C, H towards H and domain F would return G, H towards H. As a consequence, domain A would have now other two different paths to reach H and then can ask each domain to get the VTs.

Due to the current characteristics of the Internet, going uphill only one level from a given domain is enough to have a high number of possible different paths. As mentioned in [18], as we move from customers to the core, the inter-level connectivity raises significantly. At the same time, not all BGP routes need to be returned to the source domain. In a real Internet scenario with thousands of routes, only some of them should be selected based on local policies defined by the provider.

3) *Brief Comparison between the Models:* The advantages of the push model are:

- When the source domain needs to find a path, the VTs will be already available in the local domain since they were previously advertised. It is not necessary to get the VTs at the time of path calculation;
- Each domain can advertise VTs to the neighboring domains as desired. This mechanism allows to offer promotional VTs with specific characteristics and call the attention of other domains. It is more business-oriented;
- More dynamic. The VTs are immediately advertised in case of any local domain event, e.g., a failure in a link/router.

The disadvantages of the push model are:

- It is more complex. The advertising mechanism should respect the policies of each domain relationship;
- It generates more traffic. VTs are advertised as desired by each domain. However, it does not mean that such VTs will be used by other domains.

The advantages of the pull model are:

- It is simpler than the push model. The pull model will query each domain of a given path to get the VTs. It is not necessary to have an advertising mechanism;
- There are no scalability issues. Only the VTs of the chosen routes will be obtained. Based on previous studies [19], the mean e2e communication in the Internet traverses between 3 and 4 domains. Then, the quantity of domains to be invoked to obtain the VTs is very small.

The disadvantages of the pull model are:

- It takes more time to find a path because it needs to obtain the VTs during the path calculation;
- It is less business-oriented since the domains cannot advertise promotional VTs to call for connections;
- It is less dynamic. A domain cannot replace and advertise its VT if a given event happens. However, this problem could be solved by having a notification message to allow a domain to notify other domains about local events. Then the source domain can invoke the VTS to obtain a new virtual topology.

B. Detailing the Architecture

The architecture being proposed in this work offers a service layer by which QoS routing can be obtained among IP domains without having to change or extend the BGP routing. The architecture and the interactions between the modules are shown in Fig. 5. The service layer is formed by the Virtual Topology Service (VTS) and by the End-to-End Negotiation Service (E2ENS). The VTS is responsible for interacting with other domains to obtain the virtual topologies. The E2ENS is responsible for doing the negotiation among domains in order to establish e2e interdomain contracts. Since the focus of this paper is on the VTS, specific details of the E2ENS and the way these contracts are established in terms of format are not defined here and are left for further studies.

During the negotiation, the required QoS parameters are transferred from the head-end domain¹ to other domains in order to negotiate and establish a SLA between domains. We adopted a two-phase-star-based model by which the head-end domain negotiates with other domains to define a contract. The first phase queries the downstream domains about the possibility of reserving resources (basically bandwidth) for a given IP flow. During the first phase, the traffic parameters are analyzed in each downstream domain in order to verify if the IP flow can be accepted. The second phase confirms the contract with each domain considering that all the domains involved in the negotiation have agreed in receiving the IP flow.

In the first phase, the DiffServ Code Point (DSCP) or the MPLS label of the downstream domains should be returned to the head-end domain to configure the egress routers (in the second phase) of the upstream domains to swap data packets. This is necessary to mark the

¹The head-end domain is the domain where the request was made.

packets in order for them to be identified by the adjacent downstream domain as belonging to a specific QoS class.

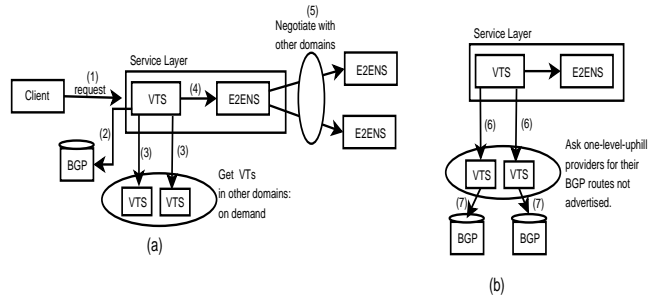


Figure 5. Service Layer Architecture and the Interactions between the modules.

