# BIND 9.11 Update

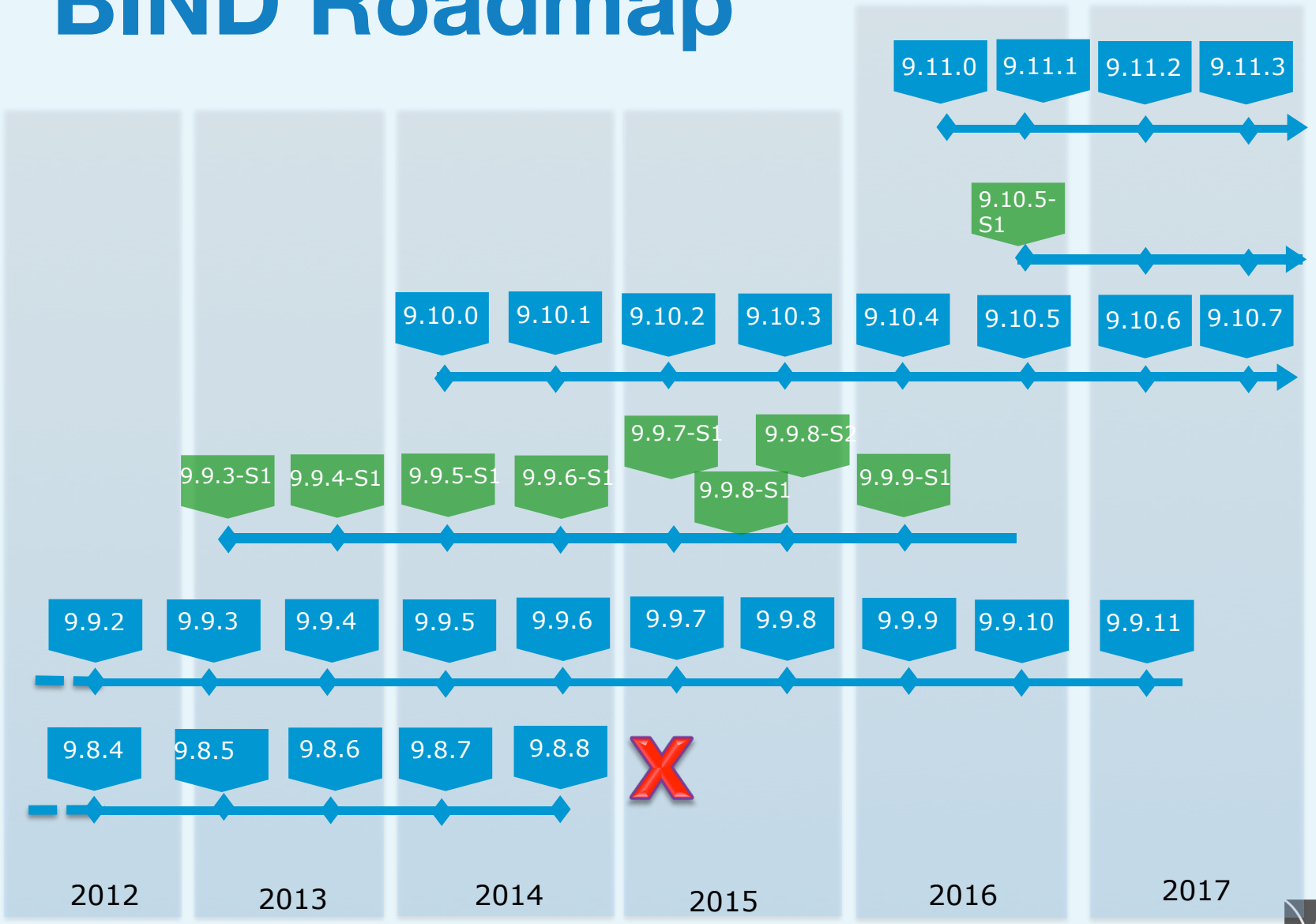
Vicky Risk, Product Manager

# Project Update

- **Recent history**
- BIND 9.11 New Features
- ISC Performance Lab
- Community Participation

# BIND team focus

# BIND Roadmap



# ‘Regular Maintenance’

- 4 Maintenance releases
- 12 Security patch releases
  - 7 cves (sorry!)
- 2 Experimental releases
- 5 –S edition releases
- Resolved 486 “issues” (bugs + feature requests)

# BIND Core Team

sw eng, team lead

Evan H.

sr sw eng

Mark A.

