Delivering Application-Layer Traffic Optimization Services based on Public Routing Information at Internet eXchange Points

Danny Alex Lachos Perez , Christian Rodolfo Esteve Rothenberg (Orientador)

Departamento de Engenharia de Computação e Automação Industrial (DCA)
Faculdade de Engenharia Elétrica e de Computação (FEEC)
Universidade Estadual de Campinas (Unicamp)
Caixa Postal 6101, 13083-970 – Campinas, SP, Brasil

{dlachosp,chesteve}@dca.fee.unicamp.br

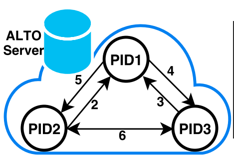
Abstract – Application-Layer Traffic Optimization (ALTO) is a recent standardized protocol that provides a guide in the form of abstract maps (ALTO information) to network applications (ALTO clients) so that they can intelligently select the path or paths to a desired resource. Our work aims to demonstrate that it is possible to provide ALTO information from public routing information at Brazilian Internet eXchange Points (IXPs). We gathered over 4 GB of raw data at 25 Brazilian public IXPs and after initial processing, we converted the BGP routing information to ALTO information and then stored it in a graph-oriented database. Finally, we aim at developing an ALTO server to deliver the ALTO information in response to requests of the ALTO clients.

Keywords – ALTO; IXP; BGP; Internet; PTTMetro; Graph Database

1. Introduction

File sharing applications, Content Delivery Networks (CDNs), real-time communication, among others, use topology information to connect nodes across the Internet and transfer a large amount of data. These distributed applications, however, use limited information with very little knowledge of the underlying network topology and thus the selection of resources are performed randomly or with a partial view of the network, impacting both the application performance and efficient use of the networking infrastructure [**?**].

Aiming to fill this gap, ALTO protocol [**?**, **?**] was designed to provide network information to applications so that they can intelligently select one or several hosts — from a set of candidates — to a desired resource available. The network information is conveyed in the form of ALTO information services (ALTO information for short) by an ALTO server (Figure **??**). At its core, the ALTO information includes a Map Service (Network Map and Cost Map). A Network Map divides all endpoints (IPv4/IPv6 addresses or prefixes) in Provider-Defined Identifiers (PIDs) and a Cost Map provides the cost path between each pair of PIDs.



PID1	192.0.2.0/24, 203.0.113.0/24
PID2	192.51.100.0/24
PID3	2001:db8::/32

NETWORK MAP

Cost	PID1	PID2	PID3
PID1	0	5	4
PID2	2	0	6
PID3	3	6	0

COST MAP

Figure 1. ALTO Information Services

The ALTO implementations prevalent today are created by the Internet Service Providers (ISPs), since they know the dynamics of their networks and the costs associated with peering and transit links [**?**]. The network information (paths, connectivity and routing information) is also available with the Internet eXchange Points (IXPs). An IXP is a shared network infrastructure that facilitates the traffic exchange between Autonomous Systems (ASes) through peering agreements and it represents a public source of BGP data. In the particular case of Brazilian IXPs [**?**], they were created to promote direct interconnection between Brazilian metropolitan network regions with great interest to exchange Internet traffic. Currently including 25 IXPs in operation, is the largest IXP ecosystem in Latin America and one of the world's top ten.

In this work, we collect BGP information at public IXPs operating in Brazil. From this raw data and using the ALTO's external interfaces, we feed an ALTO server in order to demonstrate that third-parties can create and provide ALTO services. To the best of our knowledge, there is no previous published paper on this matter in which user communities or third-parties not associated to the network operators deliver ALTO services from public information at IXPs.

2. Research Goals

The main objective of this work is to create and provide ALTO services based on public information, more specifically BGP routing information publicly available at IXPs.

