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

David Barroso <dbarroso@spotify.net>

Paolo Lucente <paolo@pmacct.net>

RIPE SEE 4 meeting, Belgrade – Apr 2015

Forewords

- Academia teaches us that long, seemingly complex, titles is cool
- This is not 1910's and we are not going to tell you we did split the atom, you can relax
- We were itching to speak either of Cloud, Big Data or SDN. We picked SDN.
- All what we are going to speak is public:
  - **https://github.com/dbarrosop/sir**
  - **http://www.pmacct.net/**

About the presenters

- **David Barroso**
  - 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.

About Spotify (2/2)

- Over 60M active users per month, 15M paying subscribers, 30M+ songs, 28k songs added per day, available in 58 markets
- Four major datacenters:
  - Stockholm, London, Ashburn, San Jose
- Users are directed to the closest datacenter:
  - 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
  - Between 64k-128k LPM prefixes in modern switches with commodity ASIC
  - Between 500k-1000k LPM prefixes in expensive routers/switches with customized ASICs

The Internet

- +500k prefixes
- Too many to fit them in commodity ASICs, ie.:
  - Trident 2 supports 32k prefixes
  - ARAD supports 64k prefixes

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 San Jose need to specifically know how to reach users in Serbia?

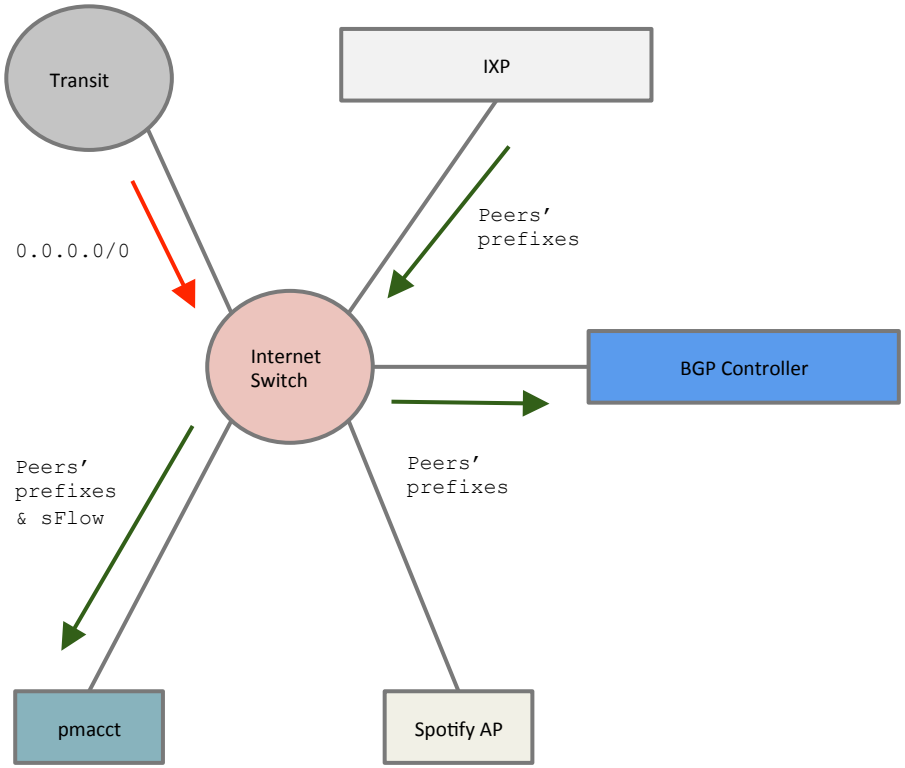
Goal of our work

- Make a selection of “needed” routes from the RIB so to be able to fit them on the FIB of a switch with commodity ASICs
- In simplest term this can be reduced to a TopN problem, where N is the amount of routes the commodity ASIC can fit

