



Waltzing on that gentle trade-off between internet routes and FIB space, an SDN story

David Barroso <dbarrosop@dravetech.com>

Paolo Luente <paolo@pmacct.net>

Forewords

- Academia teaches us that long, seemingly complex, titles is cool
- We refer here to a project that spanned in time across years 2014 and 2015
- This is something you can call SDN; aim is to foster ideas for other use-cases
- All what we are going to speak is public:
 - <https://github.com/dbarrosop/sir>
 - <http://www.pmacct.net/>
 - <https://github.com/pmacct/pmacct>

About the presenters

- **David Barroso**
 - Network Systems Engineer @Fastly
(back then Network Engineer @Spotify)
 - 10+ years in the network industry
 - Python enthusiast
 - Automation junkie
- **Paolo Lucente**
 - Principal Software Developer @pmacct
 - 10+ years measuring and correlating traffic flows
 - Service Providers are his DNA

About Spotify (1/2)

Spotify is a commercial music streaming service providing digital rights management-restricted content from record labels [...] Paid "Premium" subscriptions remove advertisements and allow users to download music to listen to offline.

Over 60M active users per month, 15M paying subscribers, 30M+ songs, 28k songs added per day, available in 58 markets

About Spotify (2/2)

- Four major datacenters:
 - Stockholm, London, Ashburn, San Jose
- Connected to some IXPs globally:
 - DE-CIX, NetNod, AMS-IX, LINX, Equinix Ashburn
- Users are directed to the *best possible* DC:
 - A combination of techniques is used as a metric
 - In case of fault or maintenance users can be redirected to another DC

FIB vs RIB (1/2)

- RIB (Routing Information Base)
 - A representation in memory of all available paths and their attributes
 - This information is fed by routing protocols
- FIB (Forwarding Information Base)
 - A copy of the RIB (usually in hardware) where some attributes are resolved (like next-hop or outgoing interface)

FIB vs RIB (2/2)

- RIB (Routing Information Base)
 - Virtually unlimited (limited only by the memory of the device)
- FIB (Forwarding Information Base)
 - Limited by the underlying hardware

The Internet

- +500k prefixes
- Too many to fit them in commodity ASICs, ie. at the time of the project a typical switch would look like:
 - ~32.000 routes
 - As small as 1RU
 - 72 x 10G ports
 - 262 W
 - ~ 30.000 USD



When you travel ... (1/2)

- Do you carry an atlas?
- Or do you carry a local map?

So .. (granted I'm close to content or eyeballs, ie. I'm not in the business of routing the internet for 3rd parties):

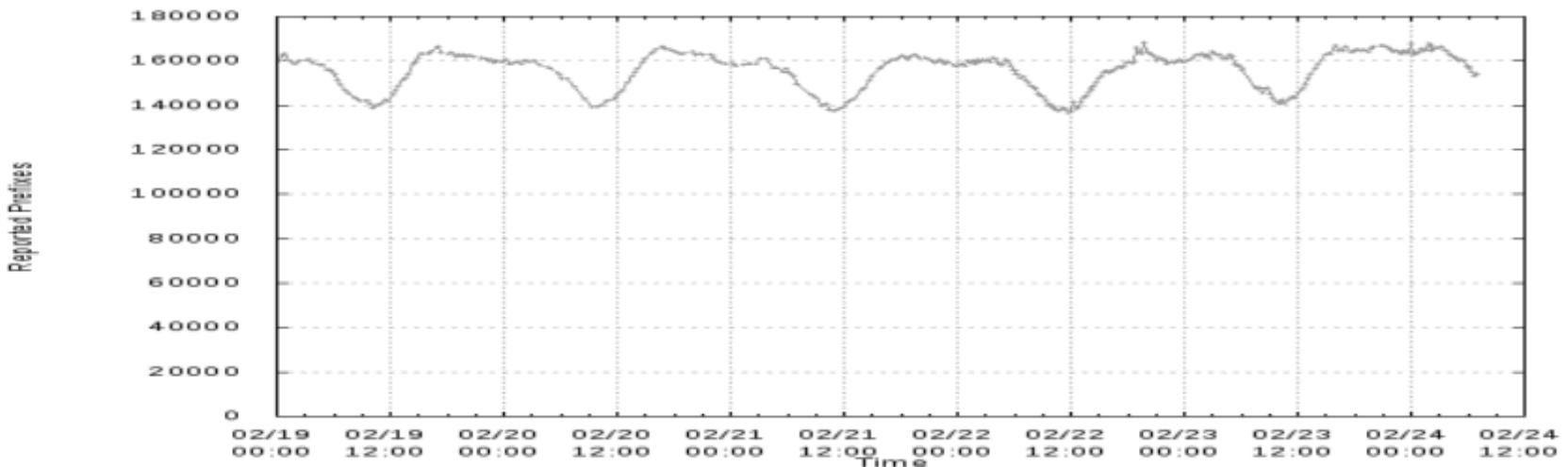
- Why do I need all the prefixes?
- What if I only install the prefixes I really need?

When you travel ... (2/2)

- Example: Spotify datacenter in Stockholm
 - Total prefixes: ~519k
 - Prefixes from peers: ~150k
 - Average # of active prefixes per day: **~16k**
- Example explained:
 - Spotify streams music to users
 - Users are typically served from the closest DC
 - Why would the Spotify DC in Stockholm need to specifically know how to reach users in San Jose?