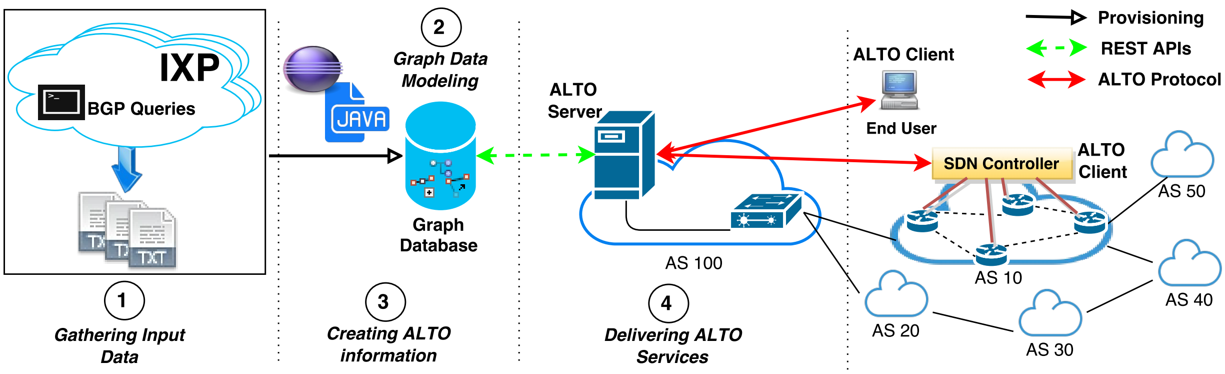


Figure 2. Desing of ALTO as a Service

Providing a mechanism for giving peers information which enables choosing closest neighbors rather randomly, would allow a better experience for the user and can be less costly for the network operators.

Applications could have a large impact on the overall traffic generated using services with knowledge of network topology, in order to establish a connection of a good host that contains the desired resource.

We envision that working at a layer above, with a better understanding of the underlying network would give applications a detailed guide to achieve the desired resources from a set of candidate resources.

To achieve these objectives, the following steps are required:

1. Retrieve routing information from all Brazilian public IXPs.
2. Transform the routing information in ALTO information.
3. Store the ALTO information in a graph-based database.
4. Deliver ALTO information services.

3. Design of ALTO as a Service

3.1. Architecture

Based on the requirements of the previous section, we defined our architecture proposal (Figure **??**). We have four main processes: (i) Gathering Input Data, (ii) Graph Data Modeling, (iii) Creating ALTO Information and (iv) Delivering ALTO information services. Finally, those services are used by ALTO clients.

Details of processes are explained in the next sub-sections[1].

3.2. Gathering Input Data

As a result of this first process, we collect BGP routing information publicly available. To do this, we use a method similar to Brito et al. [**?**], which is also part of our prior work. Each Brazilian IXP is publicly accessed via telnet through the LG servers. Once connected, BGP commands are executed and raw data collected are locally stored in text file format. Next, our dataset passes through a preprocessing, consisting mainly on organizing, filtering and formatting information.

Equally important is to recognize that our dataset does not fully represent the BGP information between the ASes of Brazil. For example, LG servers can mirror the routing information on multilateral agreements; while, for bilateral agreements, such information cannot be accessed because two ASes implement direct peering in a private fashion. On the other hand, it is impossible to assure that information publicly available do not suffer additional filtering by IXP administrators.

For data sharing a reproducibility purposes, the full dataset was placed in our research group repository[2] and it is publicly available.

3.3. Graph Data Modeling

Our data model used in Neo4j[3] graph database is created on the basis of IETF standard RFC 7285[**?**],

[1] It is important to indicate that the first process was completed, the second and third are currently in progress and the latter process is included in the Future Work section.
[2] https://github.com/intrig-unicamp/ixp-ptt-br/
[3] http://neo4j.com

which allowed us to store data in the form of ALTO information after processed as described in step 1 Figure **??**. After, ALTO client requests may then be responded by an ALTO server using graph queries (written in Cypher[4]) in our graph database.