Two key components of our work

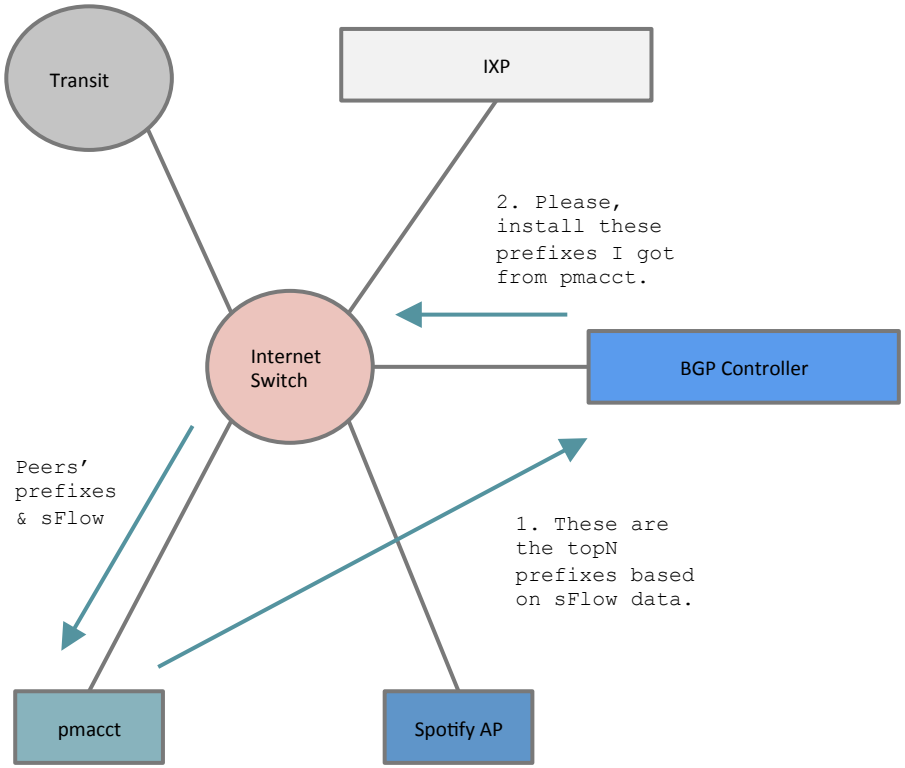
- **pmacct** - Collector that can aggregate traffic by network, AS, BGP peer, etc. BGP information can be obtained by peering with other routers (more later)
- **Selective route download** - Feature that allows to pick a subset of the routes on the RIB and install them on the FIB.

Overview



- 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 and the BGP Controller
- pmacct will receive in addition sFlow data