+1 data point on “active prefixes” (1/3)

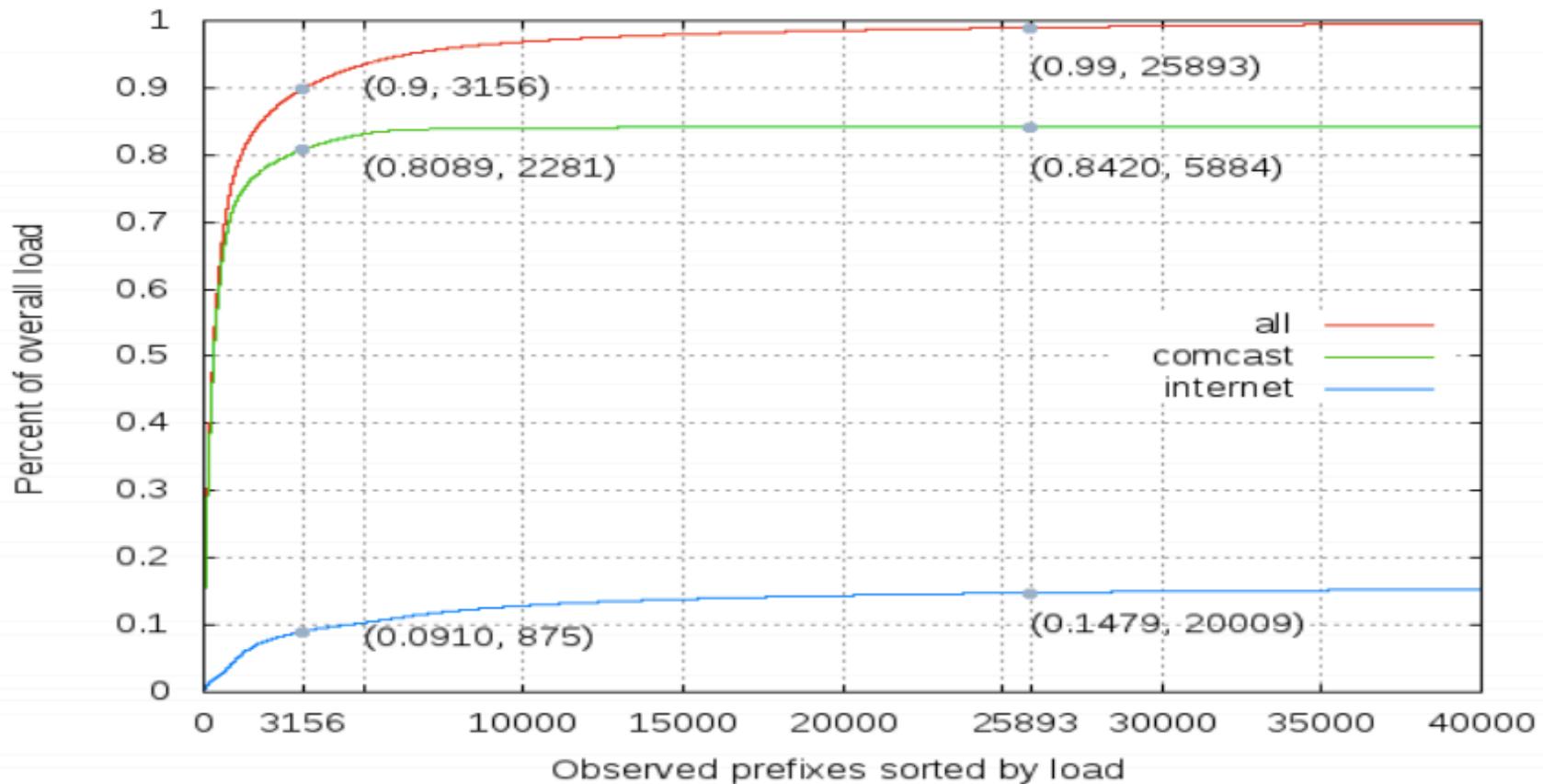
IPv4 prefixes with traffic



- 575k prefixes in the FIB
- 160k prefixes have traffic
- 415k prefixes (72%) with no measurable traffic.

+1 data point on “active prefixes” (2/3)

Details on those 160k prefixes



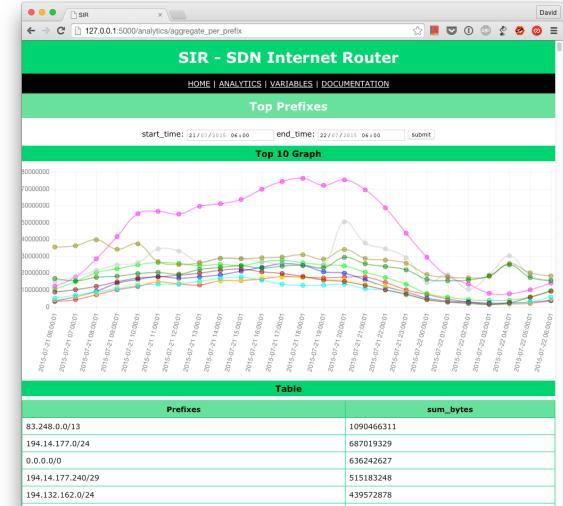
+1 data point on “active prefixes” (3/3)

What does this mean?

- Out of 575k IPv4 FIB entries:
 - No traffic to 415k prefixes (72%)
 - Observe traffic for 160k prefixes (28%)
- Out of the 160k prefixes with traffic:
 - 90% of traffic carried by 3156 prefixes (0.005%)
 - 99% of traffic carried by 25893 prefixes (4.5%)
- 549k prefixes carry 1% of traffic

