

The RIPE logo is displayed in a light blue, sans-serif font. To its right, there are two vertical white bars of different heights and two horizontal light blue bars of different widths, creating a cross-like graphic element.

RIPE

Anycasting the DNS resolving service for a country-wide ISP network

Kostas Zorbadelos
Network & Systems Engineer



OTE's IP network

- Largest ISP in Greece, coming from the old Greek Telecommunications Organization monopoly
- Presence with nodes all over Greece
- Multiple multi-10Gbs exit points to the global Internet via a subsidiary company, OTEGLOBE
- Interconnections in GRIX

OTE's IP services

- Retail broadband IP connectivity and corporate customers with leased lines / VPN services
- Traditional ISP services (web / domain hosting, email...)
- IPTV and VoIP (IMS) services

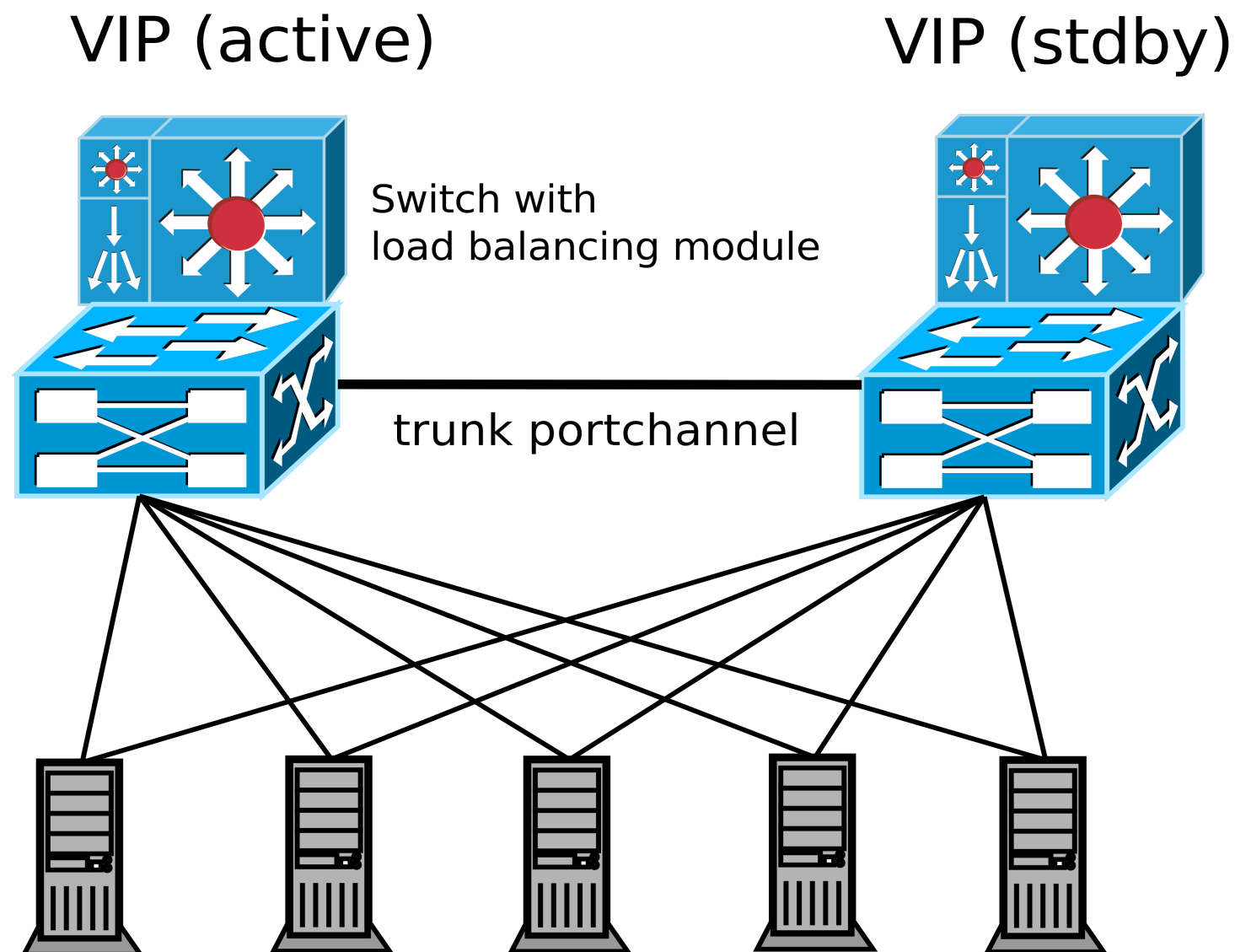
DNS services in OTE's network

- Authoritative DNS for both company domains and customers (~35K zones)
- Domain registrar for .gr
- DNS resolving for all customers in OTE's IP network (even customers with their own AS numbers and IP resources)
- DNS resolving for the infrastructure

Legacy resolving service setup

- Will focus on the resolving services for the rest of the presentation
- Farms of machines behind load balancers in two physical DCs
- Clients are served two resolver IPs and directed to the primary and secondary locations
- Primary location basically a Single Point of Failure (SPOF)

Legacy resolving (schema)



Motivation for service redesign (1/2)

- The services worked fine for many years
- Skepticism even inside the company for the service redesign
- There have been problems however in rare extreme cases
- Load balancer components EOL from the vendor

Motivation for service redesign (2/2)

- No reliance on a single DC location for such a critical service – geographic distribution
- DNS has all the characteristics for a service to be anycasted (connectionless over UDP or short-lived TCP connections, stateless query/responses)
- Many operators already operate anycast DNS services (with root operators the most prominent)
- No need for load balancer components, load balancing handled by the network routing layer
- Simplification of network setup

Initial planning

- Initial target was the infrastructure resolvers
- Tests and evaluations conducted in VM environments
- Initial thought was to have separate interfaces in each DNS node for management / measurements and resolving traffic

Choices and testing

- The dual interface node setup was abandoned
- Too much complexity for little benefit
- /30 point-to-point interfaces on each node
- Easy to remember IPs chosen for the anycast services
- Prototype system consisted of 3 nodes in 2 locations
- Testers were the Engineering teams' workstations (we first tested on ourselves)
- We then proceeded with a production setup for the infrastructure (routers, servers, eng teams' workstations)

DNS anycast node presence

- Having settled on the software choices on each node, the main design choice was *where* the nodes should reside
- The network topology and location of users should be considered
- The infrastructure restrictions also play a major role (not all places could host servers)
- Main targets high availability and optimal (given the restrictions) load distribution