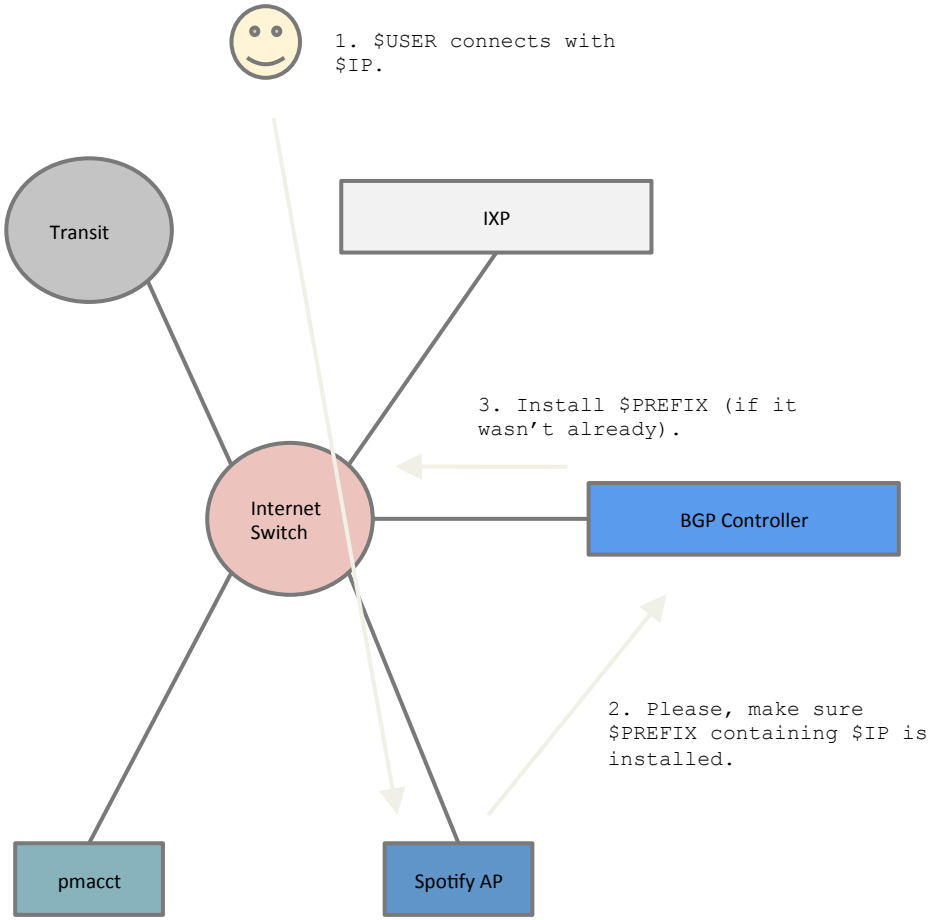
pmacct



- 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

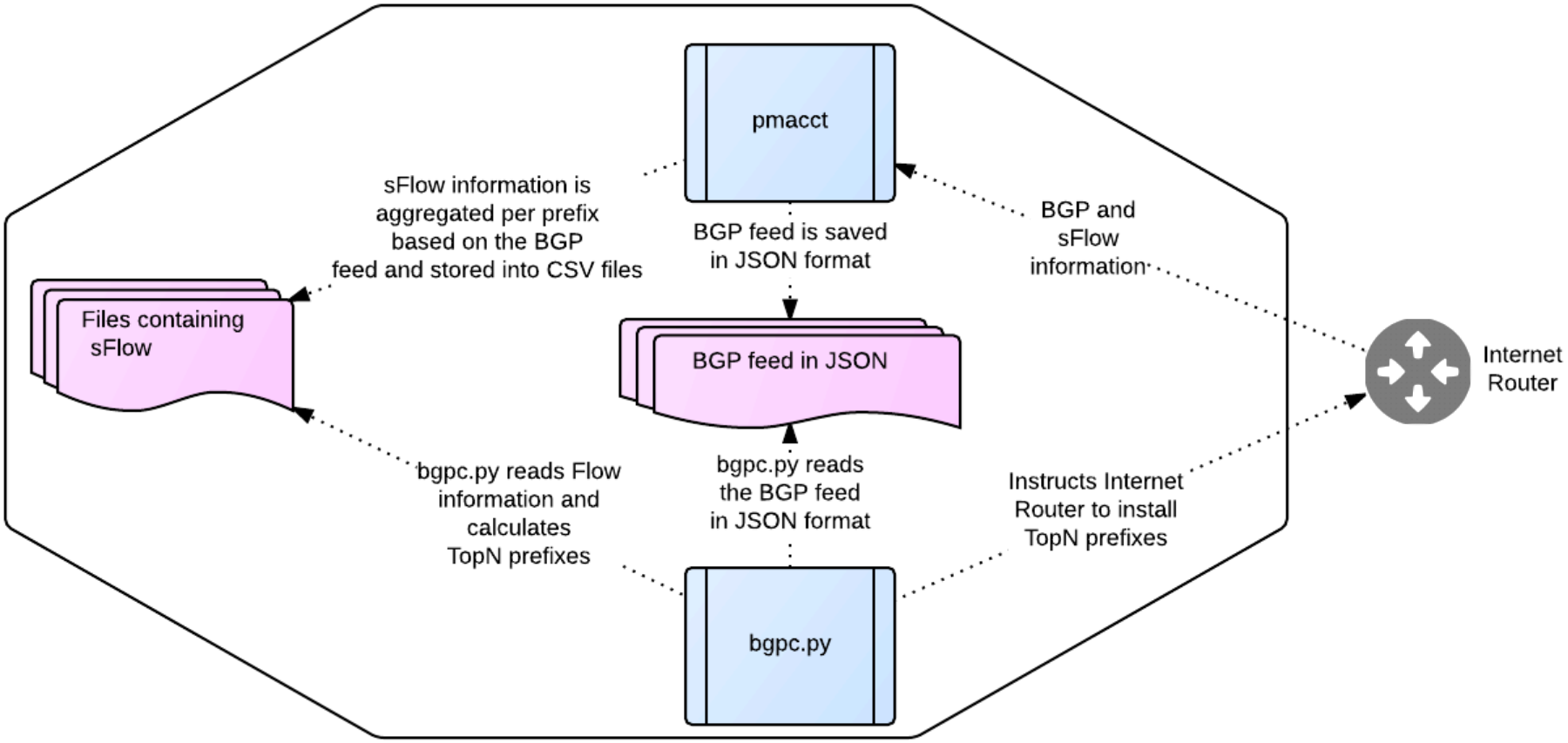
* N is a number close to the maximum number of entries that the FIB of the Internet Switch can support

