**A White Paper on Reducing FIB Size through Virtual Aggregation**

Paul Francis, Hitesh Ballani, Tuan Cao, Cornell University

# 1 Introduction

This white paper describes how an ISP can reduce its FIB size, and in so doing delay the time at which the ISP must upgrade its routers. The approach, called Virtual Aggregation (VA) works with existing routers. An ISP can autonomously deploy VA — it does not need to coordinate with its neighbor ISPs. While VA results in some increased load and path latency, the increase is minimal. VA can shrink FIB size by an order of magnitude or more with very little load and latency increase.

While some ISPs are overflowing their FIBs today, we believe that even ISPs that are not yet in this situation may require VA in the near future. This is because FIB size may increase dramatically as IPv4 addresses expire and the address space becomes more fragmented, as well as because IPv6 usage may increase. A prepared ISP can absorb this increase using VA.

# 2 Virtual Aggregation; Basic Idea

The basic idea behind VA is quite simple. The address space is partitioned into large prefixes — larger than any aggregatable prefix in use today. These prefixes are called *virtual prefixes*. All virtual prefixes do not need to be the same size. They may be a mix of \6's, \7's, \8's, and so on. Each ISP can independently select the size of its virtual prefixes. Virtual prefixes are not themselves physically aggregatable. VA makes the virtual prefixes aggregatable by organizing *virtual networks*, one for each virtual prefix. In other words, a virtual topology is configured that causes the virtual prefixes to be aggregatable, thus allowing for routing hierarchy which shrinks the FIB.

Specifically, the virtual networks are configured as follows. Some fraction of routers in the ISP are assigned to be within each virtual network. A router that is assigned to virtual prefix VP1 is said to be an *aggregation point* for VP1. Each router may be an aggregation point for multiple virtual prefixes. Nominally, the best configuration is one whereby each POP has at least one router for each virtual prefix. However, this is strictly speaking not necessary.

When a router is an aggregation point for a given virtual prefix VP1, it has a FIB entry (and RIB entries) for every real prefix within VP1. (We show later how these prefixes are learned.) In addition, the aggregation point router advertises, through BGP, that the virtual prefix VP1 is reachable through itself. If a given router is not an aggregation point for VP1, then it does not need to have any FIB entries for real prefixes within VP1. Rather, it has a single FIB entry indicating how to route a packet to the nearest aggregation point for VP1. As a result, routers can avoid keeping the real prefixes for a large part of the IP address space — all of the space for which they are not aggregation points — thus reducing FIB size.

A typical path for a packet is as follows. Upon entering the ISP at the ingress router, the packet is forwarded hop-by-hop using IP to the nearest aggregation point for the virtual prefix to which the destination address belongs. Upon reaching the aggregation point, the packet is encapsulated in an MPLS header and forwarded to the appropriate egress router, and from there exits the ISP. This is illustrated in Figure 1. Note that other encapsulation schemes are possible, for instance IP-in-IP instead of MPLS. However, MPLS is the most efficient, simplest, and most robust configuration we are aware of.

Routers may maintain prefixes in addition to those within its aggregation point virtual prefixes. These additional prefixes are called *popular prefixes*. The idea here is that, if a large volume of traffic goes to a certain prefix, that prefix may be installed in many or all routers. For instance, in Figure 1, we see that the path from A to C takes an extra hop — via B. This creates additional latency and load. If very few packets take this path, then the load is small and may be tolerated. If a lot of packets take this path, on the other hand, then the real prefix may be installed in router A as a popular prefix. In this case, the packets would not traverse router B, but rather would take the shortest MPLS path directly from router A to the egress router C.