sw eng

Mukund S.

sw eng

Witold K.

director of sweng

Stephen M.

qa manager

Jeremy R.

qa engineer

Curtis B.

research fellow

Ray Bellis

# New in 2015

sw eng, team lead

Evan H.

sr sw eng

Mark A.

sw eng

Mukund S.

sw eng

Witold K.

director of sweng

Stephen M.

qa manager

Jeremy R.

qa engineer

Curtis B.

research fellow

Ray Bellis

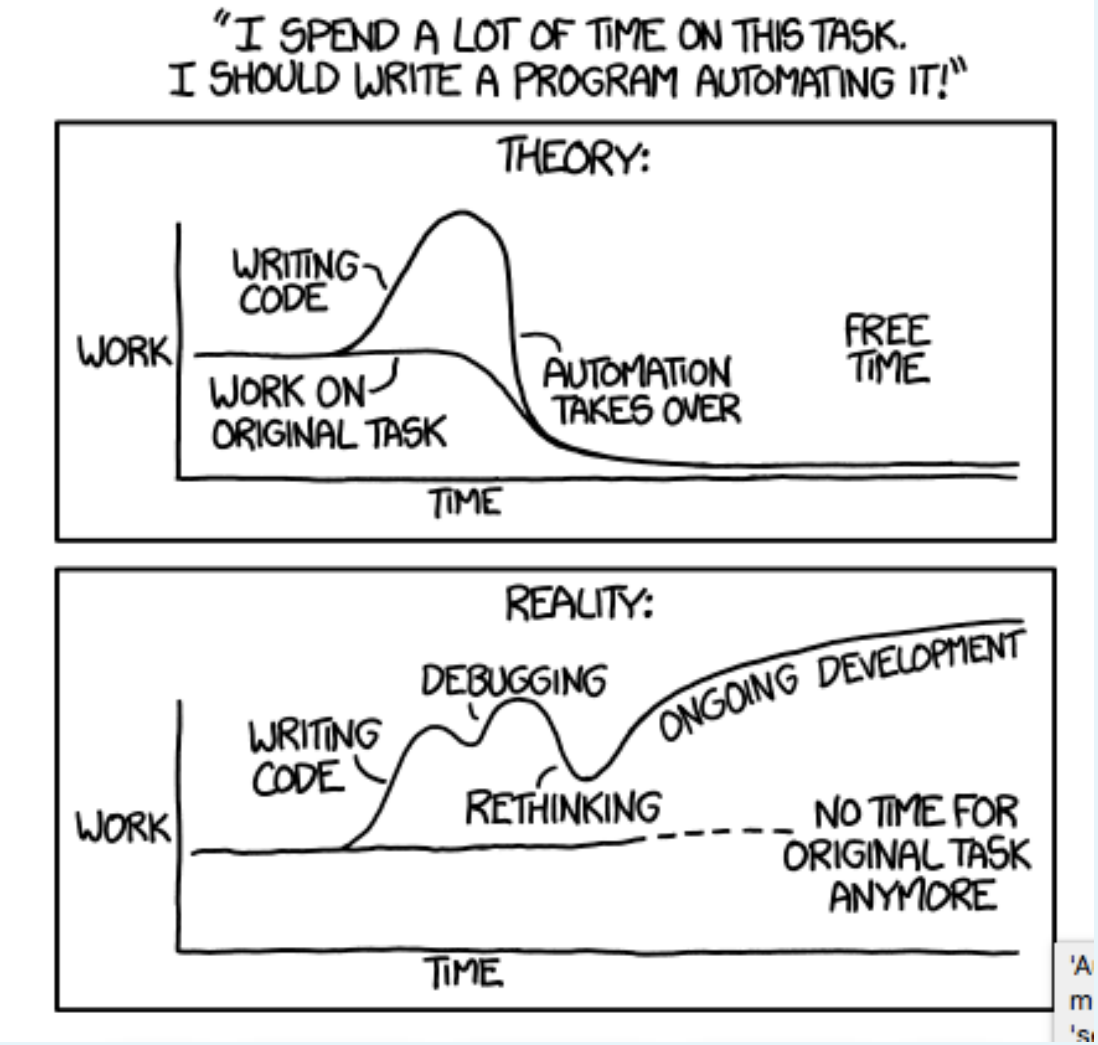
# 9.11 feature decisions

- 2014-2015 we focussed on Resolver DDOS mitigation
  - hadn't done much lately for authoritative users
- Provisioning feature requests from large operators and the OpenStack project