Spotify AP

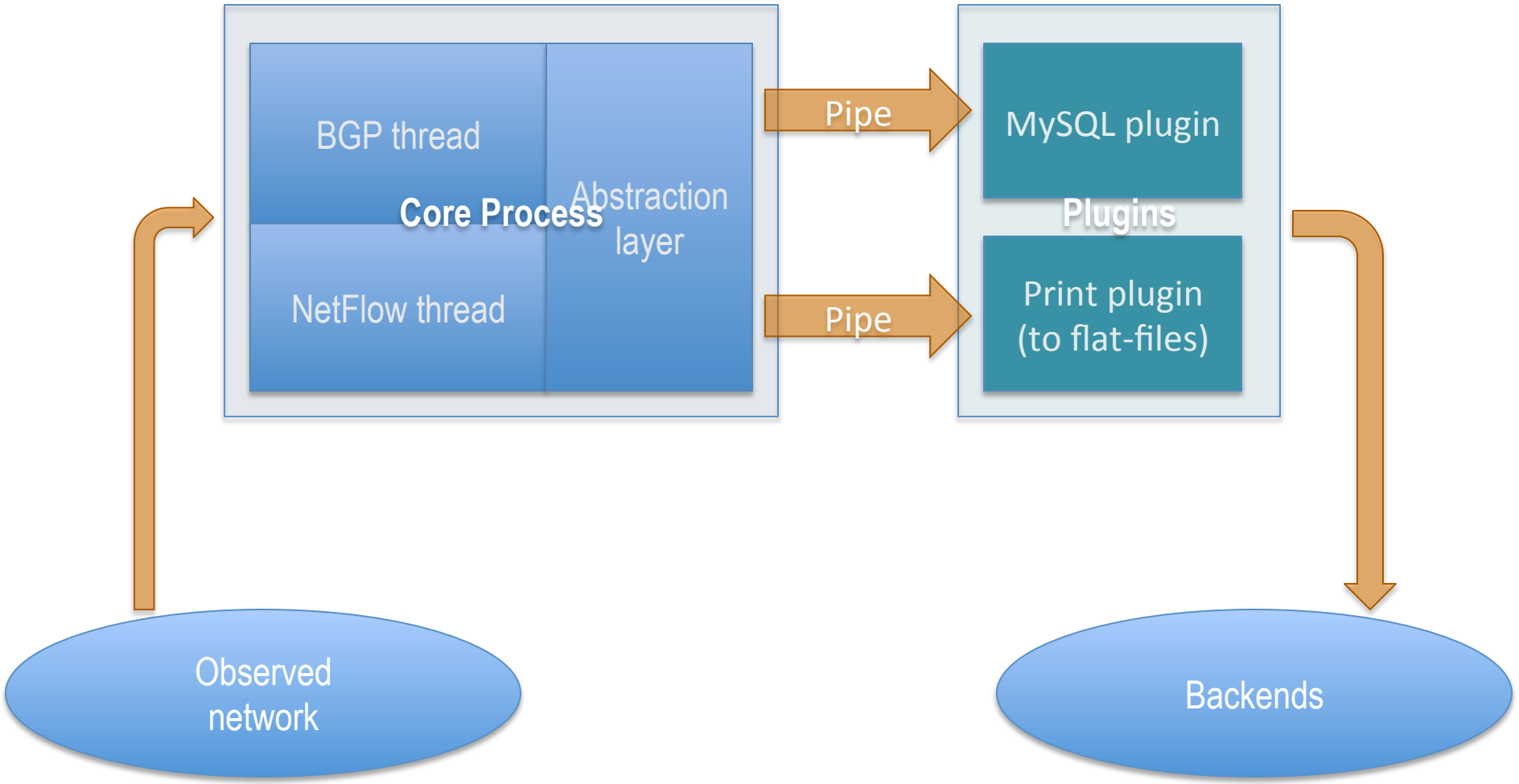


- $USER connects to the service
- The application informs the access point that $USER has connected and requests that the $PREFIX containing his/her $IP is installed on the FIB
  - It might be installed already as another user within the same range might have connected previously or because pmacct reported that prefix as being one of the TopN prefixes
- The BGP controller instructs the Internet Switch to install the prefix if necessary