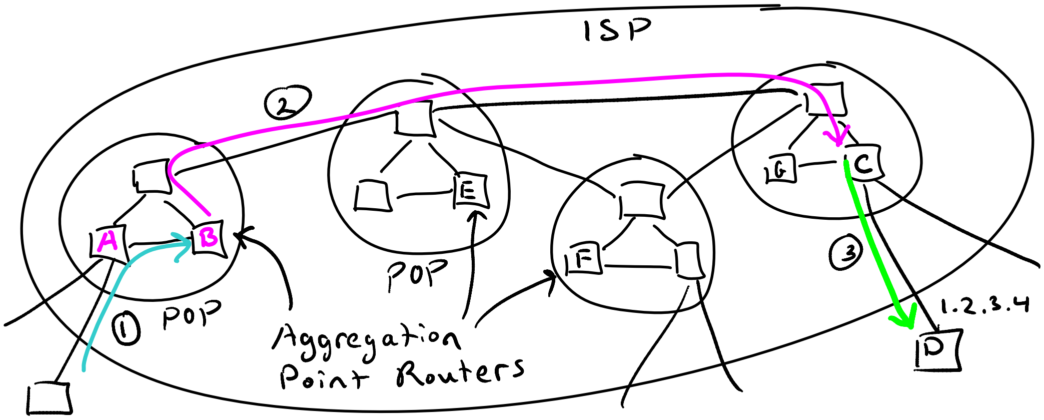

**Figure 1: The basic path taken through an ISP by a packet using Virtual Aggregation. The packet enters at an ingress router A. It is forwarded with IP routing (1) to the nearest aggregation point router B for the virtual prefix of the destination address. Router B puts the packet into an MPLS tunnel (2) destined for the egress router C. The egress router C pops the MPLS header, maps the incoming MPLS tag to a neighboring router D, and forwards the packet to D (3). Note that although routers E, F, and G are also aggregation points for the same virtual prefix, the packet does not need to traverse these routers.**

# 3 Virtual Aggregation; Details

In this section, we describe in detail how the routers are configured for virtual aggregation.

## 3.1 MPLS Tunnels

Every router in the ISP has an MPLS tunnel for every external adjacent router in neighbor AS's (whether the neighbor is a customer network or a neighbor ISP). These tunnels are created as follows. The interface IP address of every adjacent router (e.g. router D's address 1.2.3.4) is statically configured at the egress router (e.g. router C) as a /32 and imported into OSPF. MPLS LDP is used to dynamically configure MPLS tunnels to each of these /32's. As a result, every router has a FIB entry for every adjacent /32 that maps into an MPLS tunnel. At the egress router, a static mapping from the incoming MPLS tunnel to the neighbor router is created. As a result, when the egress router receives a packet on the incoming MPLS tunnel, it forwards the packet to the appropriate adjacent router without requiring a FIB lookup on the destination IP address. For example, in Figure 1, router C is able to forward the packet to router D without having a FIB entry for the destination prefix.

## 3.2 Incoming eBGP prefixes

With normal BPG routing (i.e. without Virtual Aggregation), router C would be expected to eBGP peer with router D. With VA, however, this cannot be done in the normal way, because router C will in general not be an aggregation point for most of the real prefixes advertised by router D, and therefore cannot maintain these prefixes. We currently have two ways to get around this problem. In one, we peer with a route reflector over multi-hop BGP. In the other, we use a trick associated with administrative distance (i.e. in Cisco routers set it to 255) to suppress loading the RIB entry into the FIB.

### 3.2.1 Route Reflector with Multi-Hop Peering

In this mode, the adjacent router D will peer multi-hop with a standard route reflector that has been configured for this purpose. This route reflector is called a *conduit router* because it is the conduit for incoming prefixes via eBGP. Note that the conduit router must maintain the full DFZ routing table. The conduit router, however, does not forward data packets.

This setup is shown in Figure 2. Here we show standard route reflectors acting as conduit routers CR. CR eBGP peers with external adjacencies (1), either through multi-hop BGP, or directly by configuring an Ethernet bridge at router C. Either way, for prefixes received from eBGP, the CR filters them according to the aggregation point they belong to, and according to whether they are popular prefixes, and delivers the prefixes to internal routers using iBGP (2). The BGP Next-Hop attribute is preserved as the external router interface (i.e. 1.2.3.4 for prefixes learned from router D). When the CR advertises prefixes to the external peer, the Next-Hop attribute is set to that of the directly connected router. In this example, the CR sets the Next-Hop attribute for prefixes advertised to router D as 5.6.7.8.

The CR also iBGP peers (3) with other CRs in the ISP (again using normal router reflectors), which would in turn filter prefixes to the appropriate routers. In this iBGP peering, the Next-Hop attribute would still be preserved as that of the external router. This allows internal routers to know which MPLS tunnel to use to forward packets.