# Provisioning Performance

- RNDC del zone
- Notify congestion

# Unmaintained Scripts



# We Wanted

- a standardized provisioning method that didn't require users to maintain scripts for updating slaves
- faster zone deletion from NZF
- faster updates/ notify rate limiting
- a fast database option
  - makes a lot of sense for an ISP or hoster

# Catalog Zone

- a new zone on the master that contains a list of zones (the catalog)
- updates to this zone are propogated to the slaves the same way updates to any other zone are propogated
- based on Paul Vixie's MetaZones proposal from 2004

# Catalog Zones: Adding a new zone

**Today**

1. add the zone to the master
2. connect to each slave, add the new zone
   - ① add to the next slave
   - ② another slave
   - ③ yet another slave
   - ④ yet another slave
   - ⑤ yet another slave
   - ⑥ yet another slave
   - ⑦ yet another slave
   - ⑧ yet another slave

**With Catalog Zones**

1. add the zone to the master
   - ① add the zone to the catalog zone on the master

# Configuration

## Master

```
options {
        listen-on {
        10.53.0.1;
        };
        allow-new-zones yes;
};

zone “catz.isc.org” {
        type master;
        file “catz.isc.org.db”;
        allow-transfer {
        10.53.0.2;
        };
};
```

## Slave

```
options {
        listen-on {
        10.53.0.2;
        };
        allow-new-zones yes;
        catalog-zones {
        zone “catz.isc.org”;
        };
};
zone “catz.isc.org” {
        type slave;
        masters {
        10.53.0.1 }
        };
};
```

# Catalog zones work with

- Views (you can have different CZs for different views)
- DNSSEC. Zones can be In-line signed on the master before transferring.
- Multiple tiers Master -> transfer master -> slave

# CZ DONT work with

- RPZ zones cannot be in catalog zones
- Catalog zones cannot be in catalog zones

# Zone options supported

- master
- allow-update
- allow-transfer
- keys
- allow-query

# CZ looks like a winner

- Tested with 100,000 zones in one catalog, will test more
- Deleting a zone from a catalog zone is *much faster* than if the zone is in a NewZoneFile (NZF)
- Please beta test and give us feedback!

# IETF Draft

DNS catalog zones
draft-muks-dnsop-dns-catalog-zones-00

current status = “expired”
will be updated before the next IETF
hoping for other implementations

# dyndb api

- Developed by Petr Spacek and Adam Tkac for RedHat's FreeIPA (LDAP)
- Uses BIND RBT - performance is ~ 95% of 'native' zone files!
- much faster than DLZ, works with DNSSEC
- We are hoping for contributions of other backends, such as LMDB or Cassandra

https://fedorahosted.org/bind-dyndb-ldap/

# RNDC

*its either a security vulnerability, or our primary remote management api*

- RNDC –r (result code: ... EXISTS)
- RNDC showzone
- RNDC (read only mode)

# New in BIND 9.11

- Catalog zones
- dyndb api (Petr Spacek, RedHat)
- RNDC showzone, mod zone, view-only mode
- dnstap logging (Robert Edmonds)
- Performance improvements
- EDNS Client-subnet (auth)
- dig EDNS test updates

- **DNSSEC key maintenance**
- **CDS/CDSKEY auto generation**
- **Negative Trust Anchor**
- IPv6 bias
- Cookies/RRL/stats
- Squelch duplicate servers
- Refuse any (Tony Finch)
- RSSAC02 stats

# dnssec-keymgr

- python script intended to be scheduled in a cron job
- reads a policy definition file (default: /etc/dnssec.policy) and creates or updates DNSSEC keys to ensure that a zone's keys match the policy for that zone.
- New keys are created when necessary
- Existing keys' timing metadata is adjusted as needed to set the correct rollover, etc.
- If the policy changes, all applicable keys are corrected

# dnssec-keymgr

- Policy Classes
  - different profiles for zones needing higher security
- Algorithm policies (e.g. default key size for a given algorithm)
- Policy options
  - algorithm, TTL, ‘coverage’, key size, roll period, pre-publish, post-publish

*thanks to Sebastian Castro, .NZ for his help on this tool*

# IPv6 Bias

- Glue  (in 9.9.9, 9.10.4+)
  - Prefer A for IPv4 connections
  - Prefer AAAA for IPv6 connections
- SRTT adjustment
  - when enabled, default value is 50 MS
  - gives IPv6 address 50 MS advantage in selection