Hypothesis and goal

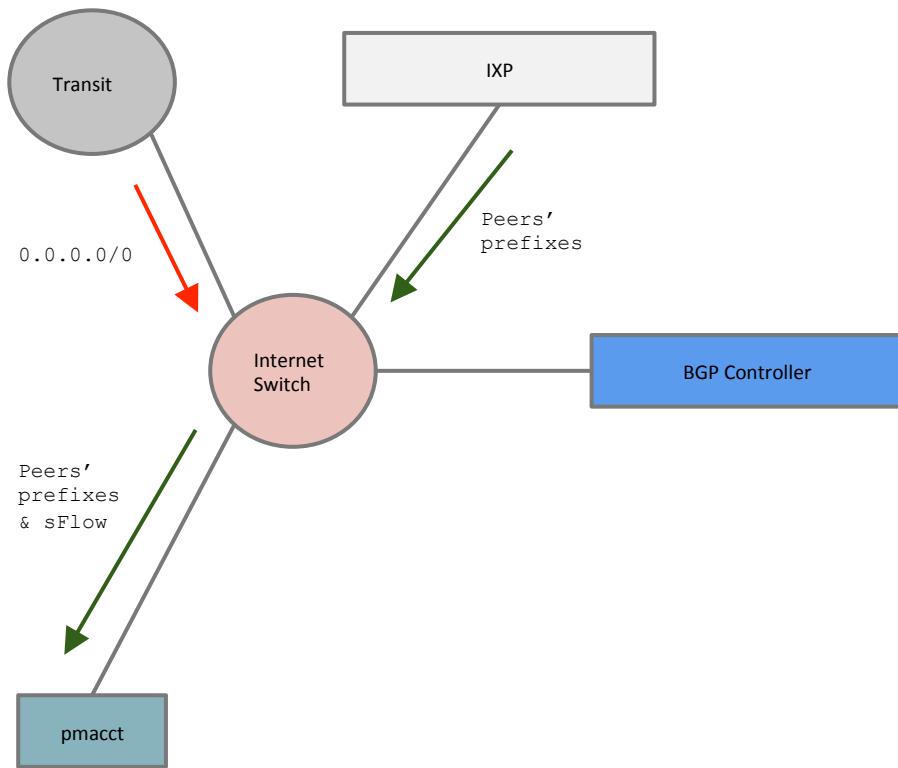
- By analyzing traffic patterns we could lower the amount of prefixes up to the point where we could fit them into a switch
- In simplest term this can be reduced to a TopN problem, where N is the amount of routes the commodity ASIC can fit



Key components of the work

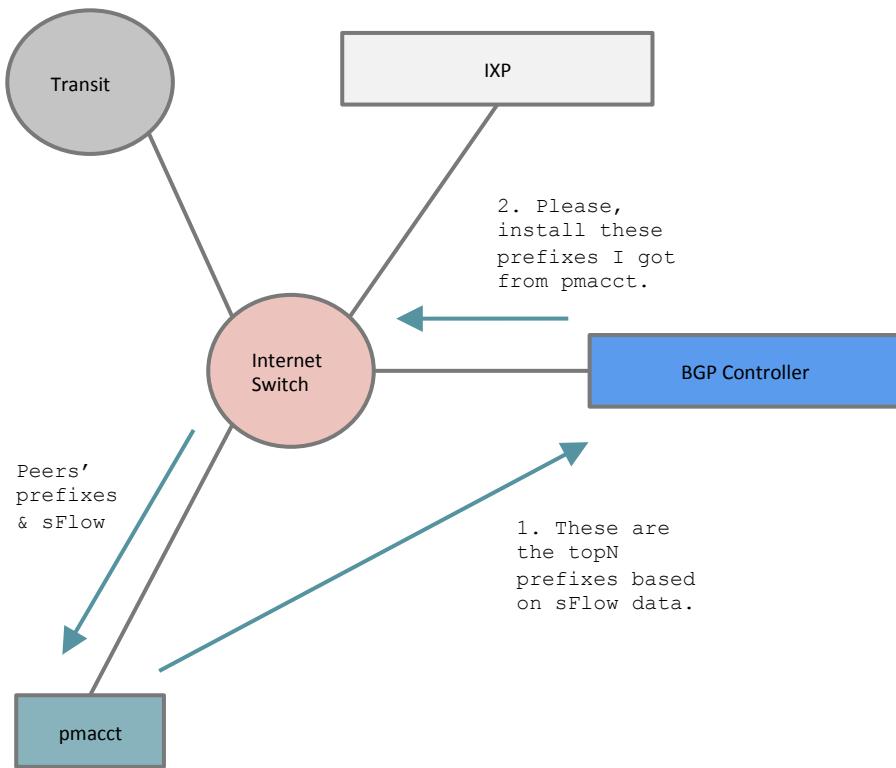
- **pacct** - Collector that can aggregate traffic in a flexible way; BGP information can be obtained by peering with other routers
- **SIR** – an agent to expose information, ie. traffic per BGP prefix or traffic per ASN. This data is provided both via a WebUI and an API
- **Selective Route Download (SRD)** - Feature that allows to pick a subset of the routes on the RIB and install them on the FIB

Prototype overview (1/3)



- Transit will send the default route to the Internet Switch. The route is installed by default in the FIB
- We receive from the IXP all the peers' prefixes. Those are not installed, they are forwarded to pmacct
- pmacct will receive in addition sFlow data