Internals


pmacct
sFlow information is aggregated per prefix based on the BGP feed and stored into CSV files
BGP feed is saved in JSON format
BGP and sFlow information
Files containing sFlow
BGP feed in JSON
Internet Router
bgpc.py reads Flow information and calculates TopN prefixes
bgpc.py reads the BGP feed in JSON format
Instructs Internet Router to install TopN prefixes
bgpc.py

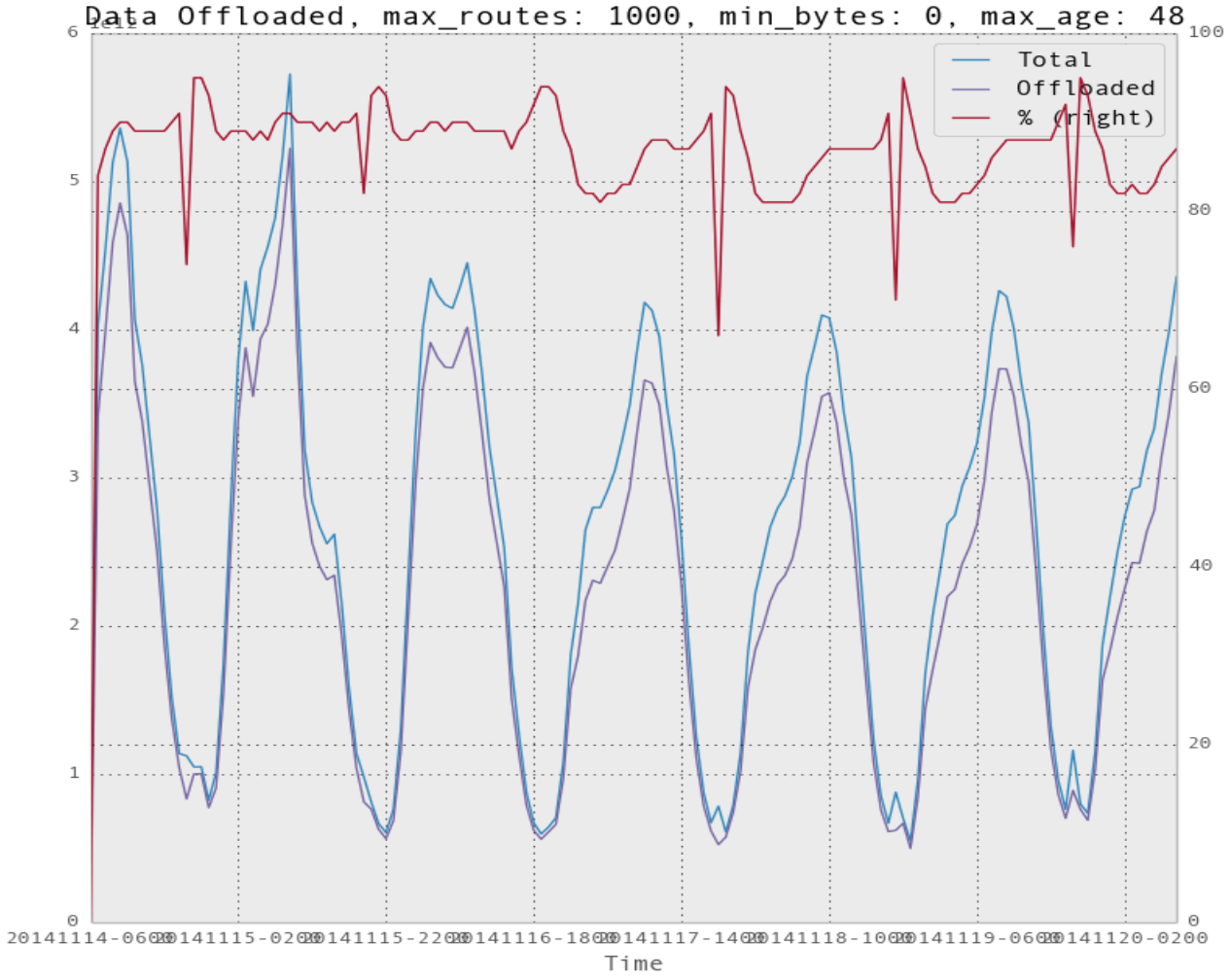
pmacct internals



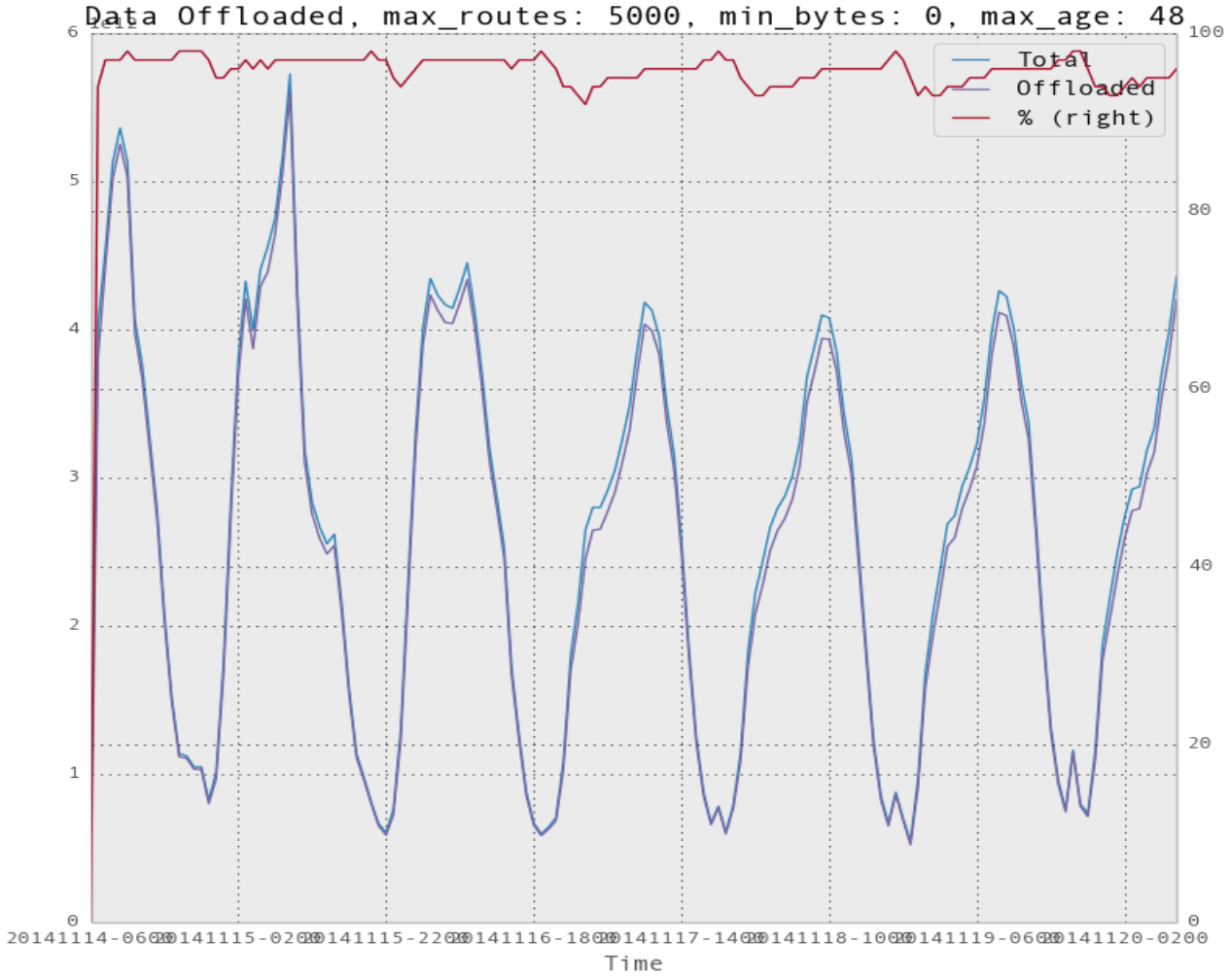
pmacct: BGP integration

- pmacct introduced a Quagga-based BGP daemon
  - Implemented as a parallel thread within the collector
  - Doesn't send UPDATEs and WITHDRAWs whatsoever
  - Behaves as a passive BGP neighbor
  - Maintains per-peer BGP RIBs
  - Supports 32-bit ASNs; IPv4, IPv6 and VPN families
  - Supports ADD-PATH (draft-ietf-idr-add-paths)
- Why BGP at the collector?
  - Telemetry reports on forwarding-plane, and a bit more
  - Extended visibility into control-plane information