The graph data model in Figure **??** shows an overview of the components involved in our graph structure, which are listed below:
- Nodes: The entities identified in our model and their respective properties are shown in Table **??**.
- Relationships: After defining nodes, we can connect them together to describe their interactions. Table **??** shows the relationships used in our model.

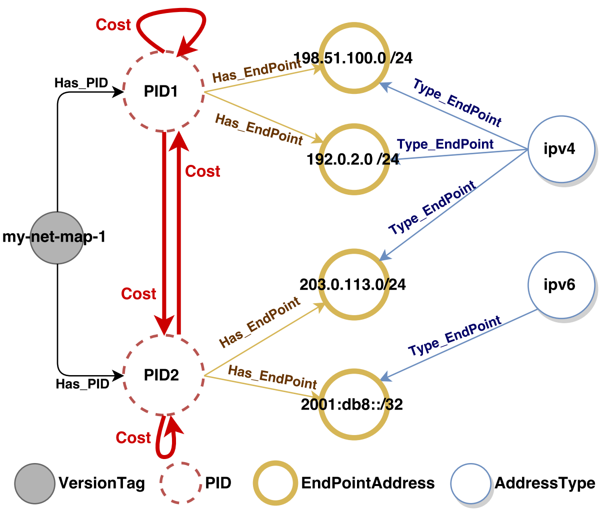


Figure 3. Overview Graph Structure

3.4. Creating ALTO Information

After having our dataset and the data model ready, we need to create the ALTO information and populate our graph database. An ALTO server uses this data to provide information in the form of Map Service. In this section, two Map Service proposals are described. The overall process is the same for both proposals: (i) two algorithms are programmed to read our input dataset, (ii) create Network and Cost Map and (iii) store them in the Neo4j graph database using REST interfaces. The difference lies in how the endpoints are grouped (in PIDs) to create the Network Map and the compute of path cost between PIDs to create the Cost Map.

[4]Cypher is Neo4j's graph query language.

3.4.1. Map Service grouped by AS

Network Map, The Autonomous System Numbers (ASNs) serve as our grouping, where each ASN corresponds to a PID. Each prefix (be it IPv4 or IPv6) will be associated with a particular group (Figure **??** step 1).
Cost Map, The path cost is the absolute topological distance, expressed in number of traversed ASes (Figure **??** step 2).

3.4.2. Map Service grouped by IXP

Network Map, The IXPs serve as our grouping. Each IXP corresponds to a PID and each prefix (IPv4 and IPv6) will serve as address contained within each grouping (Figure **??** step 1).
Cost Map, The path cost is the absolute physical separation distance (in Km) between IXPs based on their geo-location in Brazil (Figure **??** step 2).

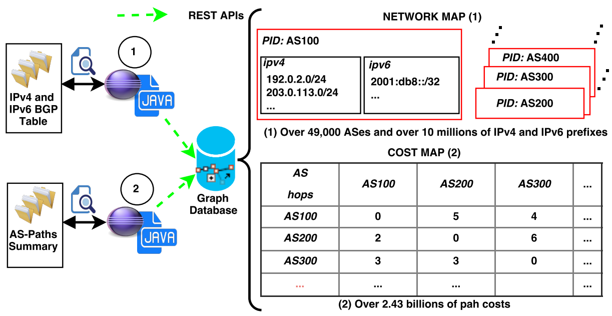


Figure 4. Map Service grouped by AS

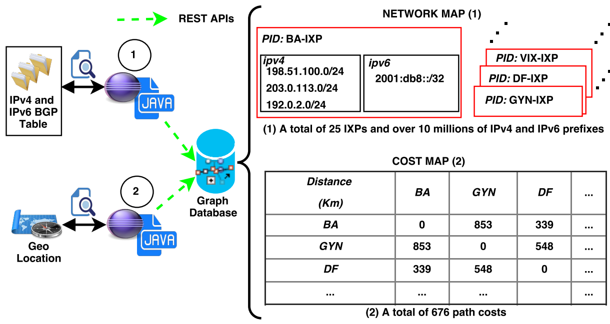


Figure 5. Map Service grouped by IXP

4. Future Work

In order to deliver ALTO Information Services we intend to use the OpenDaylight (ODL) architec-