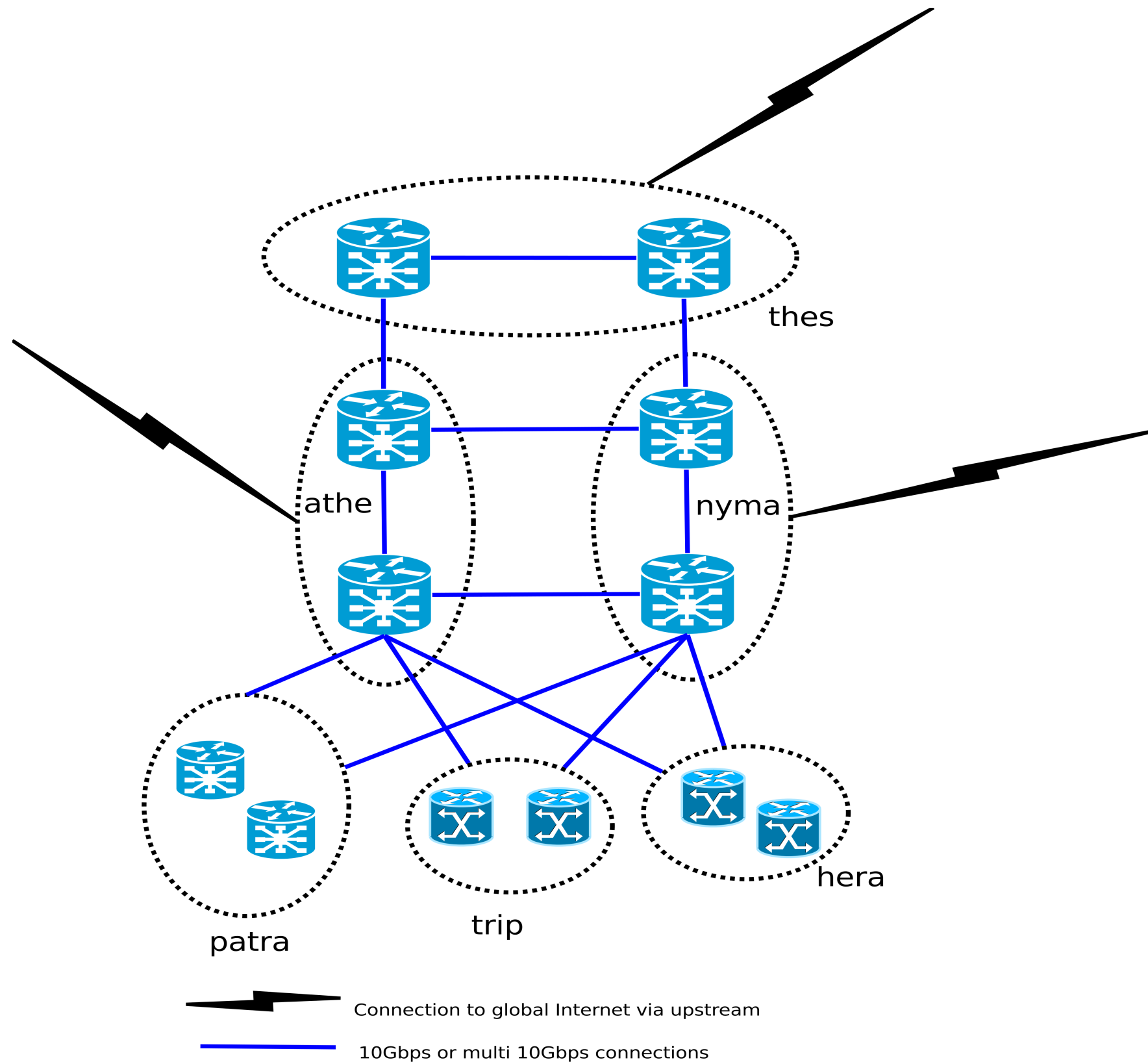
Network topology

- We consider the typical “core”, “distribution” and “access” model
- The core has POPs in major cities (2 locations in Athens, Thessaloniki, Patra, Herakleio, Tripoli) with multi 10G interconnections
- The distribution layer performs layer 3 aggregation and has presence in many more locations (~40 POPs)
- The access layer (considering L3 access) is even more distributed
- Cisco gear in core, distribution, Juniper in access

The Core

- Main and only function to quickly forward packets
- Multi 10Gbps connectivity
- Contains the exit points to the global Internet and GRIX
- Not recommended (by the vendor) to contain IP service nodes

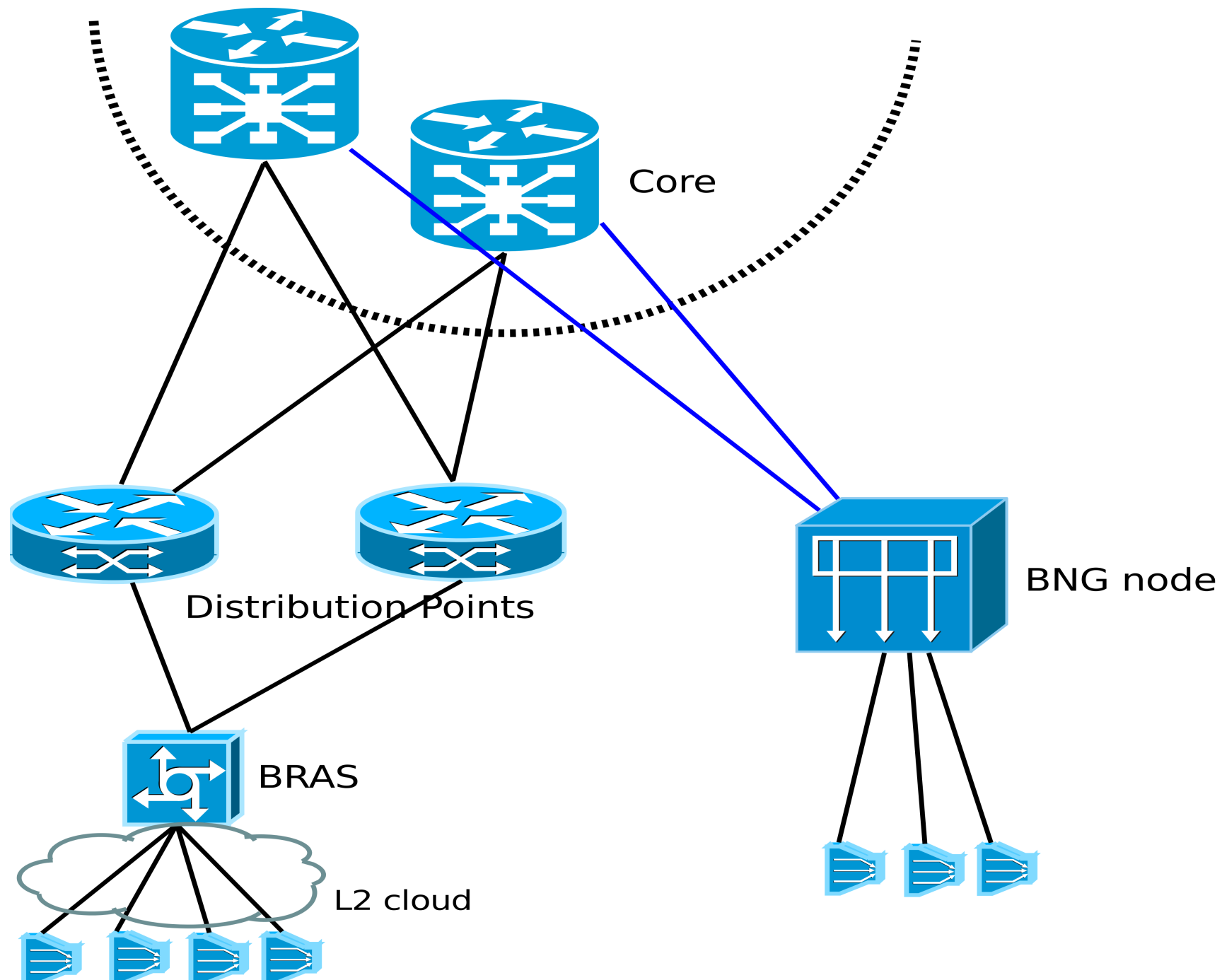
The Core (schema)