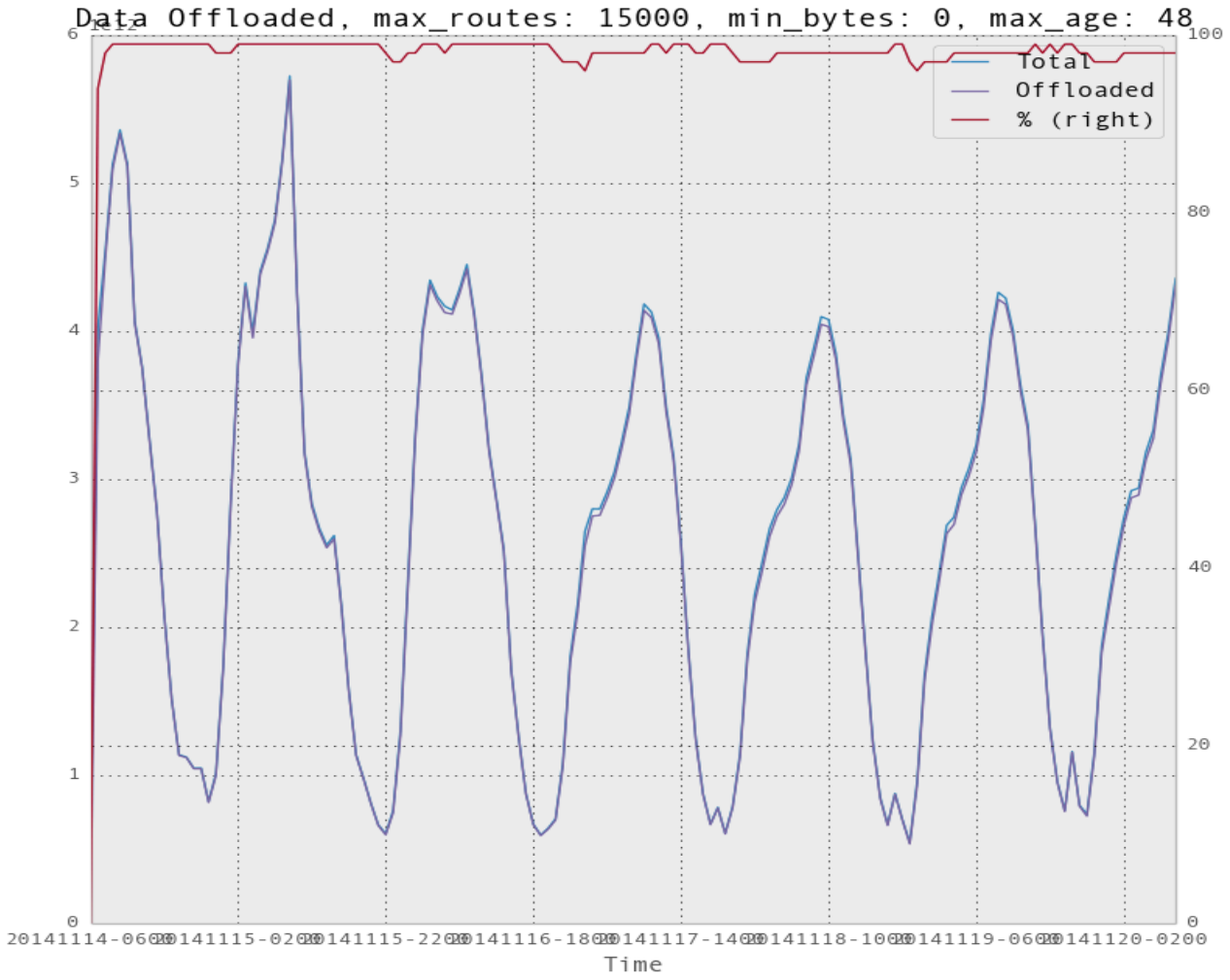
Results: top 1k routes (1/4)