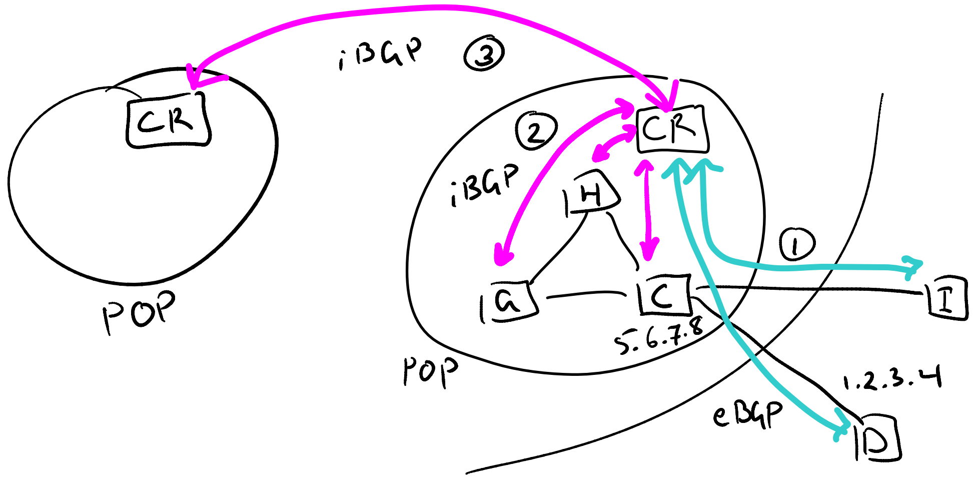

**Figure 2: External routers D and I peer with a conduit router CR (a standard route reflector) using eBGP (1). This eBGP peering may be multi-hop eBGP, or may be L2 bridged through router C. CR iBGP peers with router G, H, and C as a route reflector client (2). CR filters out the appropriate prefixes depending on which virtual prefixes G, H, and C are aggregation points for. The CR also iBGP peers with other CRs (3).**

### 3.2.2 FIB Suppression with Admin Distance

In this mode, the topology and basic configuration are as shown in Figure 1 (i.e., route reflectors are not needed per se). Border routers (i.e. Router C) operate as normal with the exception that the admin-distance command is used to suppress installing RIB entries into the FIB. At least on Cisco routers, this is done by setting the admin distance of entries received from a specified neighbor, that are in a specified access list, to 255 (the largest possible admin distance). This prevents these entries from going into the FIB. With this "255" trick, in spite of the fact that many entries are not installed into the FIB, BGP continues to advertise those entries as normal. In other words, the only effect of this trick is that a specified set of FIB entries are suppressed.

As an example, consider the following Cisco commands and outputs for a router R1 whose eBGP peer is the router at address 198.18.1.200 (note that this example is not related to the testbed experiment described in Section 4):

```
R1#show run
router bgp 456
   neighbor 198.18.1.200 remote-as 123
   distance 255 198.18.1.200 0.0.0.0 1

R1#show run | include access-list 1
   access-list 1 deny 8.0.0.0 1.255.255.255
   access-list 1 permit any
```

The first command shows that an admin distance of 255 has been established for entries heard from neighbor 198.18.1.200 which are in access list 1. The second command shows that access list 1 specifies all entries not within prefix 8.0.0.0/7. The intent here is that R1 is an aggregation point for virtual prefix 8.0.0.0/7. Of course, the access list would be much larger if it allows popular prefixes in addition to virtual prefixes.

We have configured this in the WAIL testbed for Cisco routers. (From our reading of Juniper and Foundry documentation, we believe that the same "255" trick exists for those vendor products.) We found, however, that with Cisco routers the access list results in slow configuration of the routing table. For instance, with full (250K entry) routing tables from the neighbor, it required roughly 30 minutes to establish the peering relationship. This is obviously unacceptable, so while the "255" trick proves that route reflectors are not needed per se, some modifications to router implementations may be needed to make this work in practice.

# 4 Testbed Experiment

We configured Virtual Aggregation as described for route reflectors above in a small router testbed consisting of Cisco 12000's, Cisco 7301's, and switches (see Figure 3). The testbed used is the WAIL testbed at the University of Wisconsin. The configuration models two POPs in an ISP (POP2 and POP3) connected to two external routers (R0 and R8). Two conduit routers are modeled (R2 and R7). The goal of the configuration is to transmit a packet from R8 to R0 that follows a VA path. The packet is destined for an address in prefix 170.168/16. Routers R3 and R6 are configured as aggregation points for that prefix.

All internal routers (not including the conduit routers) are configured with OSPF and MPLS (LDP). Router R1 is configured with a static route to R0 (198.18.1.200):

```
R1#show configuration | include ip route
ip route 198.18.1.200 255.255.255.255 GigabitEthernet0/2 198.18.1.200
```