Distribution / Access

- Distribution nodes aggregate access nodes
- Access nodes mostly terminate PPP sessions of users, but also contain leased line business customers
- Each access node has high availability backhaul connections to at least 2 distribution nodes
- Major project underway to simplify / consolidate access - distribution layers (BNG instead of BRAS nodes)

Distribution / Access example



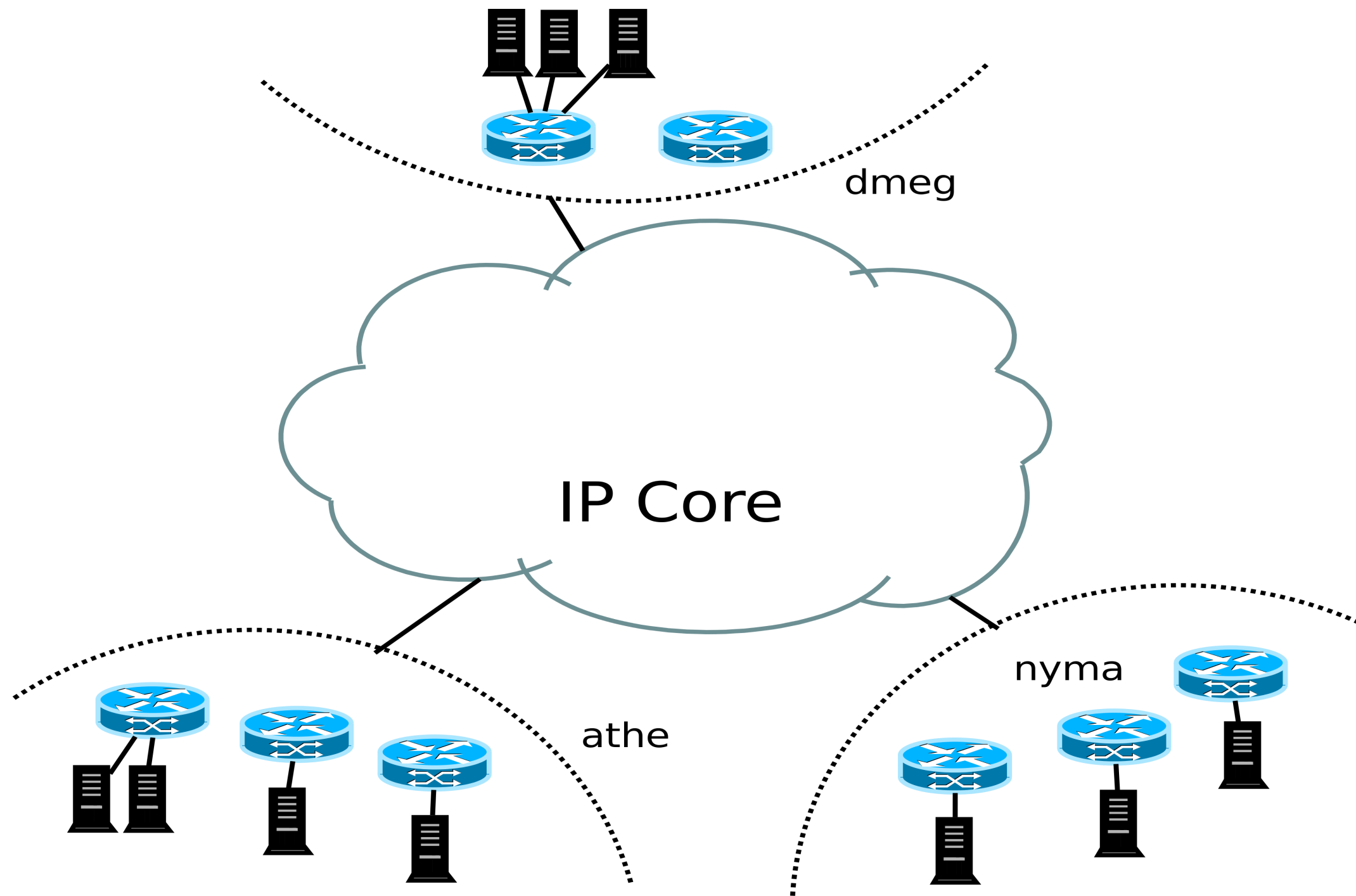
DNS anycast nodes (revisited)

- With the above information on the topology the DNS anycast nodes were placed in few selected locations (POPs) at the distribution layer
- Too many locations at the access layer
- Not all distribution layer POPs suitable to host server equipment
- Easier access to nodes for the relevant operation teams an extra criterion

Final setup

- Cluster of 10 machines in 3 geo-locations
- 2 locations in downtown Athens and 1 in the company headquarters
- 2 machines used for infrastructure resolving (in 2 locations) having DNSSEC validation enabled
- 8 machines used for end-users resolving (were 5+5 in the legacy setup)