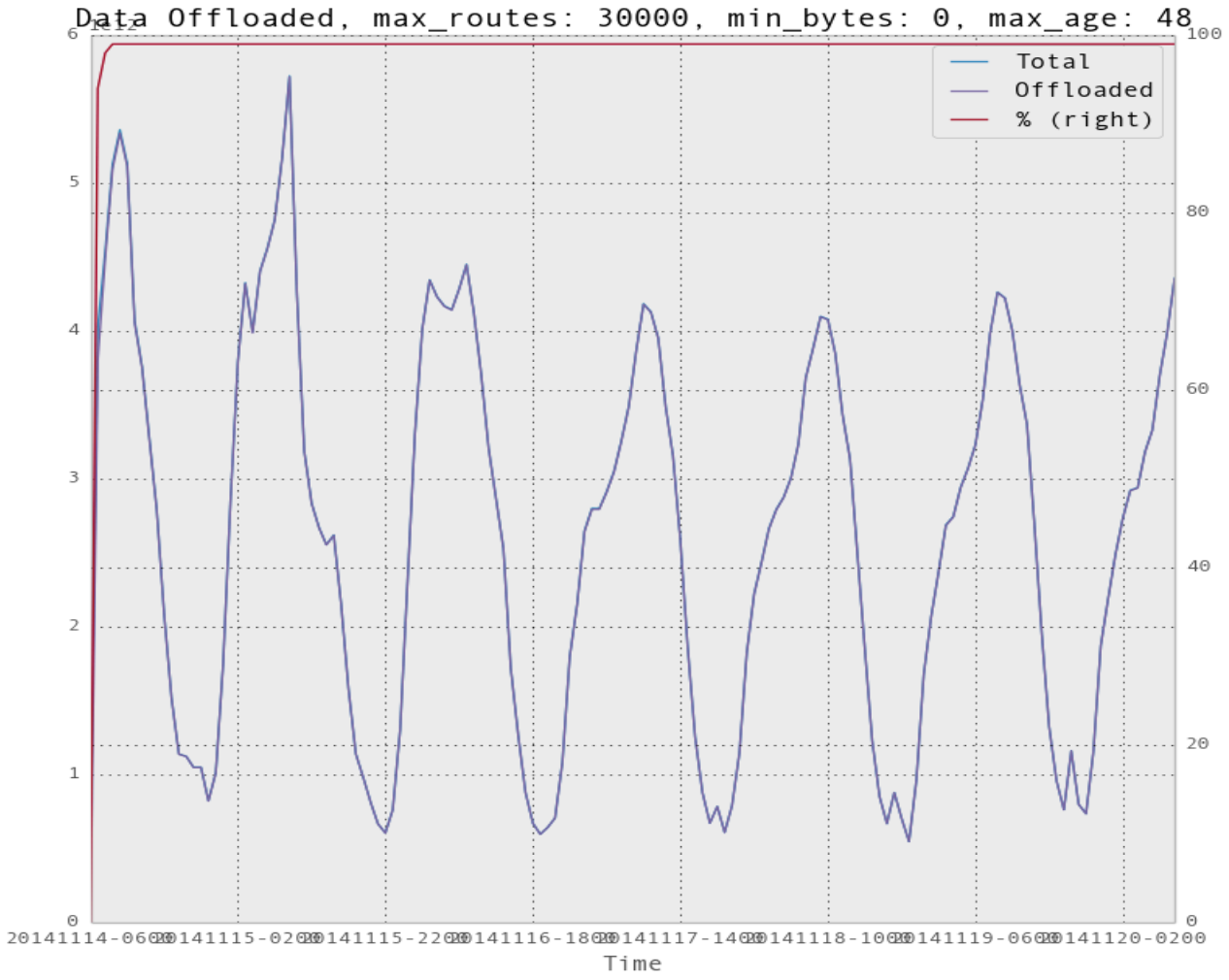
Results: top 5k routes (2/4)



Results: top 15k routes (3/4)



Results: 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

Present and future (1/2)

- Demo run in Spotify Stockholm datacenter, connected to Netnod:
  - Info gathered but no actual changes performed on the Internet Router there
- Pilot to be run very soon by Spotify in cooperation with a major IXP in Europe

Present and future (2/2)

- The BGP controller only computes top prefixes and passes all the information used and the results to plugins
- Plugins can in future do with this information whatever they want:
  - Build reports
  - Build a prefix list and send it to a router
  - Compare possible next-hops, AS PATH's with other active/passive measurements to choose peers based on reliability, latency, etc.

```
# cat etc/config.yaml

max_age: 48
csv_delimiter: ";"
max_routes: 30000
min_bytes: 0
packet_sampling: 10000

… (output omitted)

plugins:
  - 'prefix_data.SavePrefixData'
  - 'statistics.RouteStatistics'
  - 'statistics.OffloadedBytes'
  - 'bird.Bird'

… (output omitted)
```

Thanks! Questions?

David Barroso <dbarroso@spotify.net>
Paolo Lucente <paolo@pmacct.net>