Our approach works as follows. First of all, the source domain tries to attend the customer request by going through the local BGP routes towards the destination (Fig. 5 (a)). If such routes do not satisfy the customer requirements in terms of QoS, then the source domain can ask its providers for other BGP routes that were not advertised (Fig. 5 (b)). When a request coming from a client is done (step 1 in Fig. 5 (a)), the service layer validates such request and reads the BGP routing table (step 2 in Fig. 5 (a)) to obtain the routes available to reach the destination required by the client. After obtaining the BGP routes, the VTS is invoked to get the VTs of each domain belonging to the available routes (step 3 in Fig. 5 (a)). When all the VTs are gathered, the source domain will have a topological view in terms of QoS routing towards the destination. The source domain can then apply a CSPF algorithm and find the shortest path that satisfies the QoS requirements. After having calculated the interdomain path, the E2ENS will be invoked to perform the negotiation of the traffic parameters between the domains and possibly establish a SLA (steps 4 and 5 in Fig. 5 (a)). If after negotiating with other domains, the local domain concludes that the VTs obtained taking into account the local BGP routes do not attend the customer requirements, then steps 6 and 7 in Fig. 5 (b) are executed. Step 6 asks the providers of the local domain for the BGP routes towards the destination that were not advertised by BGP. After this phase, the local domain will have other routes and then steps 3, 4 and 5 in Fig. 5 (a) are again executed to get the VTs of the new routes and negotiate with the domains. If after executing these phases the local domain was not able of finding a route that satisfies the customer requirements, the request can be refused or sent using the best-effort forwarding.

IV. IMPLEMENTATION AND VALIDATION

A. Implementation

The implementation of our architecture considers only the Pull Model. The Pull Model can be put into practice in a shorter period of time since it reflects a very practical scenario and considers the integration with BGP routing. The Pull Model can then be incrementally replaced by the Push Model. The Push Model was used to provide

interdomain connections in optical networks [12]. In this current paper it will be used to be compared with the Pull Model considering the IP network scenario.

In this work, the implementation validates the integration between the service layer (mainly the VTS) with the Internet routing protocol, i.e., the BGP. The service layer was totally implemented using Web services. The main objective of Web services is to help organizations drive their business towards a service-oriented enterprise (SOE) [9]. We show how the service layer interacts with BGP to obtain routes towards a destination and how the VTS interacts with other VTSs in other domains to get BGP routes and virtual topologies. Details about the Web Services infrastructure including aspects related to service registration, service look up and service binding are not considered in this paper. Such details can be found in [10], [12].

To validate our approach, we deployed the architecture in a real scenario running BGP with eight domains. Each domain has its virtual topology represented in XML files. Fig. 6 shows an example of a VT described in XML. In this example, the VT has only an abstract cost (described by the XML tag “weight”) in each virtual link.

```

<?xml version="1.0"?>
<graph>
<node id="mosqueiro/0" weight = "0"/>
<node id="mosqueiro/1" weight = "0"/>
<node id="mosqueiro/2" weight = "0"/>
<edge source="mosqueiro/1" target="mosqueiro/2" weight="10"/>
<edge source="mosqueiro/0" target="mosqueiro/1" weight="6"/>
<edge source="mosqueiro/0" target="mosqueiro/2" weight="6"/>
</graph>
    
```

Figure 6. A VT described in XML.