Final setup (schema)



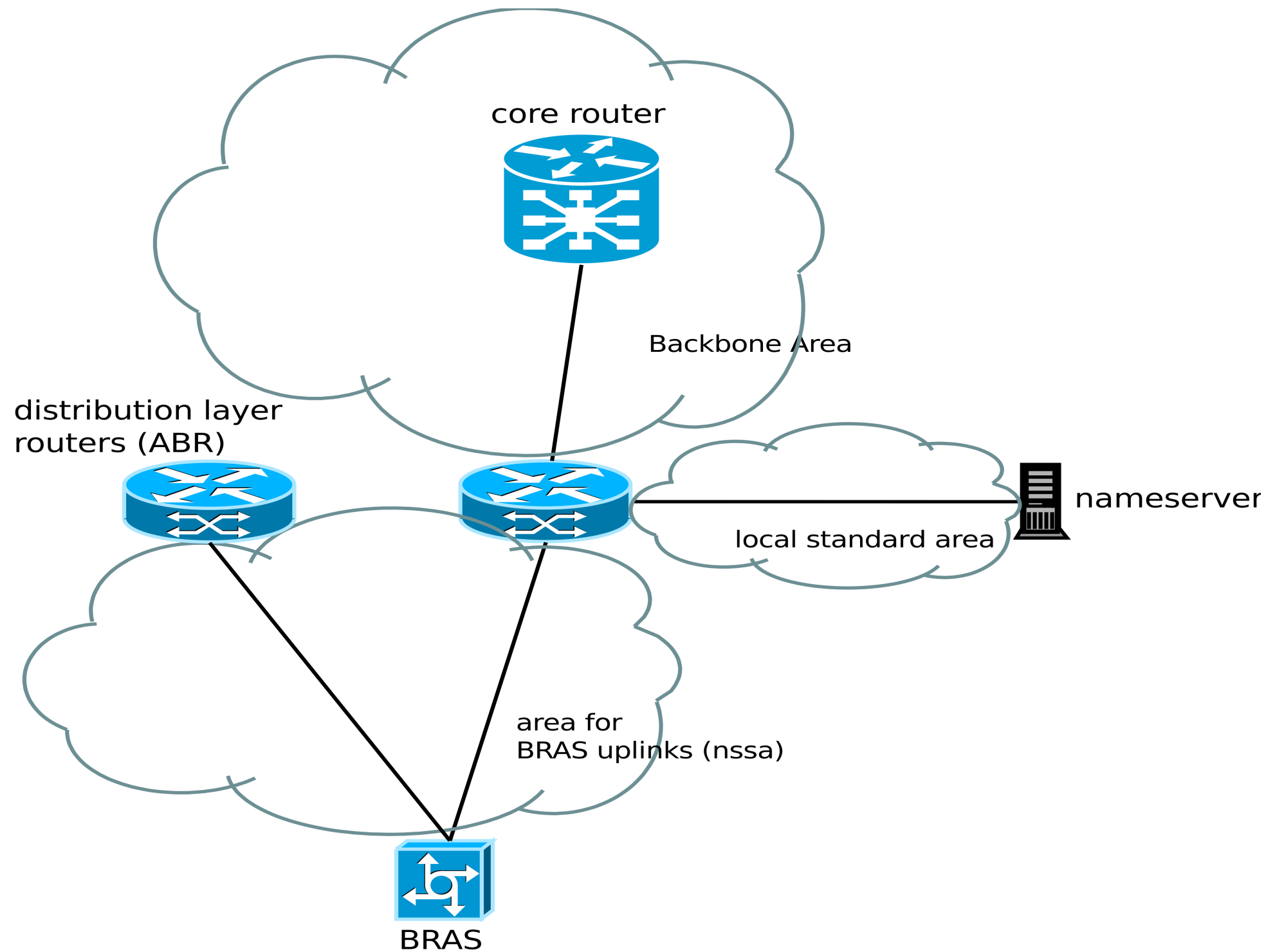
Software choices

- Linux CentOS 6.x operating system
- Quagga ospfd to announce the /32 anycast IP(s)
- Iptables on each node (host based security)
- BIND 9.9 as nameserver
- No big departure from the legacy setup to also accommodate the operation teams and existing tools

Configuration details

- bind runs chrooted
- /24 range for p-t-p links of servers split to /30s
- ospfd timers tuned (fast hello support, minimal failover to another node)
- OSPF routes not injected in server's routing table (default route is enough)
- IPtables rule protection, allow measurement services (snmp, bind) and ssh access from specific management LANs
- Administration of servers and console access via the central management LANs

OSPF topology



Quagga config

```
! Zebra configuration saved from vty
!   2012/10/17 13:50:59
!
hostname <nodename>.otenet.gr
password <vtypwd>
enable password <enpwd>
log syslog
!
interface em4
    ip ospf network point-to-point
    ip ospf authentication message-digest
    ip ospf message-digest-key 1 md5 <pwd>
    ip ospf dead-interval minimal hello-
multiplier 3
    ip ospf priority 0
!
! (continued next...)
```

```
! (...continued)
interface lo
    ip ospf cost 10
    ip ospf priority 0
!
router ospf
    ospf router-id <server IP>
    log-adjacency-changes detail
    passive-interface lo
    network <anycast IP>/32 area X.X.X.X
    network <server IP>/30 area X.X.X.X
!
access-list MATCHALL permit any
route-map IMPORT deny 10
    match ip address MATCHALL
!
line vty
```