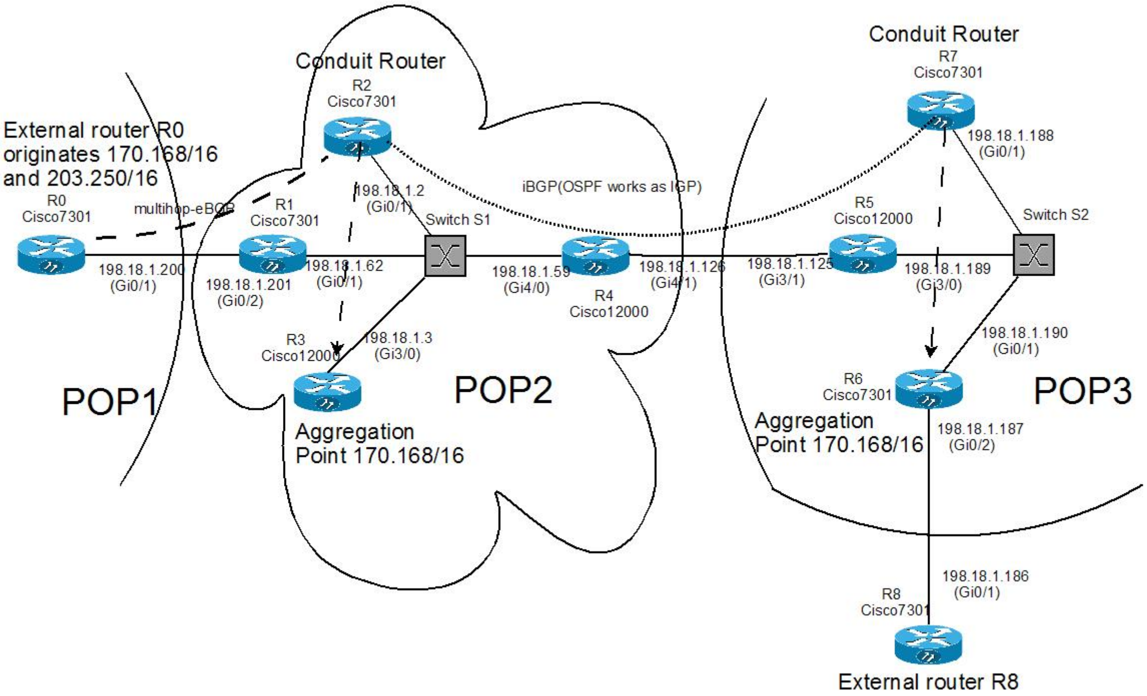

**Figure 3: Configuration used to test Virtual Aggregation.**

This /32 route is carried in OSPF to all internal routers, and LDP establishes an MPLS from all internal routers to R1.

External router R0 eBGP peers with conduit router R2 using multi-hop eBGP. R0 originates 170.68/16 and 203.250/16. R2 is configured to advertise 170.68/16 to the aggregation point router R3, but to suppress advertisements of 203.250/16. R2 does, however, advertise both prefixes to conduit router R7, which in turn advertises only 170.168/16 to the other aggregation point router R6. As such, there is a FIB entry for 170.168/16 in routers R3 and R6 with a next hop of R0's interface 198.18.1.200:

```
R6#show ip bgp
BGP table version is 6, local router ID is 6.6.6.6
   Network               Next Hop            Metric LocPrf Weight Path
*>i170.168.0.0      198.18.1.200            0    100      0 1 i
```

With this in place, we may see how a packet flows from R8 to R0. When the packet reaches R6, it finds the entry in its FIB shown above. For the next hop 198.18.1.200, R6 finds a corresponding entry in its MPLS-forwarding table:

```
R6#show mpls forwarding-table
Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
26     26          198.18.1.200/32   0      Gi0/1      198.18.1.189
```

It then labels the packet with a tag of 26 and forwards the labeled packet to the next hop. Subsequently, the labeled packet goes all through an established LSP towards the edge router R1. When the labeled packet reaches R1, R1 looks up its MPLS-forwarding table:

```
R1#show mpls forwarding-table
Local  Outgoing    Prefix            Bytes tag  Outgoing   Next Hop
tag    tag or VC   or Tunnel Id      switched   interface
27     Untagged    198.18.1.200/32   31518      Gi0/2      198.18.1.200
```

Based on this, it untags the packet, and forwards it to router R0.

## Preliminary Performance Results

We analyzed the expected performance of VA using data from a Tier-1 ISP across North America. In this study, we used a pretty naïve assignment of aggregation points: that is, we assumed that any given POP had either two aggregation points for every virtual prefixes, or that it had no aggregation points for any virtual prefixes. We are currently analyzing a more sophisticated assignment of aggregation points, and expect the performance to significantly improve over what we show here. In this study, we assumed that every virtual prefix is a /7.