# New in BIND 9.11

- Catalog zones
- dyndb api (Petr Spacek, RedHat)
- RNDC showzone, mod zone, view-only mode
- dnstap logging (Robert Edmonds)
- Performance improvements
- EDNS Client-subnet (auth)
- dig EDNS test updates
- DNSSEC key maintenance
- CDS/CDSKEY auto generation
- Negative Trust Anchor
- IPv6 bias
- Cookies/RRL/stats
- Squelch duplicate servers
- Refuse any (Tony Finch)
- RSSAC02 stats

# BIND 9.11.0 Schedule

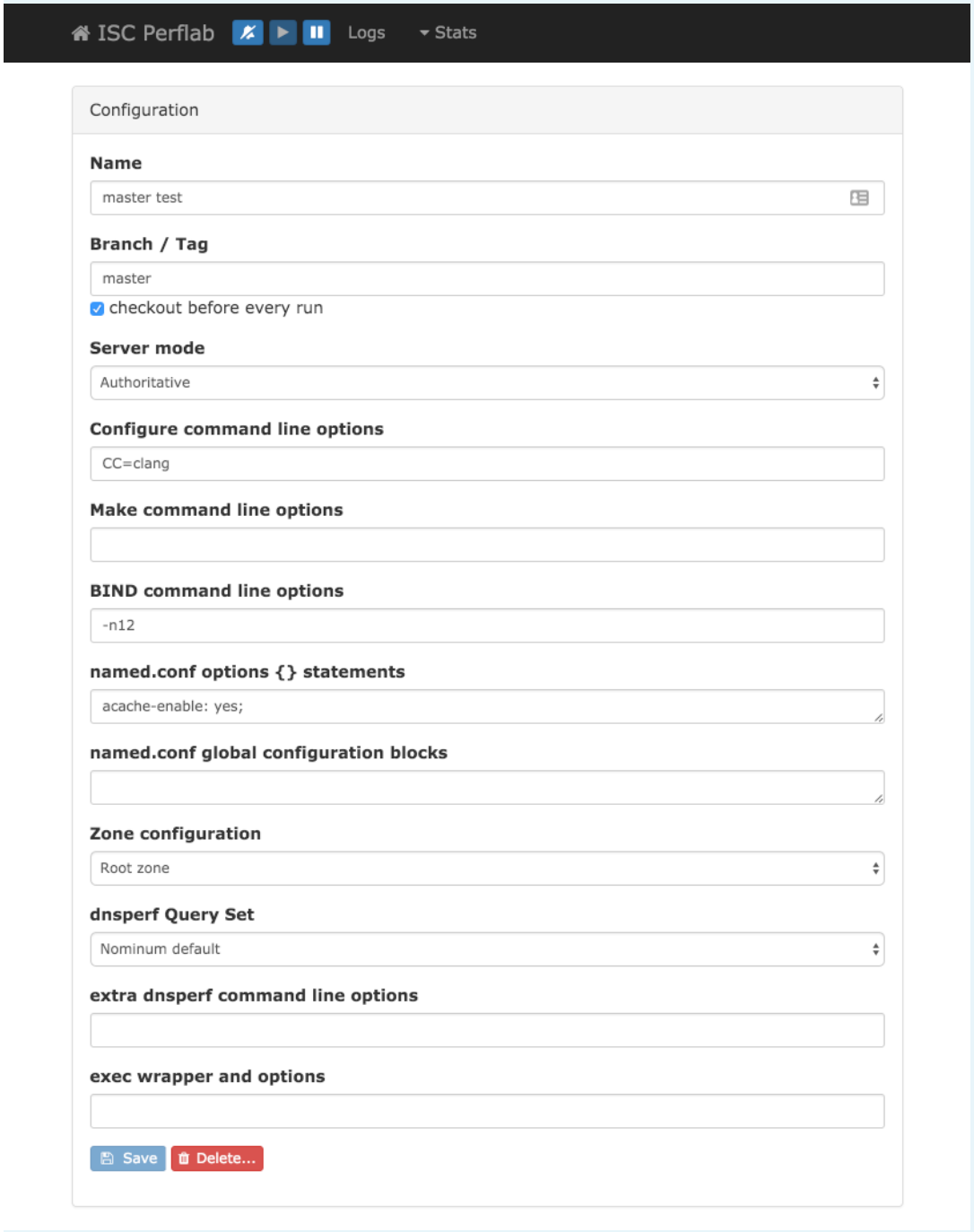
| | |
|---|---|
| ALPHA1 | 23 March |
| ALPHA2 | 25 May |
| ALPHA3 | 1 June |
| BETA | 28 June |
| RC | 26 July |
| FINAL | 2 August |