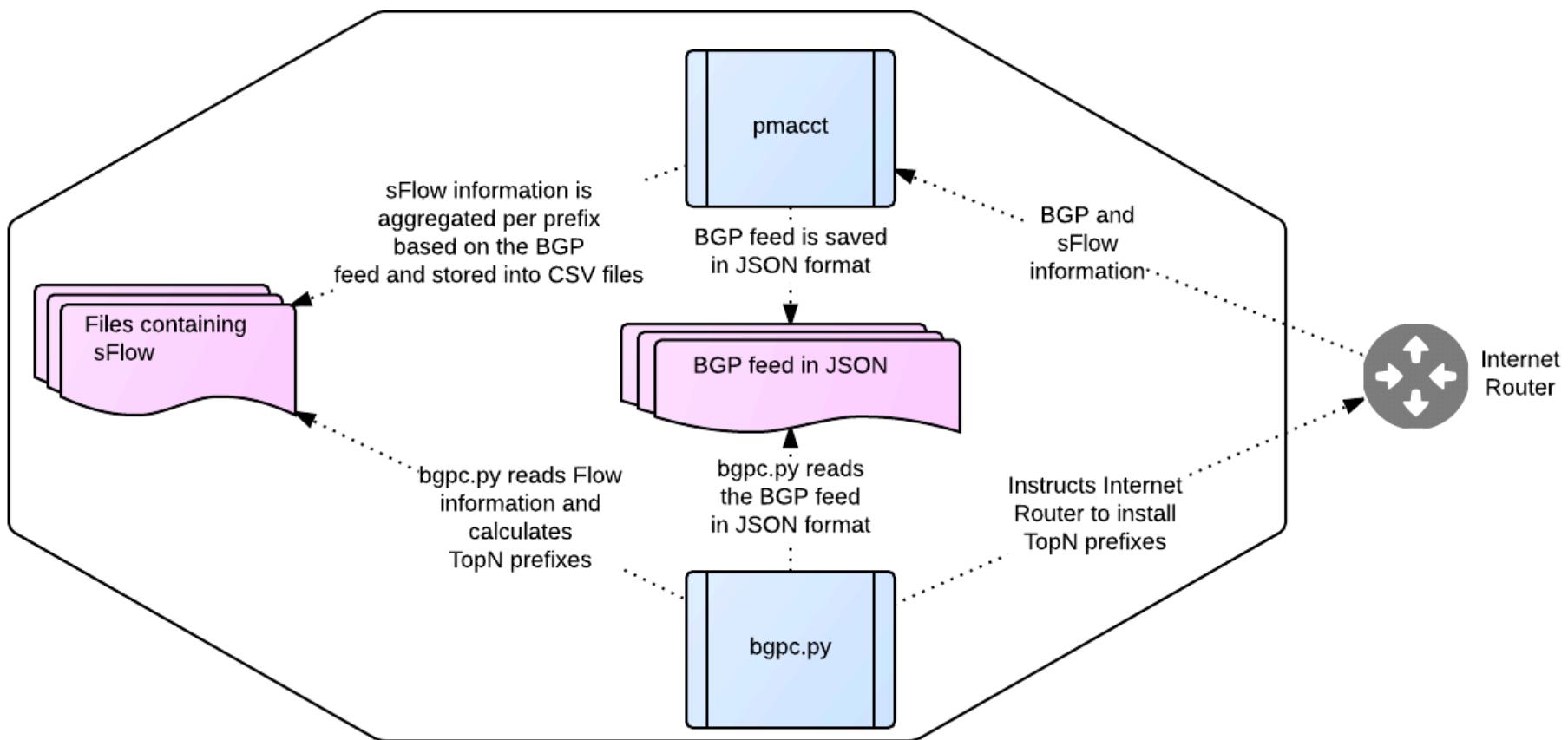
Prototype overview (2/3)



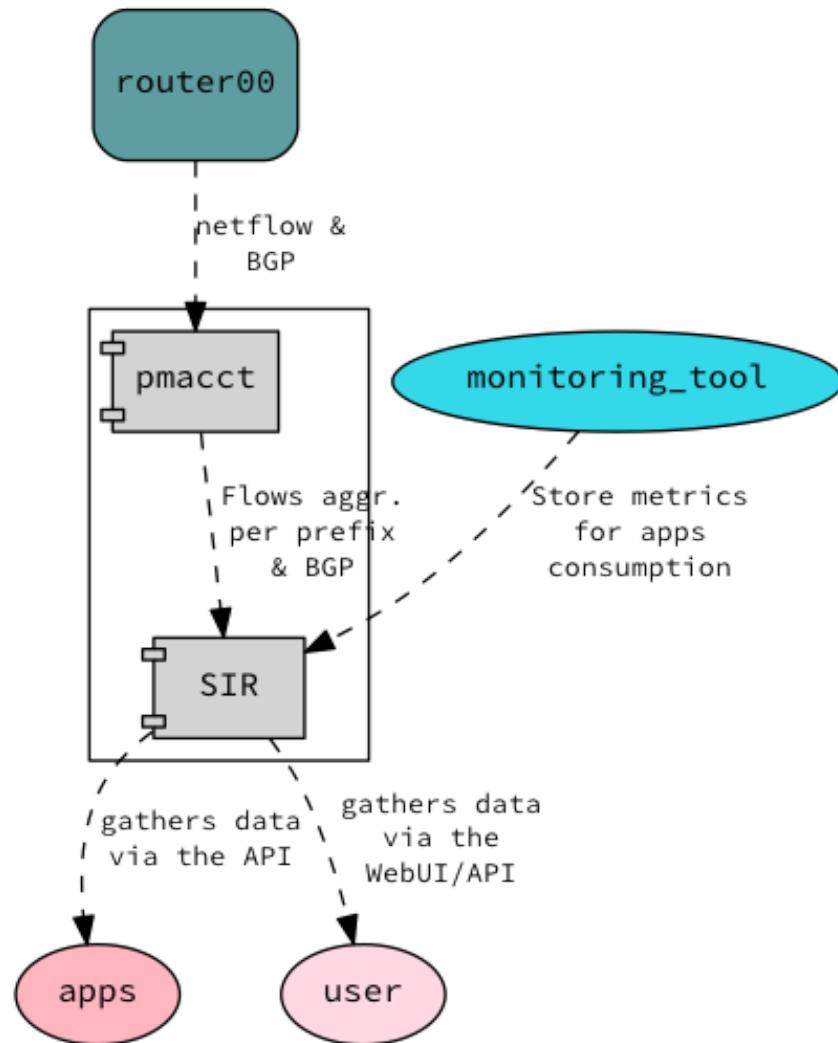
- pmacct aggregates sFlow data using the BGP information previously sent by the Internet Switch
- pmacct reports the TopN* prefixes to the BGP Controller
- The BGP controller instructs the Internet switch to install those TopN* prefixes

* N is a number close to the maximum number of entries that the FIB of the Internet Switch can support