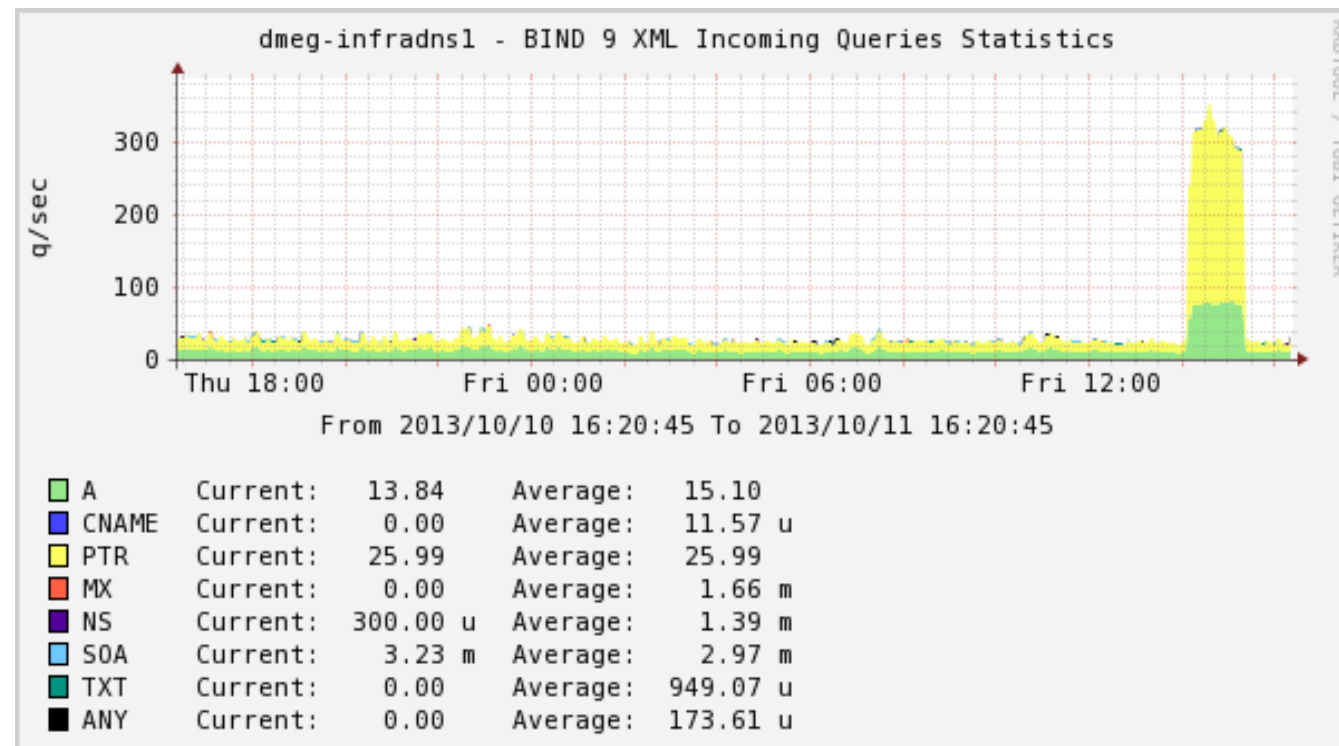
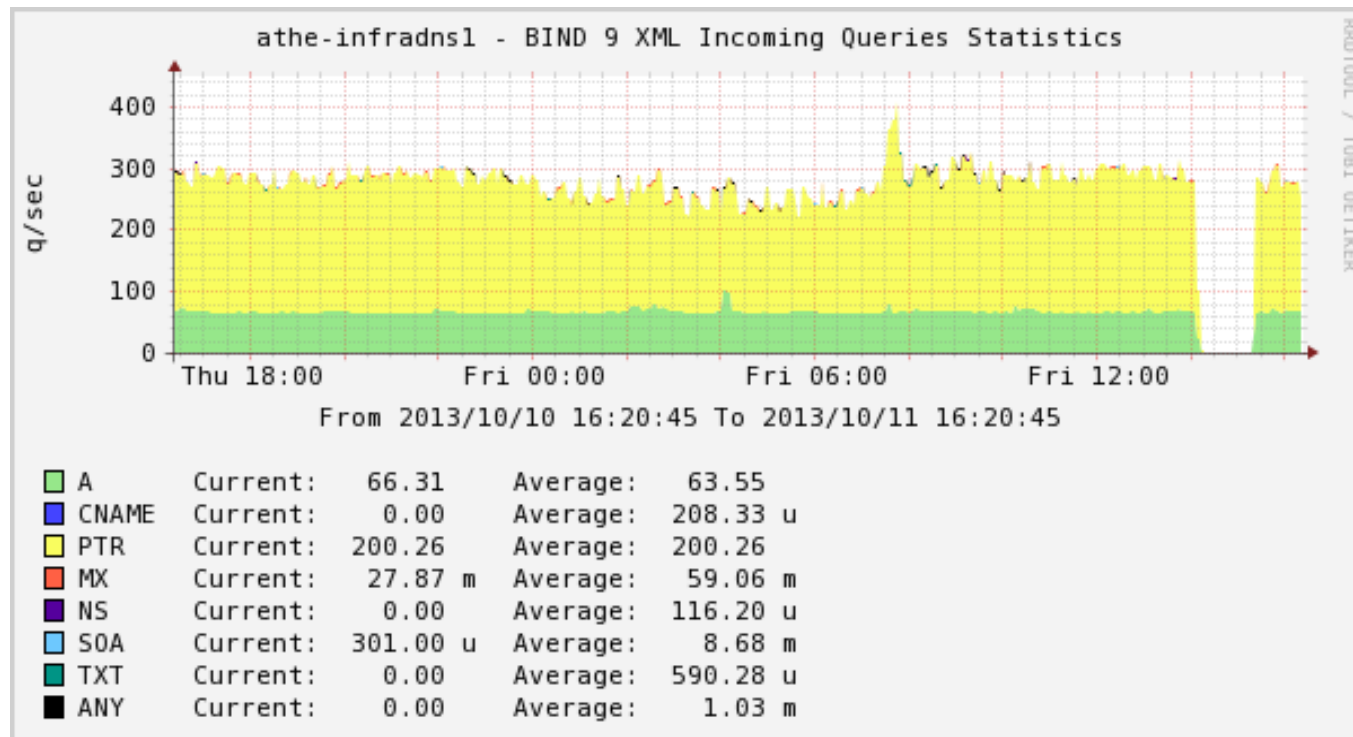
Node failover

- With the aforementioned timers failover time is a few seconds (terminating the ospf server process or shutting the link)

```
[kzorba@server ~]# while [ 1 ] ; do dig +short @<anycastIP> id.server  
txt chaos; sleep 1; done  
"athe-node1"  
"athe-node1"  
"athe-node1"  
<athe-node1 ospf process down>  
"athe-node1"  
"athe-node1"  
"athe-node1"  
"athe-node1"  
"dmeg-node1"  
"dmeg-node1"  
...
```