Fig. 6 represents a VT with three nodes (“mosqueiro/0”, “mosqueiro/1” and “mosqueiro/2”) and three virtual links connecting the three nodes: virtual link from “mosqueiro/1” to “mosqueiro/2” and cost 10, virtual link from “mosqueiro/0” to “mosqueiro/1” and cost 6, virtual link from “mosqueiro/0” to “mosqueiro/2” and cost 6. These XML files store the current state of the domain in terms of QoS. They are usually fed by the administrator (to define QoS classes) and by dynamic tools using probing mechanisms such as *ping* and *traceroute*.

The border routers are running BGP daemons to exchange reachability information. Each border node is represented by a virtual machine. We have used the QEMU virtual machine [20] for this testbed. The BGP MIBs are obtained by using the UCD-SNMP suite [21] (currently called Net-SNMP). The communication between the BGP daemon and the UCD-SNMP agent was done by using the SMUX protocol. Fig. 7 shows the architecture of a node and how the integration between the service layer and BGP is done (the VTS is responsible for interacting with the SNMP in each node. Then, only the VTS is shown in the figure). We created a gateway responsible for receiving the socket request from the service layer and converting it to a *SNMP get command*. The Java language

was used to implement the architecture. The Web services were created using the Apache AXIS 1.2 [22]. The communication between Web services is done through XML-based Simple Object Access Protocol (SOAP) messages over HTTP. The document SOAP style was used as the model for Web services interactions.

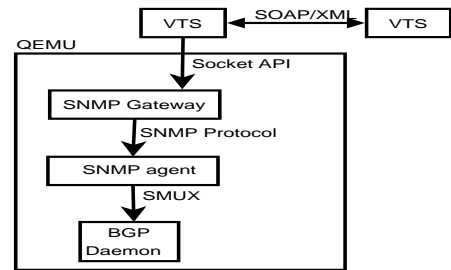


Figure 7. Node Architecture and the Integration between the service layer and BGP.

B. Validation

Fig. 8 shows the scenario, the interdomain topology and the virtual topology of each domain used for this work. For sake of simplicity, each virtual link has only an abstract cost that represents the QoS of the virtual link.

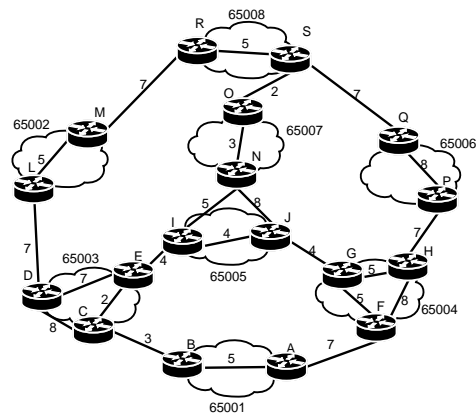


Figure 8. Scenario used in our tests.

In this scenario, the AS 65001 needs to send an interdomain QoS-enabled IP flow towards AS 65008. Firstly, AS 65001 gets its local BGP routes towards 65008. By observing its BGP routes it realizes that there are two paths to AS 65008: the path going through 65003, 65002, 65008 and the path going through 65004, 65006, 65008. Then, the VTs of each domain in both routes are obtained by invoking the VTS. After getting the VTs, the local domain calculates the best path and invokes the E2ENS to negotiate the traffic parameters in each domain. If the acceptance of the IP flow is not possible through either of these two routes, then the local VTS invokes the VTS located in the AS 65003 and VTS located in the AS 65004. By doing so, the local 65001 VTS will obtain the BGP routes not advertised by the BGP running on those ASes. In this case, both ASes (AS 65003 and AS 65004) will return the same route 65005, 65007, 65008. Then, the local AS 65001 knows two new routes and can ask

the AS 65004, 65003, 65005 and 65007 for their virtual topologies.

Observe that although the AS 65001 already has the VTs of AS 65003 and AS 65004 (obtained in the previous interaction), it needs to invoke again those ASes to get the VTs. This is necessary because the VTS in each domain only returns the VT related to the route being analyzed. This is done in every domain. Such mechanism allows each domain to have a very fine control of its virtual topologies. Also, it avoids that the requesting domain gets not allowed information about other routes towards the destination. For example, when the local VTS in AS 65001 invokes the VTS located in AS 65004 to get the AS 65004 VT related to the route 65001, 65004, 65006, 65008, the AS 65004 VTS returns the virtual topology composed of the virtual links F-H and H-P. It does not return other links connecting the AS 65004 to other domains.