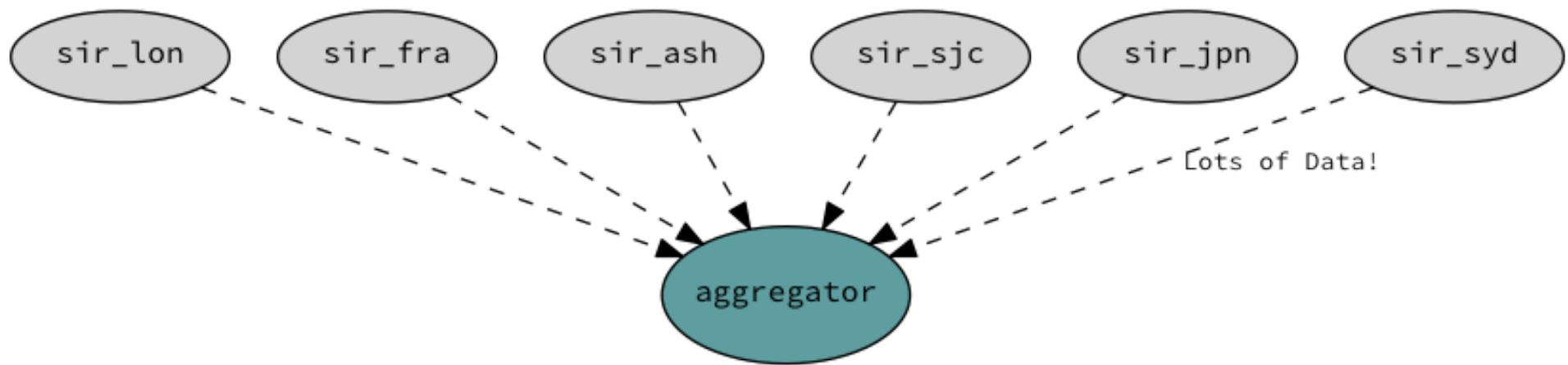
Prototype overview (3/3)



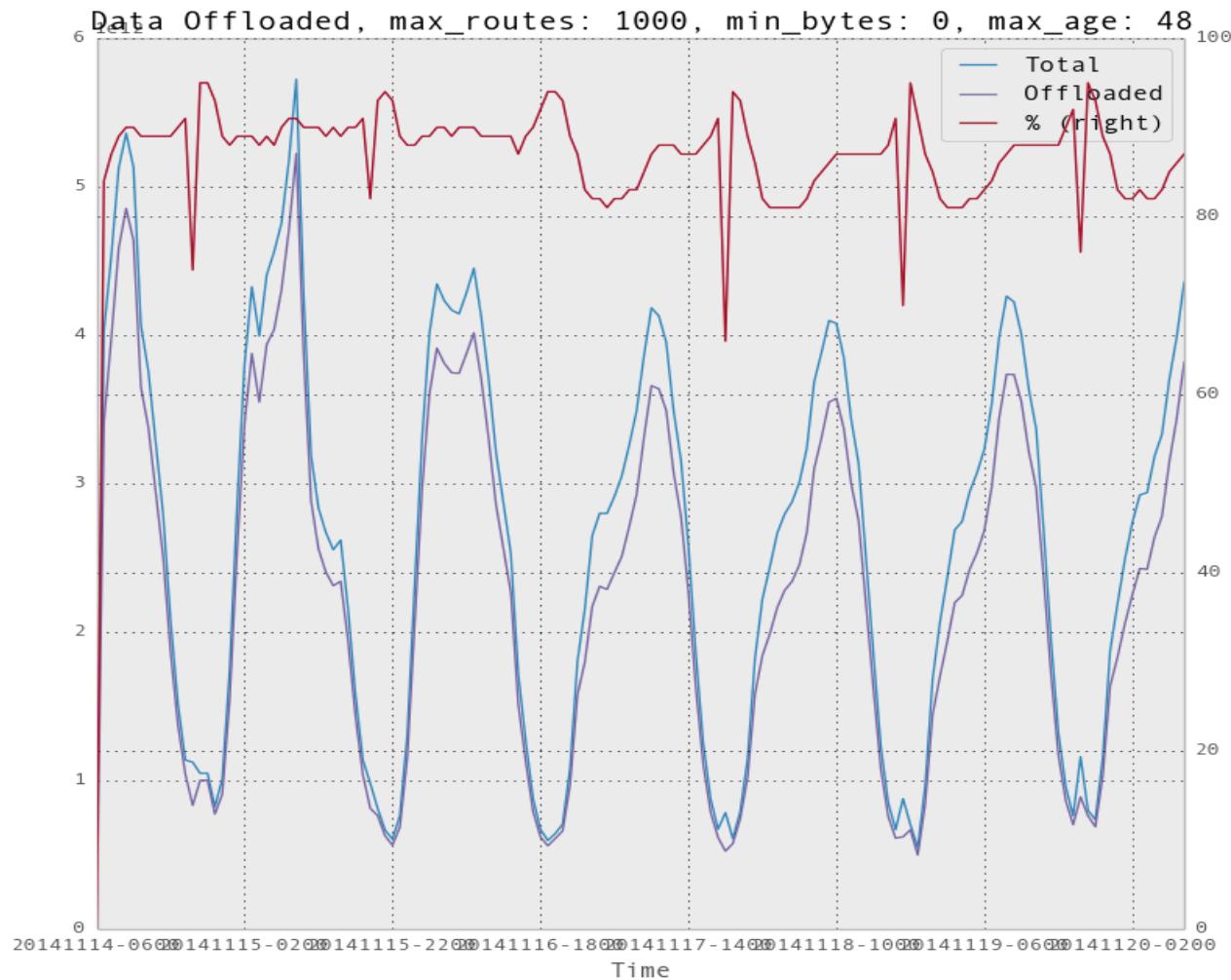
Refactored prototype – SIR (1/2)



