

BIRD

Internet Routing Daemon



CZ.NIC z. s. p. o.

Ondřej Filip / ondrej.filip@nic.cz

Oct 9, 2009 – RIPE 59 / Lisbon

Project history

- Project started in 1999
- Seminar project – Charles University
– Prague
- Project slept for a while
- Small reincarnation in 2003 and 2006
- Project fully renewed since Q3 2008 – part of CZ.NIC Labs



Project goals

- Opensource routing daemon – alternative to Quagga/Zebra
- Fast and efficient
- Portable, modular
- Support current routing protocols
- IPv6 and IPv4 in one source code

Features



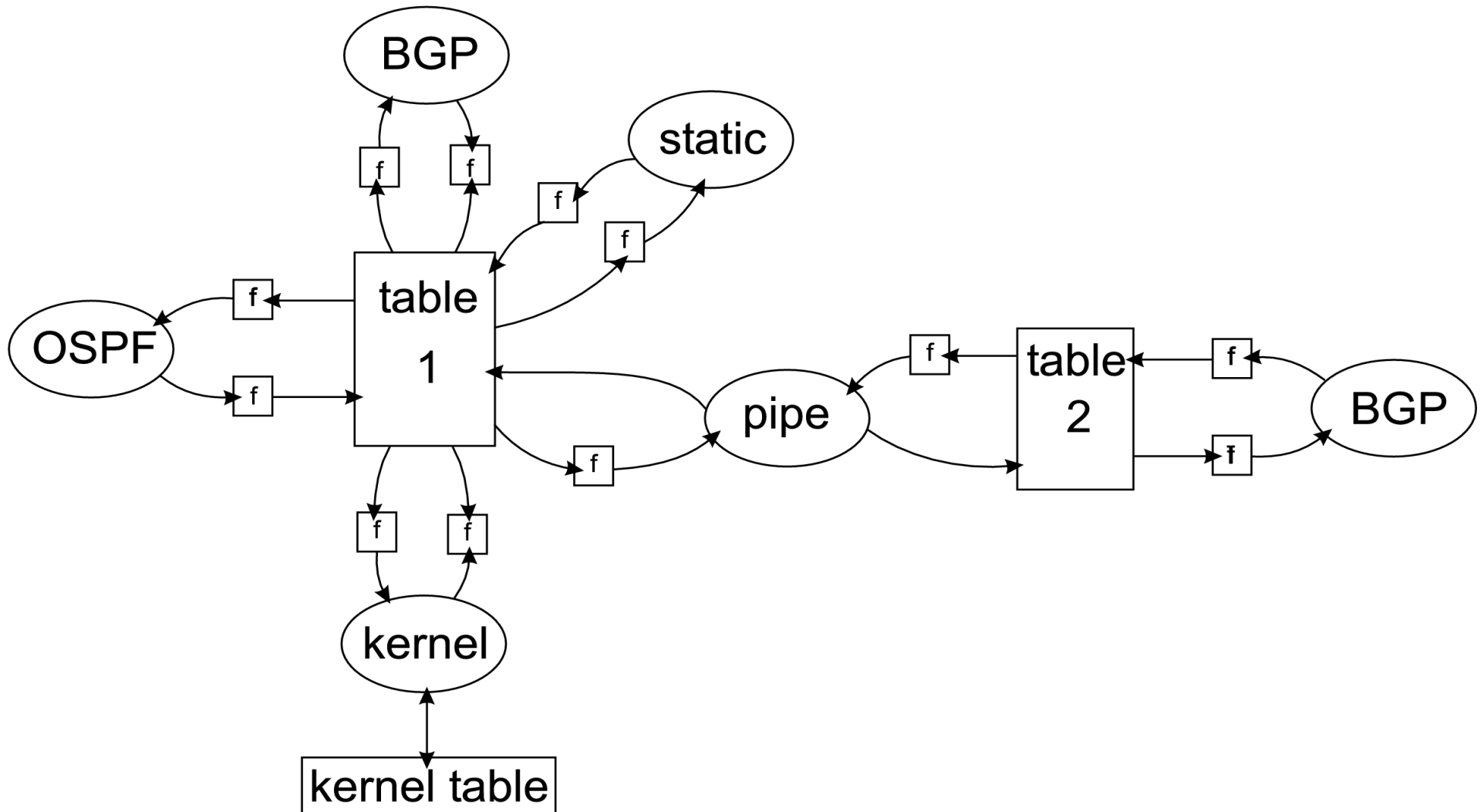
- Portable – Linux, FreeBSD, NetBSD, OpenBSD
- IPv4/IPv6 support
- Static routing
- RIP, RIPv2, RIPvng
- OSPFv2, (OSPFv3 in alpha version, to be finished by the end of 2009)
- BGP (v4 and v6), route reflector
- Route server
- ASN32 (ASPLAIN), MD5