After the source domain finds a route that satisfies the QoS requirements and the negotiation has established the contracts with every downstream domain, it is necessary to verify if the route chosen considering QoS constraints is the same as the BGP route chosen by the BGP protocol. Observe that before finding a QoS route, the data packets are being sent towards a given prefix by using the BGP route selected by the BGP algorithm. Then, if the QoS route is different from the one being used, i.e, the egress router and the next hop are different, the Local.Pref attribute of BGP needs to be changed in order to modify the internal routing to use the egress router that represents the QoS-enabled route. Observe that the QoS-enabled route is a BGP route and as such it respects business relationships between the domains. The difference is that without the service layer QoS routing view, the Local.Pref together with other BGP metrics are used to choose the best path towards a given prefix following business rules. When the QoS is taken into account, another route can be selected and then the Local.Pref attribute can be used to change the routing in each domain following QoS constraints. As an example, suppose that in Fig. 8 the AS 65001 has selected the route through 65004, 65006 and 65008 using the normal traditional BGP route selection mechanism. However, after collecting the QoS-related information using the VTS, AS 65001 decides that the best route that satisfies the QoS requirements goes through 65003, 65005, 65007, 65008. Then, the Local.Pref attribute should be modified to point to egress router C.

We have evaluated our prototype using the scenario presented above. We analyzed the times comparing the Push Model with the Pull Model. The main difference between both is that in the Push Model, when a domain wants to send IP packets with QoS towards a destination, the virtual topologies would be present in the source domain because they were advertised earlier. In this case, only the path calculation and the negotiation are necessary. In the Pull Model (on demand model), the virtual topologies need to be obtained at the time of the

request. We run 100 requests and collected the average time. The numbers are shown in Tab. I and discussed below.

TABLE I.
AVERAGE TIME: PUSH X PULL.

Model	time
Push	205 ms
Pull (local BGP routes)	1 sec
Pull (one-level uphill)	3 sec

The average time for the Push Model is 205 ms to attend each request. This includes the path calculation and the two-phase negotiation protocol. The Pull Model took 1 second on average considering only the local BGP routes. This is the time to read the BGP routes using the UCD-SNMP agent, collect the virtual topologies in each domain for each route, apply the CSP algorithm and negotiate with each downstream domain. When the Pull Model needs to go uphill one level in the hierarchy of our scenario, the average time to attend each request increased to 3 seconds. In our case, the VTS needs to obtain the BGP routes from AS 65003 and AS 65004. It is important to say that in the Pull Model, the time to invoke the SNMP agent going through our gateway as shown in Fig. 7 is 664 ms. This includes the communication time to invoke the SNMP gateway (Socket API), to invoke the SNMP agent and to parse the answer from the SNMP agent to return to the service layer. The command being used to read the BGP MIB is the *snmpwalk*. The SNMP agent performance depends on the size of the BGP table. In a real Internet scenario with thousands of entries, the *snmpbulkwalk* should be used.

C. Final Discussion

The contracts between domains should be established considering aggregated traffic demands so that flows to the same destination are seen as a single flow. This can be done by having a traffic matrix to estimate the amount of traffic towards the same destination. This aggregation avoids route oscillation and instability in the routing table. If every single IP flow were treated individually, the Local.Pref attribute should be changed every time a new route to the same destination is found to attend the new QoS requirements.