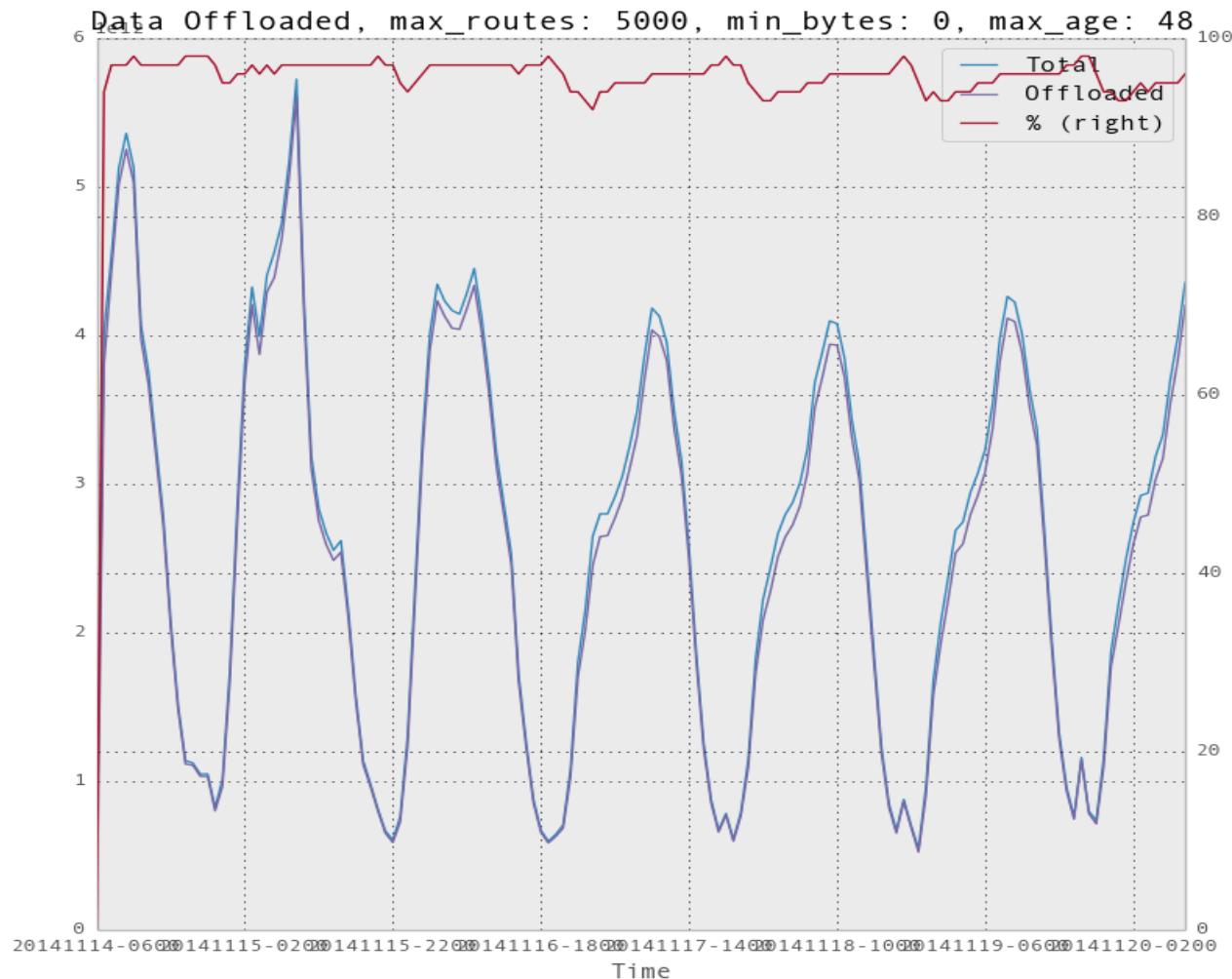
Refactored prototype – SIR (2/2)



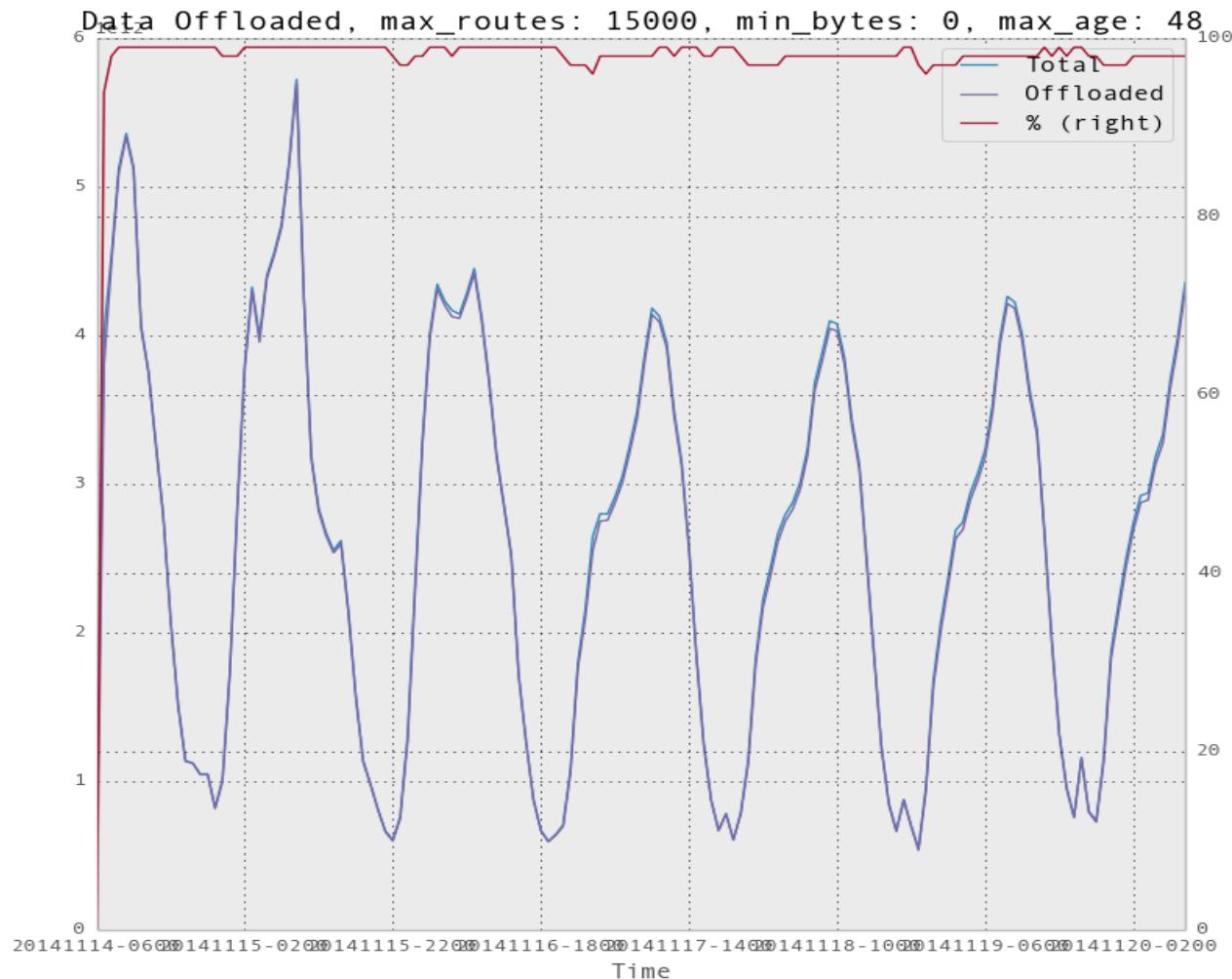
Results from Stockholm DC prototype: top 1k routes (1/4)