Features



- Multiple routing table - RIBs (internal and also synchronization with OS)
- Protocol PIPE
- Multiple routers, route reflectors on a single system
- Powerful configuration
- Very powerful filtering language
- Command line interface (show, restart, ...)
- Automatic reconfiguration

Design



Configuration sample

```
log "/var/log/bird.log" all;

router id 193.51.100.238;

protocol static {
    route 10.0.0.0/8 drop;
    route 172.16.0.0/12 drop;
    route 192.168.0.0/16 drop;
}
filter bgp_out {
    if (net = 192.175.48.0/24 ) &&
        (source = RTS_DEVICE) then accept;
    else reject;
}
protocol bgp NIX_1 {
    local as 112;
    neighbor 193.51.100.235 as 6981;
    import all;
    export filter bgp_out;
}
```

CLI example



```
bird> show protocols
name      proto    table    state  since  info
direct1   Direct  master   up     Apr11
kernel1   Kernel  master   up     Apr11
device1   Device  master   up     Apr11
static1   Static  master   up     Apr11
NIX_2     BGP     master   up     Apr11  Established
NIX_1     BGP     master   up     Apr25  Established
ospf1     OSPF    master   up     Apr11  Running
bird>
bird> show status
BIRD 1.1.3
Current server time is 06-08-2009 22:01:06
Last reboot on 11-07-2009 22:54:12
Last reconfiguration on 30-07-2009 06:25:25
Daemon is up and running
bird>
```


CLI example (cont)



```
bird> show route
10.0.0.0/8      via 200.30.10.3 on eth2 [ospf1 13:10] E2 (150/5/1000)
127.0.0.0/8    dev lo [direct1 13:09] (240)
200.30.20.0/24 via 200.30.10.3 on eth2 [ospf1 13:10] I (150/10)
200.30.10.0/24 dev eth2 [direct1 13:09] (240)
                dev eth2 [ospf1 13:10] I (150/5)
200.0.10.0/24  dev eth0 [direct1 13:09] (240)
                dev eth0 [ospf1 13:09] I (150/5)
172.16.0.0/16  via 200.30.10.3 on eth2 [ospf1 13:10] E2 (150/5/1000)
195.47.235.0/24 via 194.50.100.246 on eth1 [NIX2 Apr11] (100)[AS688i]
                via 194.50.100.245 on eth1 [NIX1 Apr25] (100)[AS688i]

bird>
bird> show route protocol ospf1
10.0.0.0/8      via 200.30.10.3 on eth2 [ospf1 13:10] E2 (150/5/1000)
200.30.20.0/24  via 200.30.10.3 on eth2 [ospf1 13:10] I (150/10)
200.30.10.0/24  dev eth2 [ospf1 13:10] I (150/5)
200.0.10.0/24   dev eth0 [ospf1 13:09] I (150/5)
172.16.0.0/16   via 200.30.10.3 on eth2 [ospf1 13:10] E2 (150/5/1000)
```