Dynamic IP flows should be aggregated into an already established QoS class. If not possible, the flow should be sent using the normal best-effort forwarding. However, there could be a mechanism for a given local domain to reroute all or some of the current IP flows in order to give QoS to new traffic demands. The local domain (through the VTS) could ask its downstream domains to rearrange the current IP flows without affecting the pre-established SLAs. If this rearranging is possible, then dynamic IP flows could be aggregated into a QoS class while keeping the QoS of all the previous flows. This issue is left for further study. Also, in more dynamic scenarios the Push Model should take into account a threshold to re-advertise the virtual topologies. Only when the threshold is reached

(e.g. bandwidth lower than a given quantity), a new virtual topology is advertised.

The evaluation proved that the Push Model presents lower times than the Pull Model. However, the Push Model is more indicated to a regional scenario considering condominiums of domains. The Pull Model represents a more practical scenario and can be used in the Internet in a shorter period of time since the Push Model depends on how to group the condominiums. Our intention was not to evaluate all the scenarios but to prove the feasibility of our architecture in terms of integration with the Internet routing protocol. As can be seen, the service layer offers more information to the source domain. It can have a general view about the e2e QoS-enabled interdomain routing alternatives without changing the BGP engine.

V. CONCLUSION

In this paper we presented an architecture to provide a service layer over the current Internet infrastructure. This is a step towards the separation between the network provider (infrastructure provider) and the service provider. The architecture aims at supporting new services for provisioning of interdomain QoS-enabled routing. We discussed how the network virtualization can be used to implement interdomain interactions and presented the Virtual Topology Service as a proof of concept for this virtualization.

Due to the limitations of BGP routing, there is a great effort to develop and deploy new mechanisms to offer new services in the Internet. Our architecture is based on the Virtual Topology Service that is responsible for offering routing information related to QoS among domains. Also, the negotiation service was used to negotiate traffic parameters, reserve resources and establish contract between domains. The use of the Web Services technology makes the architecture more business-oriented and facilitates the definition and the interaction of the services.

We analyzed the integration of the service layer with the BGP routing and proved that it is feasible in terms of how a source domain can start obtaining virtual topologies towards a given destination. The idea of going uphill in the Internet hierarchy is, to the best of our knowledge, a new way to obtain route alternatives for ASes. Based on previous studies, going one level uphill in the hierarchy is enough to have a high number of different routes towards the network prefixes.

We have previously used the virtual approach for interdomain optical networks and proved its feasibility. In this paper we migrated the architecture to attend IP networks and analyzed its advantages over the traditional Internet routing.

REFERENCES

- [1] PlanetLab: <http://www.planet-lab.org/>, 2007.
- [2] GENI (Global Environment for Network Innovations): <http://www.geni.net>, 2007.