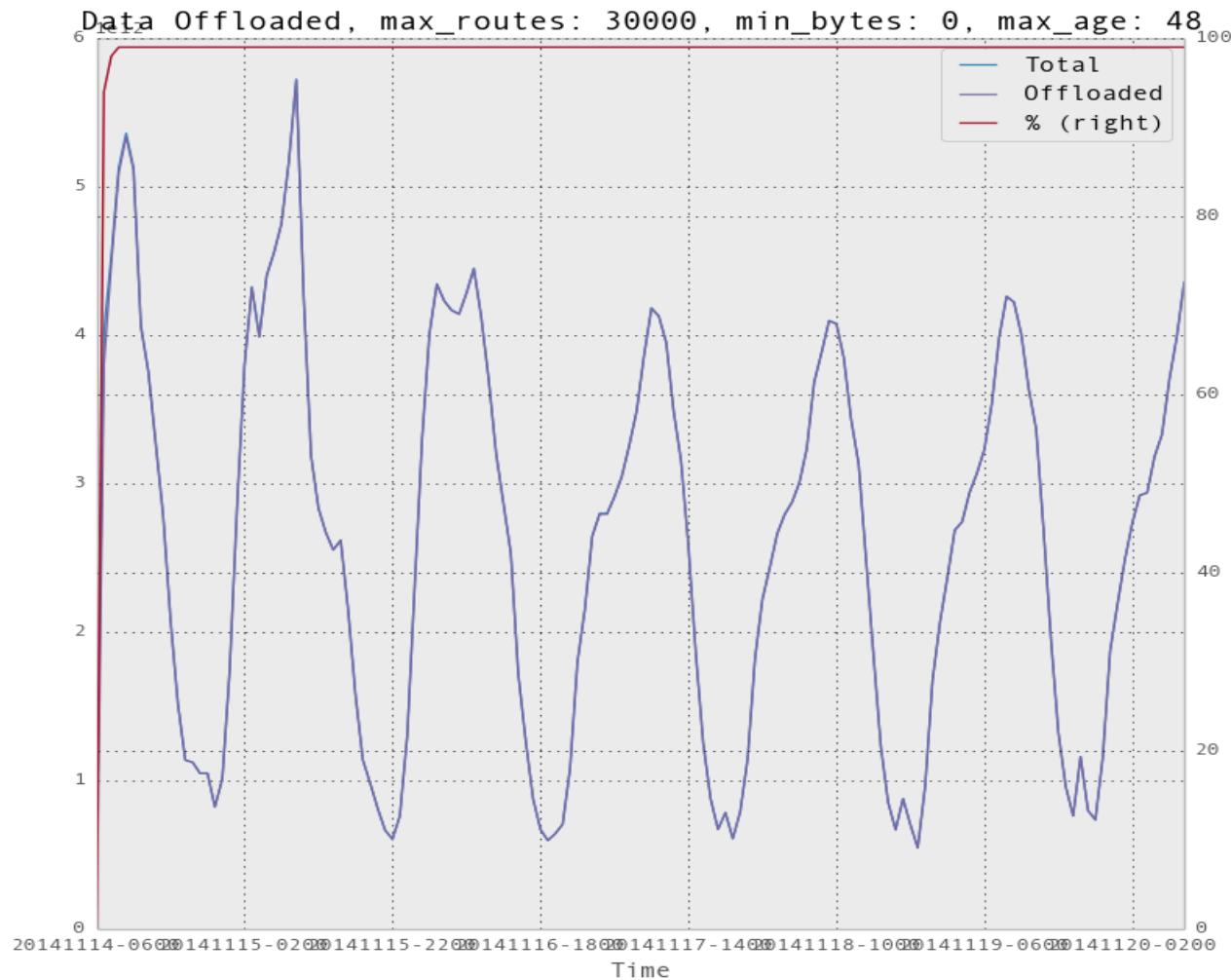
Results from Stockholm DC prototype: top 5k routes (2/4)



Results from Stockholm DC prototype: top 15k routes (3/4)