CLI example (cont)



```
bird> show route for 127.0.0.1
127.0.0.0/8          dev lo [direct1 13:09] (240)
```

```
bird> show route filter bgp_out
192.175.48.0/24     dev dummy0 [direct1 Apr1] (240)
```

```
bird> show route count
1469 of 1469 routes for 849 networks
```

```
bird> show route export NIX_1
192.175.48.0/24     dev dummy0 [direct1 Apr1] (240)
```

```
bird> show route where 127.0.0.5 ~ net
0.0.0.0/0           via 195.47.235.1 on eth0 [static1 Apr1](200)
127.0.0.0/8         dev lo [direct1 Apr1] (240)
```

```
bird> show route filter {if 127.0.0.5 ~ net then accept;}
0.0.0.0/0           via 195.47.235.1 on eth0 [static1 Apr1](200)
127.0.0.0/8         dev lo [direct1 Apr1] (240)
```

Filter example – router server

- Route server policy - NIX.CZ

Evaluation order	Community	Action
1	0:<peer-as>	Do not advertise to <peer-as>
2	47200:<peer-as>	Advertise to <peer-as>
3	0:47200	Do not advertise to any peer
4	47200:47200	Advertise to all peers

Filters example

```
define myas = 47200;
```

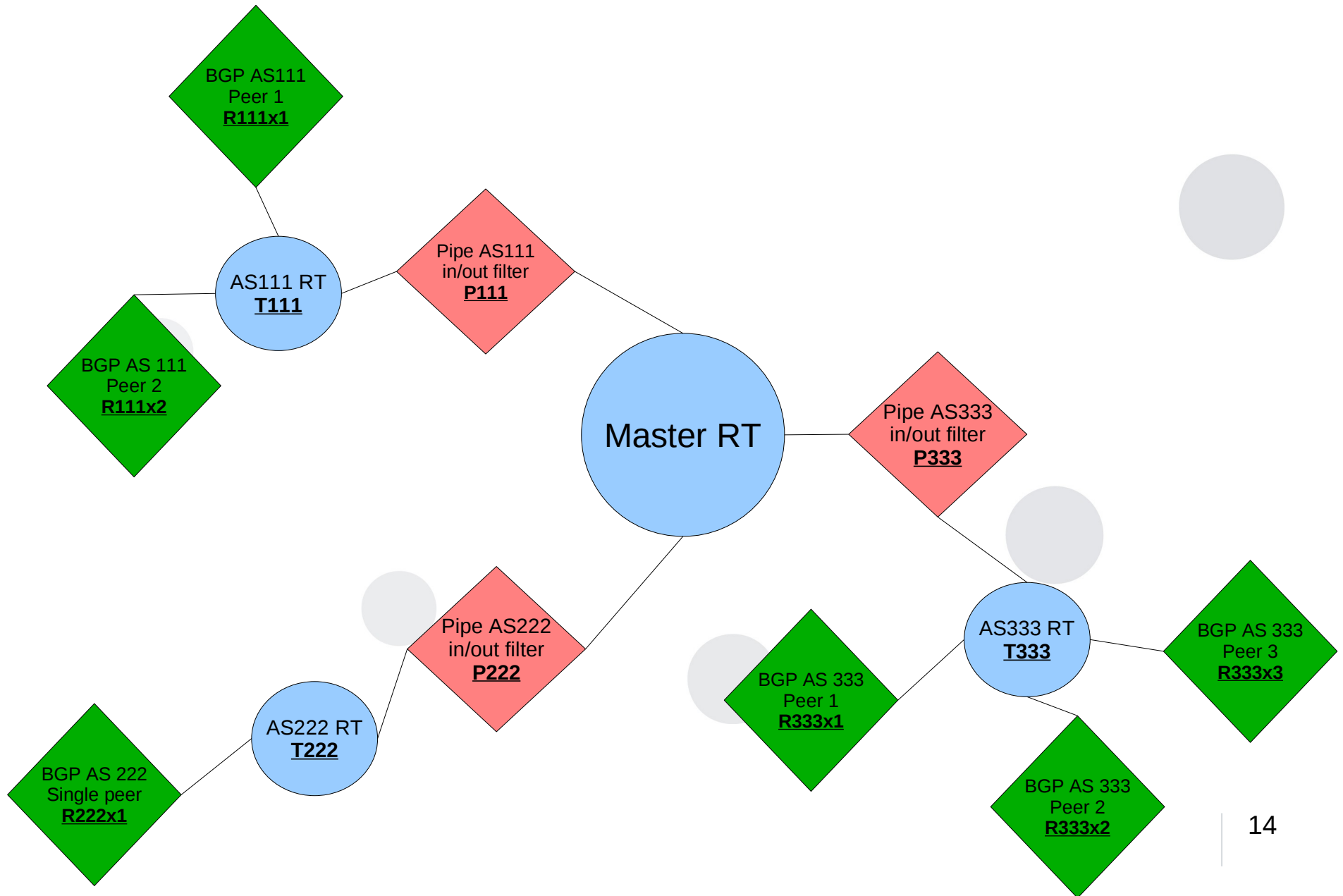
```
function bgp_out(int peeras)
{
    if ! (source = RTS_BGP ) then return false;
    if (0,peeras) ~ bgp_community then return false;
    if (myas,peeras) ~ bgp_community then return true;
    if (0, myas) ~ bgp_community then return false;
    return true;
}
```

```
protocol bgp R25192x1 {
    local as myas;
    neighbor 194.50.100.13 as 25192;
    import where bgp_in(25192);
    export where bgp_out(25192);
    rs client;
}
```