- [3] B. Arnaud, "CA*net 4 Research Program Update - UCLP Roadmap. Web Services Workflow for Connecting Research Instruments and Sensors to Networks," *Draft*, December 2004.
- [4] T. Anderson, L. Peterson, S. Shenker, and J. Turner, "Overcoming the Internet impasse through virtualization," *IEEE Computer*, vol. 38, no. 4, pp. 34–41, April 2005.
- [5] N. Feamster, L. Gao, and J. Rexford, "How to Lease the Internet in your Spare Time," *Georgia Tech Technical Report*.
- [6] NSF Report. Overcoming Barriers to Disruptive Innovation in Networking. Report of NSF Workshop, 2005.
- [7] S. Agarwal, C. Chuah, and R. Katz, "OPCA: Robust Inter-domain Policy Routing and Traffic Control," *OPENARCH*, 2003.
- [8] Web Services Activity: <http://www.w3.org/2002/ws/>.
- [9] F. L. Verdi, E. Madeira, and M. Magalhães, "Web Services and SOA as Facilitators for ISPs," *International Conference on Telecommunications (ICT'06)*, Madeira Island, Portugal, May 2006.
- [10] F. L. Verdi, R. Duarte, F. C. de Lacerda, E. Madeira, E. Cardozo, and M. Magalhães, "Provisioning and Management of Inter-Domain Connections in Optical Networks: A Service Oriented Architecture-based Approach," *IEEE/IFIP Network Operations and Management Symposium (NOMS'06)*, 2006.
- [11] F. L. Verdi, E. Madeira, M. Magalhães, E. Cardozo, and A. Welin, "A Service Oriented Architecture-based Approach for Interdomain Optical Network Services," *To appear in the Journal of Network and Systems Management (JNSM)*, Springer, vol. 15, no. 2, June 2007.
- [12] F. L. Verdi, E. Madeira, and M. Magalhães, "On the Performance of Interdomain Provisioning of Connections in Optical Networks using Web Services," *IEEE International Symposium on Computers and Communications (ISCC'06)*, Sardinia, Italy, June 2006.
- [13] M. P. H. et al., "Provisioning for Interdomain Quality of Service: the MESCAL Approach," *IEEE Communications Magazine*, vol. 43, no. 6, pp. 129–137, June 2005.
- [14] G. Cristallo and C. Jacquet, "An Approach to Inter-domain Traffic Engineering," *Proceedings of XVIII World Telecommunications Congress*, 2002.
- [15] L. X. et al., "Advertising Interdomain Qos Routing," *IEEE Journal on Selected Areas in Communications. Vol. 22. No. 10*, pp. 1949–1964, December 2004.
- [16] Z. li, P. Mohapatra, and C. Chuah, "Virtual Multi-Homing: On the Feasibility of Combining Overlay Routing with BGP Routing," *University of California at Davis Technical Report: CSE-2005-2*, 2005.
- [17] W. Xu and J. Rexford, "MIRO: Multi-path Interdomain Routing," *IEEE SIGCOMM'06. Pisa, Italy*, pp. 171–182, September 2006.
- [18] L. Subramanian, S. Agarwal, J. Rexford, and R. Katz, "Characterizing the Internet Hierarchy from Multiple Vantage Points," *IEEE Infocom*, June 2002.
- [19] J. P. et al., "Scalability Analysis of the TurfNet Naming and Routing Architecture," *First International ACM Workshop on Dynamic Interconnection of Networks*, pp. 28–32, September 2005.
- [20] <http://fabrice.bellard.free.fr/qemu/>, April 2006.
- [21] <http://net-snmp.sourceforge.net/>, April 2006.
- [22] <http://ws.apache.org/axis/>, 2005.

Fábio L. Verdi, Ph.D. He is currently a post-doc student at the Faculty of Electrical and Computer Engineering (FEEC), State University of Campinas (UNICAMP), Brazil. He received his Master degree in Computer Science and Ph.D degree in Electrical Engineering both from State University of Campinas (UNICAMP). His main interests include computer networks,

mobility, routing, service oriented architectures, inter-domain services and next generation Internet Architectures.

Maurício F. Magalhães, Ph.D. Received the B.S. in Electrical Engineering from University of Brasília (UnB), Brasília, Brazil, M.S. in Automation from School of Electrical Engineering, State University of Campinas (UNICAMP), Campinas, Brazil and Dr. Engineer from Laboratoire d'Automatique (LAG/CNRS) and Institut National Polytechnique de Grenoble (INPG), Grenoble, France. Currently he works as a Titular Professor at the School of Electrical and Computer Engineering, State University of Campinas (UNICAMP), Campinas, Brazil.

Edmundo Madeira, Ph.D. He is an Associate Professor at the Institute of Computing of State University of Campinas (UNICAMP), Brazil. He received both his Ph.D. in Electrical Engineering and M.Sc. in Computer Science from State University of Campinas (UNICAMP), Brazil. His research interests include network management, optical networks and distributed systems. He has published over 100 papers in national and international conferences and journals. He has also supervised more than 30 master and Ph.D students.

Annikki Welin. She is a Senior Researcher at the Ericsson Research, (Ericsson AB) Broadband & Transport department, Stockholm, Sweden. Her research interests include protocols and architectures. Presently she works with optical networks, management and control planes interworking to automate fast provisioning.