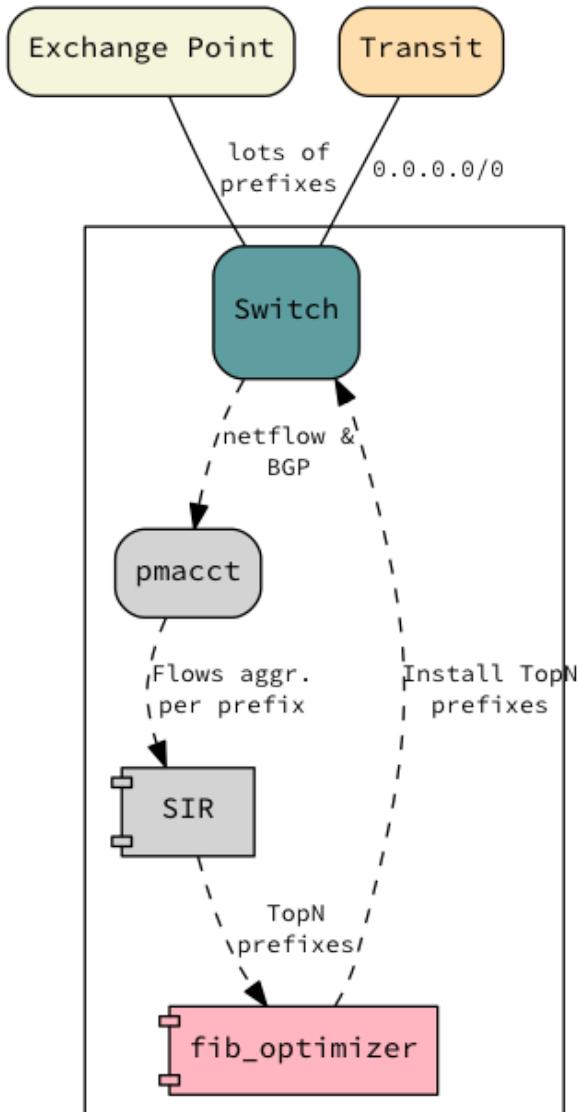
Results from Stockholm DC prototype: top 30k routes (4/4)



Considerations

- The BGP controller updates a prefix list containing the prefixes that the device must take from the RIB and install on the FIB (that is, **selective route download** applied):
 - If a prefix is removed from the RIB it will be removed from the FIB by the device
 - If the BGP controller fails the prefix list remains in the device. Allowing the device to operate normally as per the last instructions

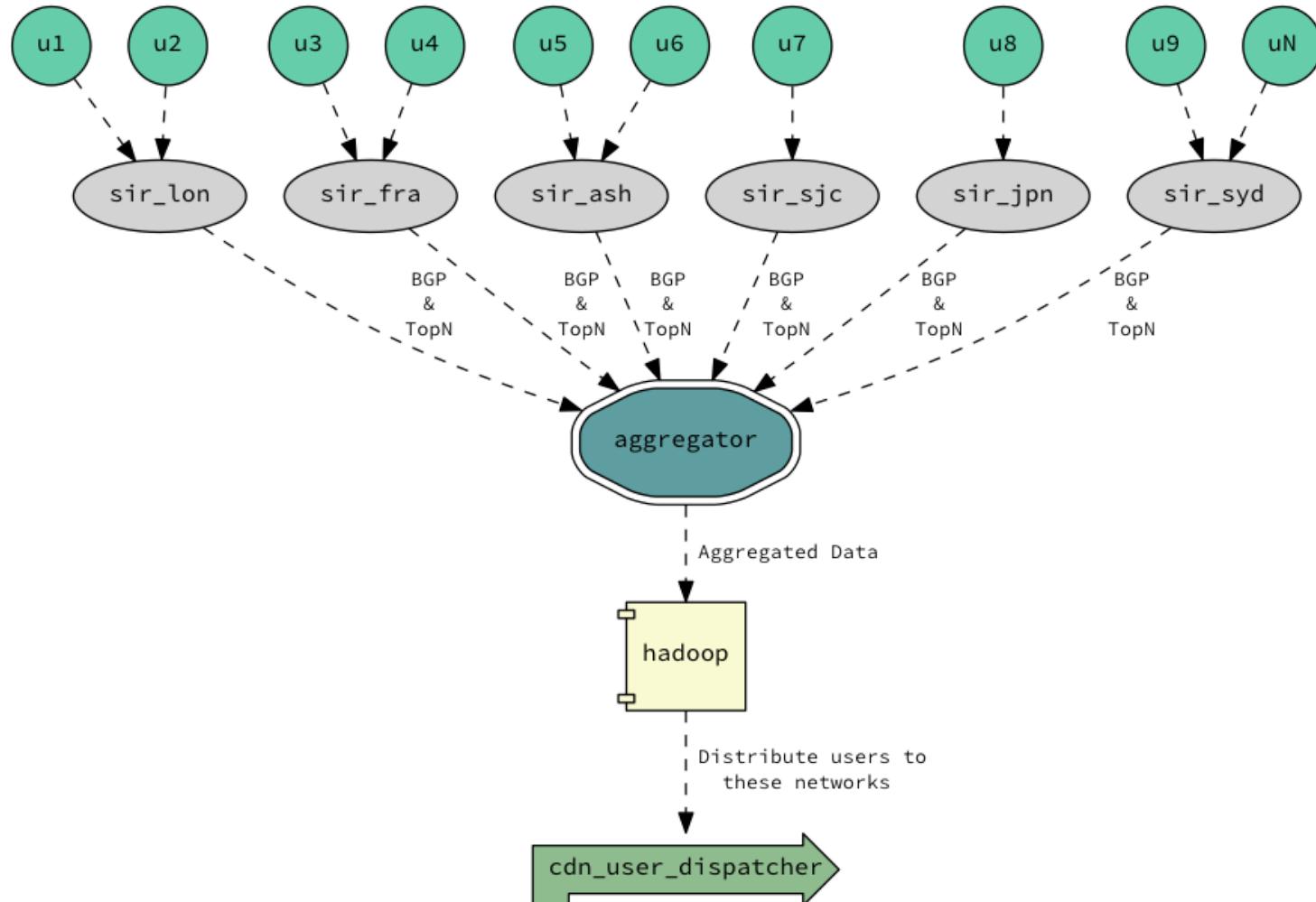
SIR use-cases: SD Peering Router



SIR use-cases: SD-CDN (1/2)

- Add metrics from other sources. Metrics like:
 - Cost of each link
 - Latency
 - Load of each site
 - Reliability
- Once all the data is in, say, Hadoop one could try to analyze global traffic patterns and metrics and distribute users to:
 - Minimize transit costs
 - Maximize capacity usage
 - Improve user experience

SIR use-cases: SD-CDN (2/2)





Thanks! Questions?

David Barroso <dbarrosop@dravetech.com>

Paolo Lucente <paolo@pmacct.net>