Filters example

```
function avoid_martians()  
prefix set martians;  
{  
    martians = [ 169.254.0.0/16+, 172.16.0.0/12+,  
                192.168.0.0/16+, 10.0.0.0/8+, 224.0.0.0/4+,  
                240.0.0.0/4+, 0.0.0.0/32-, 0.0.0.0/0{25,32},  
                0.0.0.0/0{0,7}  ];  
  
    # Avoid RFC1918 networks  
    if net ~ martians then return false;  
  
    return true;  
}
```

BIRD as a route server



BIRD at LoNAP

- First BIRD RS implementation
- Two route-servers
- BIRD and OpenBGPD
- BIRD in multiple RIBs setup
- Thanks for help with debugging!
- 25 sessions, 1000 prefixes

BIRD at NIX.CZ



- Also multiple RIBs setup
- BIRD on Linux and Quagga on FreeBSD
- Approx 100 IPv4 sessions (30 IPv6)
- No BIRD's crash since implementation (1.1.3 release)
- Approx 6000 IPv4 prefixes (60 IPv6)
- Memory consumption – 60MB (Quagga 260MB)



Other applications

- AS112 server at NIX.CZ
- BGP/OSPF router for smaller ISPs
- Router for some CZ.NIC's anycast nodes
- Used is small embedded system – part of firmware of some WiFi APs
- Implementation at some other IXPs in progress

BIRD vs Quagga

- Full IPv4 BGP table import ~300k routes
- Comparison made on Linux system
- Multiple measurements
- CPU and memory consumption (sec, MB)
- Test 1 – with OS routing table sync (bgpd+zebra)
- Test 2 – w/o OS routing table sync

BIRD vs Quagga

- Test 1

Daemon	Memory (MB)	CPU (sec)
Zebra	90 + 77 = 167	32 + 120 = 152
BIRD	30	14

- Test 2

Daemon	Memory (MB)	CPU (sec)
Zebra	87	30
BIRD	30	7

Future development

- Monthly releases
- OSPFv3 - end of 2009
- Improvement of OSPF (Opaque LSA, ...)
- Route flap dampening
- ...
- Depends on user demand

Summary

PROS

- Powerful configuration and filter language
- Lightweight, efficient
- Developed by people from IXP community
- Active development

CONS

- Unusual configuration language
- Some features in development – e.g. looking glass
- IPv4/IPv6 separation

Invitation to Prague

- Come to Prague, to see programmers of BIRD
- ... or to see museums, castles, churches....
- ...
- RIPE 60 at Prague,
- Hosted by CZ.NIC



¿Questions?

<http://bird.network.cz>
<ondrej.filip@nic.cz>