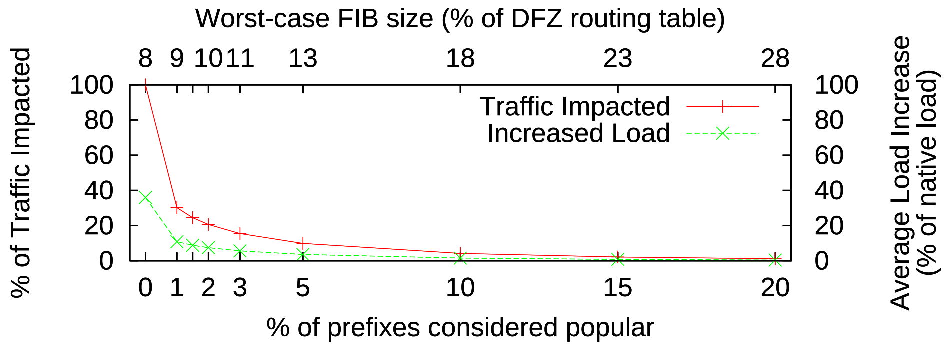

**Figure 4: Graph showing the percentage of traffic impacted (meaning that the traffic follows a path different from shortest path) and the average increase in load on routers against the percentage of prefixes that are considered popular.**

Figure 4 shows the impact of increasing the number of popular prefixes on 1) the percentage of traffic that takes a path longer than shortest path, 2) the increase in router load due to these longer paths, and 3) the worst-case FIB size expressed as a percentage of the DFZ FIB size. Worst-case here means the router in the ISP with the largest FIB (which represents a router in the smallest POP that has aggregation points for every prefix). In all cases, we assume that popular prefixes are used in order of most popular prefix first. As the percentage of popular prefixes increased, the size of the FIB increases, and the percentage of impacted traffic as well as increase in router load decreases. If we look at the data point where a 5x reduction in FIB size occurs (20% FIB size), we see that average load increase is negligible — just a few percent.

Figure 5 shows data from the same experiment, but with a different perspective. Here we show how performance would change over time if an ISP continued to use routers with 240K FIB entries for the foreseeable future: in other words, did not upgrade their routers to large FIB sizes. Here we assume that the ISP would pack its FIB with popular entries at first, but that the number of popular entries would decrease as the DFZ size increased. We assume current projections for DFZ growth — the predicted DFZ size is shown in parenthesis.

On the right-hand Y-axis, we show the average increase in router path length. In practice, most path lengths would be shortest path, with a few being one extra hop (those that first travel to the aggregation point within the same POP), and even fewer being three extra hops (those that first travel to an aggregation point within another POP). As we can see from Figure 5, load increase and path-length increase remain negligible for many years. While this figure assumes that the number of routers within an ISP stays static, what could happen in practice is that the ISP would install more routers in each POP as traffic levels or number of customers increase. Even if these additional routers also had only 240K FIB sizes, the additional routers would introduce additional cumulative FIB into each POP, thus allowing for more aggregation points per POP and subsequent smaller load and path length.

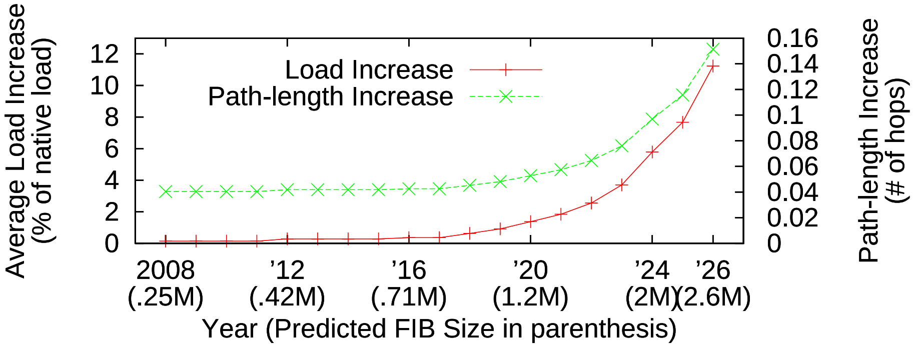

**Figure 5: Increase in path length and router load as the DFZ size increases over time, assuming that routers maintain 240K FIB entries.**

While other ISPs may show different performance results, we don't expect the results to differ significantly between ISPs. We also note that, for the ISP we studied, popular prefixes tended to stay popular over time. In other words, if an ISP measured the popularity of prefixes on a given day, most of the same prefixes would remain popular for several weeks. In our experiments (data not shown) performance based on a given selection of popular prefixes degrades by only a few percent over several weeks time.

The bottom line is that, by using VA, at least as far as FIB size is concerned, an ISP can extend the lifetime of its installed router base, even if that base currently has depleted FIBs, for many years. Of course, the ISP may want to upgrade routers for other reasons, but VA at least gives ISPs the flexibility to do that on whatever schedules make business sense, and is not driven by DFZ growth over which it has no control.