Table 1. Nodes in our graph data model

Node	Properties	Description
PID	Name	Name of a PID
EndPointAddress	Prefix	IP address and some indication of the length of the mask
AddressType	Type	"ipv4" or "ipv6" to refer to IPv4/IPv6 Endpoint Addresses, respectively
VersionTag	ResourceID	ID unique for a resource (e.g. Network Map or Cost Map)
	Tag	Version for a resource

Table 2. Relationships in our graph data model

Relationship	Start Node	End Node	Description
Has_PID	VersionTag	PID	To know the Version to which a particular PID belongs
Has_EndPoint	PID	EndPointAddress	For grouping Endpoint Addresses to a particular PID
Type_EndPoint	AddressType	EndPointAddress	To indicate the type (IPv4 or IPv6) of Endpoint Address
Cost	PID	PID	To determine the path cost between two PIDs

ture[5] to have a scalable and reliable ALTO server. ODL, an open source Software-Defined Networking (SDN) platform, includes different projects (e.g. ODL controller, OpenFlow plugin, OVSDB, etc.) and the ALTO project is included in its last software release. This initial release of ALTO in ODL is focused on implementing basic ALTO services (defined in RFC 7285) as Northbound APIs (RESTful/JSON) between an ALTO server and ALTO clients. This ALTO northbound APIs generates ALTO services from data stored in MD-SAL data store (an ODL component). In our case, however, we need to modify those northbound APIs to generate ALTO services from data stored in our Neo4j graph-oriented database.

Among other activities, we need to create Cost Maps with dynamic behavior (Performance-related criteria or Charging-related criteria) and define some experimental evaluations, both functionality and performance.

5. Conclusions

We use routing information at IXPs of Brazil to create and deliver multiple ALTO maps. With this, applications (ALTO clients) may have a better knowledge of the network to make the best decision in selecting peers, and the networks that provide ALTO information can use better their network infrastructure. We show that not just the ISPs, but also third-parties can provide ALTO services from public information at IXPs. Future research activity will implement a proof-of-concept ALTO-SDN in order to extend the Cost Maps with dynamic information such as available bandwidth, delay or packet loss rate.

[5] http://www.opendaylight.org/

References

[1] European internet exchange association 2012 report on european ixps. `https://www.euro-ix.net/documents/1117-Euro-IX-IXP-Report-2012-pdf"`.

[2] R. Alimi, R. Penno, Y. Yang, S. Kiesel, S. Previdi, W. Roome, S. Shalunov, and R. Woundy. Application-layer traffic optimization (alto) protocol. RFC 7285, RFC Editor, September 2014. `http://www.rfc-editor.org/rfc/rfc7285.txt`.

[3] S. Brito, M. Santos, R. Fontes, D. Lachos, and C. Rothenberg. Anatomia do ecossistema de pontos de troca de tráfego publicos na internet do brasil. In XXXIII Simpósio Brasileiro de Redes de Computadores (SBRC). 2015. Vitória, ES, Brazil.

[4] V. Gurbani, D. Goergen, R. State, and T. Engel. Making historical connections: Building application layer traffic optimization (alto) network and cost maps from public broadband data. In *Network and Service Management (CNSM), 2014 10th International Conference on*, pages 193–198, Nov 2014.

[5] S. Kiesel, M. Stiemerling, N. Schwan, M. Scharf, and H. Song. Application-layer traffic optimization (alto) server discovery. RFC 7286, RFC Editor, Nov 2014. `http://www.rfc-editor.org/rfc/rfc7286.txt`.

[6] J. Seedorf and E. Burger. Application-layer traffic optimization (alto) problem statement. RFC 5693, RFC Editor, October 2009. `http://www.rfc-editor.org/rfc/rfc5693.txt`.