Node failover graphs

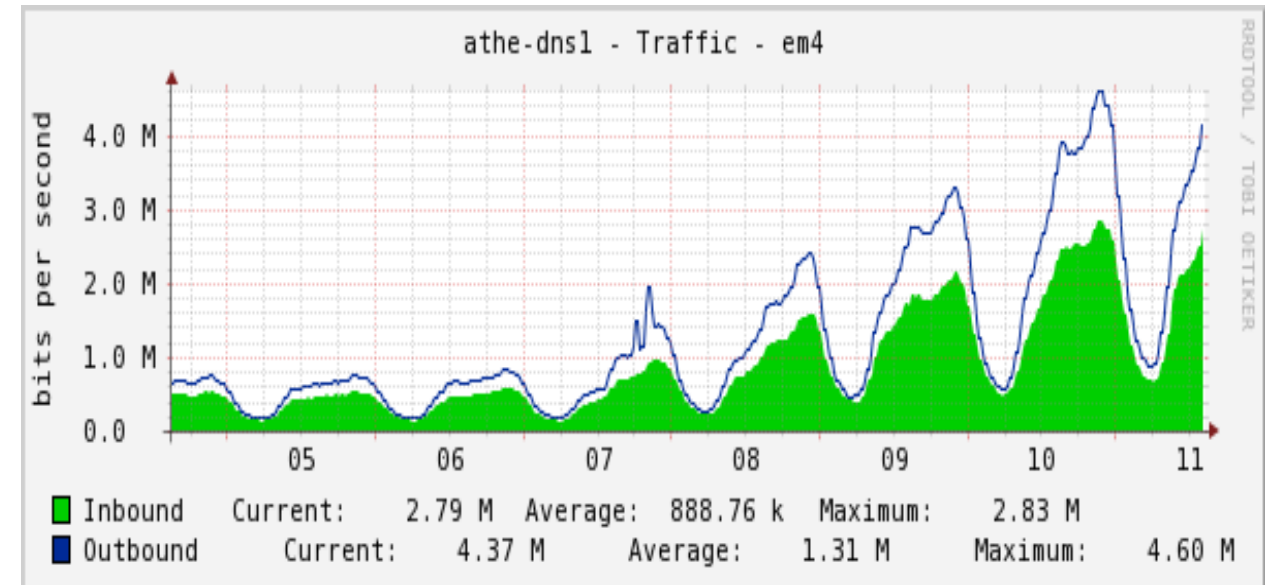
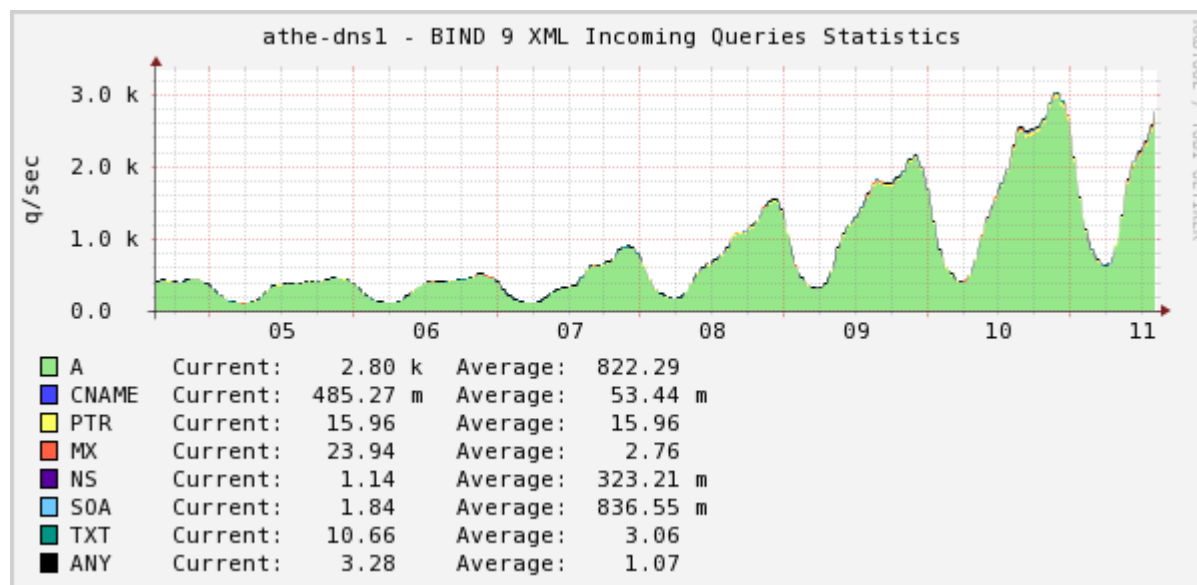
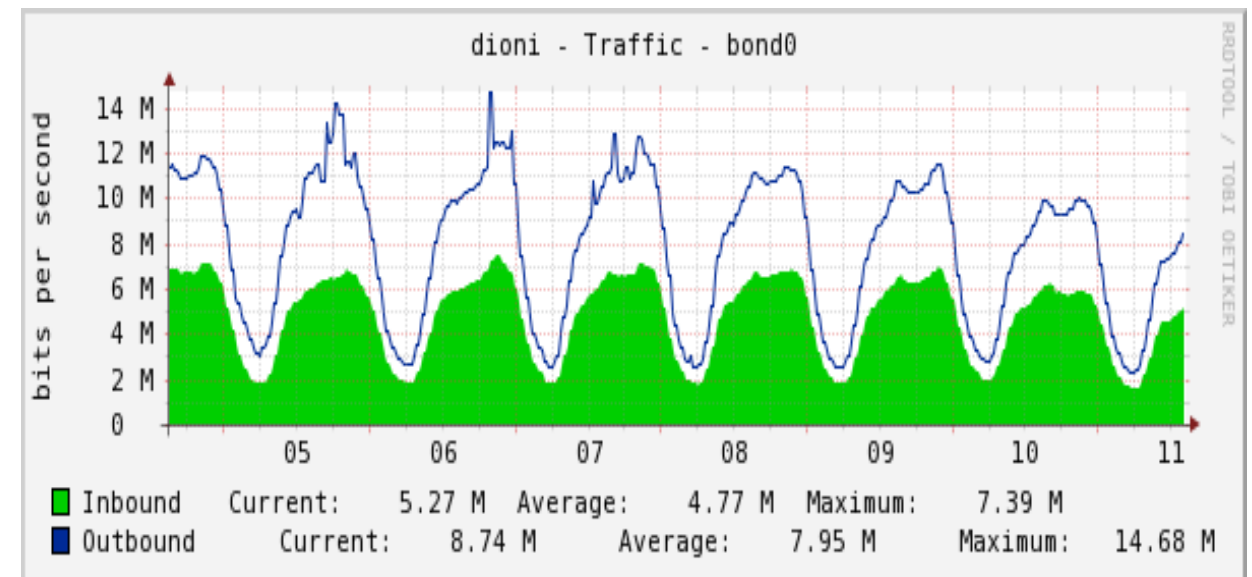
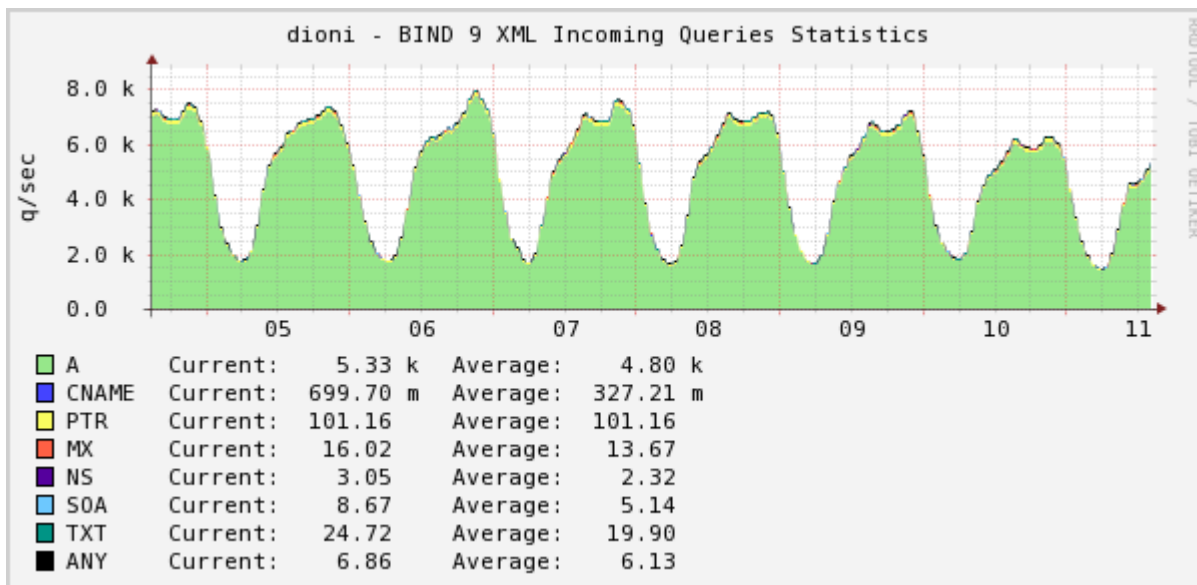


Migration of end users

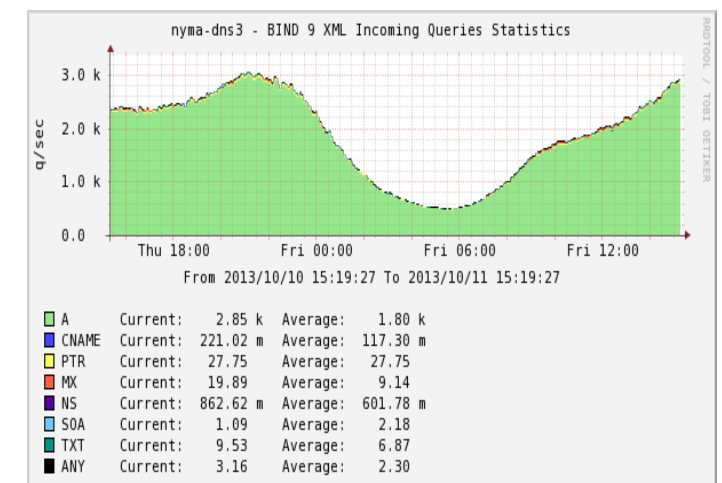
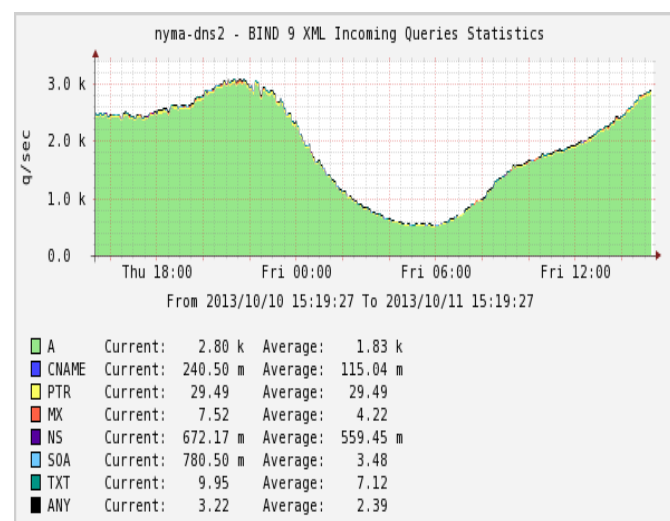
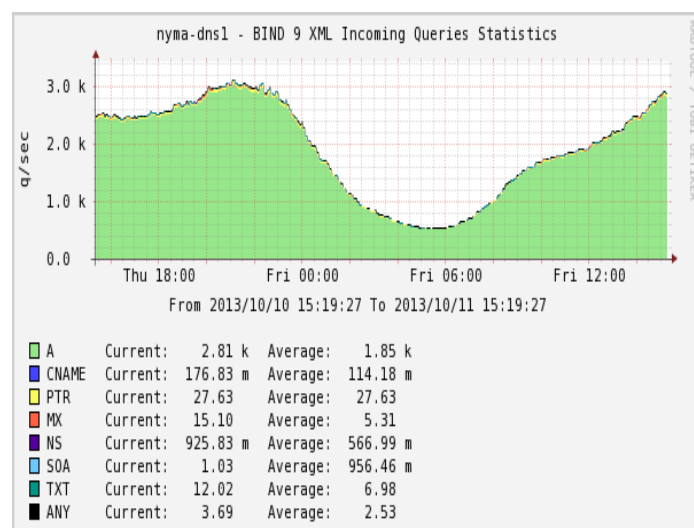
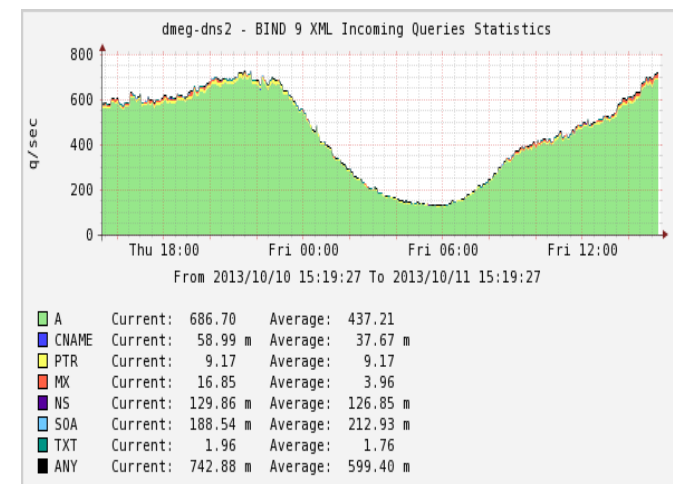
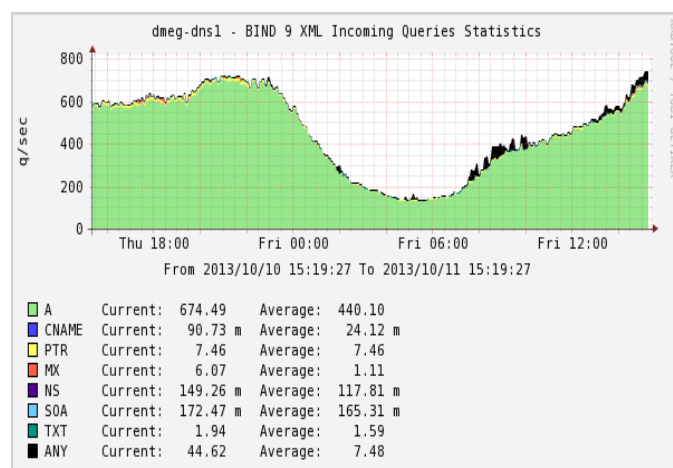
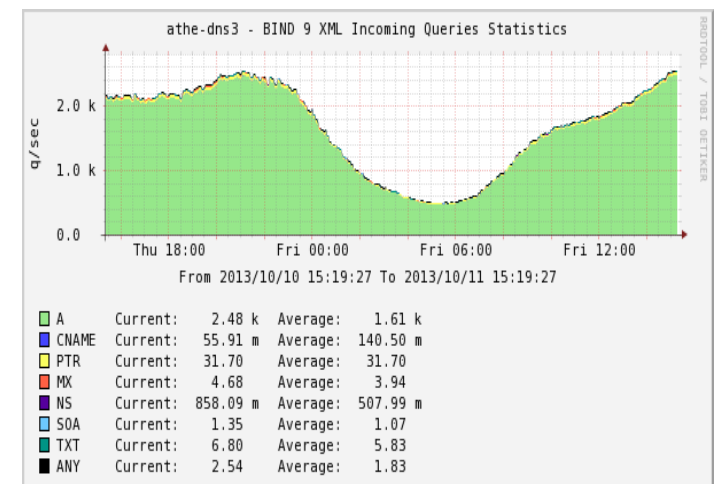
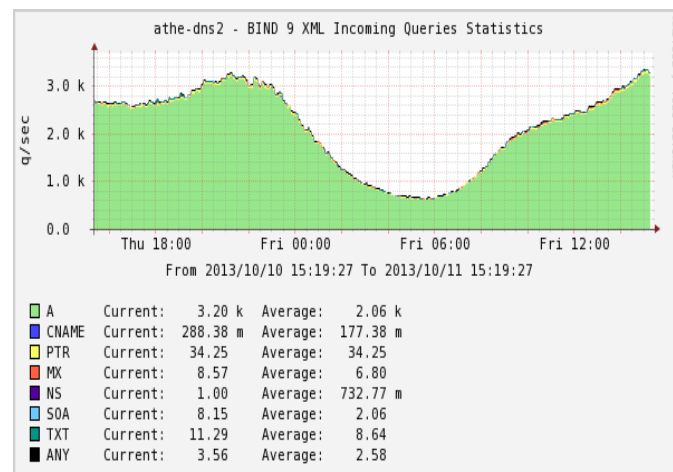
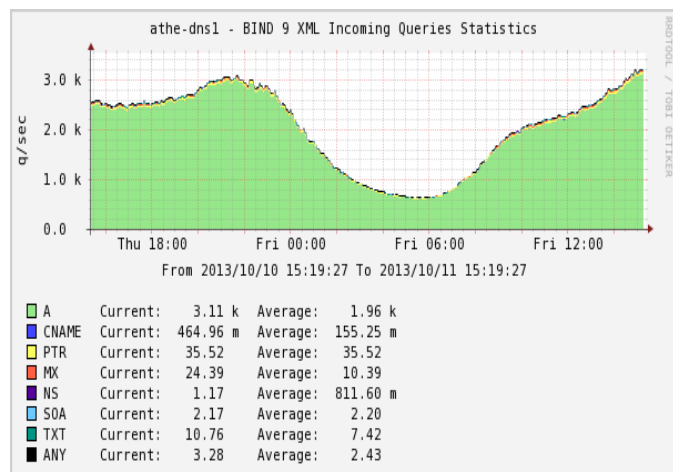
- Gradual migration by changing the PPP DNS settings on terminating routers (BRAS and BNG nodes)
- Most BRAS nodes have switched, with the PPP reconnection each user is directed to the anycast resolving cluster
- Still in progress as of this RIPE Meeting
- Smooth process so far
- Old resolver IPs will be announced in OSPF via the cluster to complete the migration



User migration graphs



Load distribution



Immediate next steps

- Dual stack the service
- IPv6 connectivity / addressing is now in place
- However, the IGP choice for IPv6 was IS-IS
- No mature open source implementation of IS-IS yet (we are willing to test)
- The plan is to proceed with static routes and tracking options (IOS IP SLAs)

Administration/monitoring/alerting

- Administration of all nodes through central management hosts on trusted LANs
- Each server has IPMI module accessible through the same management LANs
- Monitoring / fault detection / alerting provided via a Nagios infrastructure already in place
- DNS measurements from BIND using cacti plugin
- Cacti / rrdtool for graph representations

Extra considerations

- A major concern is overwhelming specific nodes in the cluster
- As a first step we have addressed it via over-provisioning
- What happens in case of cascading effects in the cluster (e.g. attack specific locations)?
- Any other challenges?

Future work

- Current set of tools adequate for operation and basic understanding of what is going on
- We can do *much* better in terms of measurements and visualization
- Real or near-real time visualization and analysis of DNS queries
- Anomaly detection and trend analysis

A few ideas and experimentation

- Receive netflow information from anycast nodes
- Collect and display DNS query data (DSC tool already available but a lot of room for improvement)
- Visualization through web interface with contemporary javascript graphing libraries (e.g d3.js)
- Storage and analysis of the data a “big data” problem
- Administrator / operator actions can be incorporated in the web interface
- Different analysis levels for different groups of people

Questions?