# New iOS app


- coming soon to the Apple app store
- port of the ISC Domain Information Grepper to iOS
- need beta testers
  - (contact ray@isc.org)

# Project Update

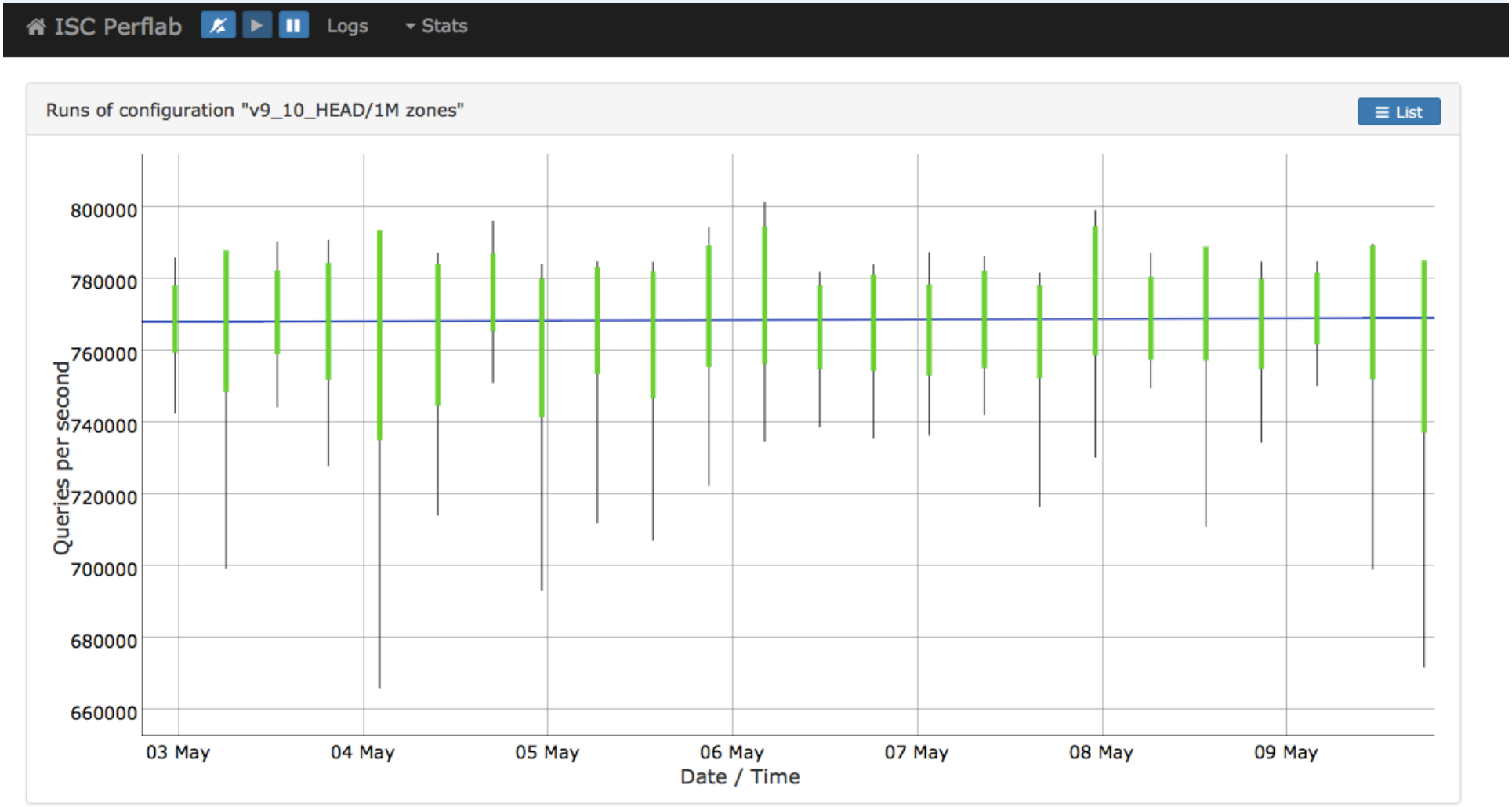
- Recent history
- BIND 9.11
- **ISC Performance Lab**
- Community participation



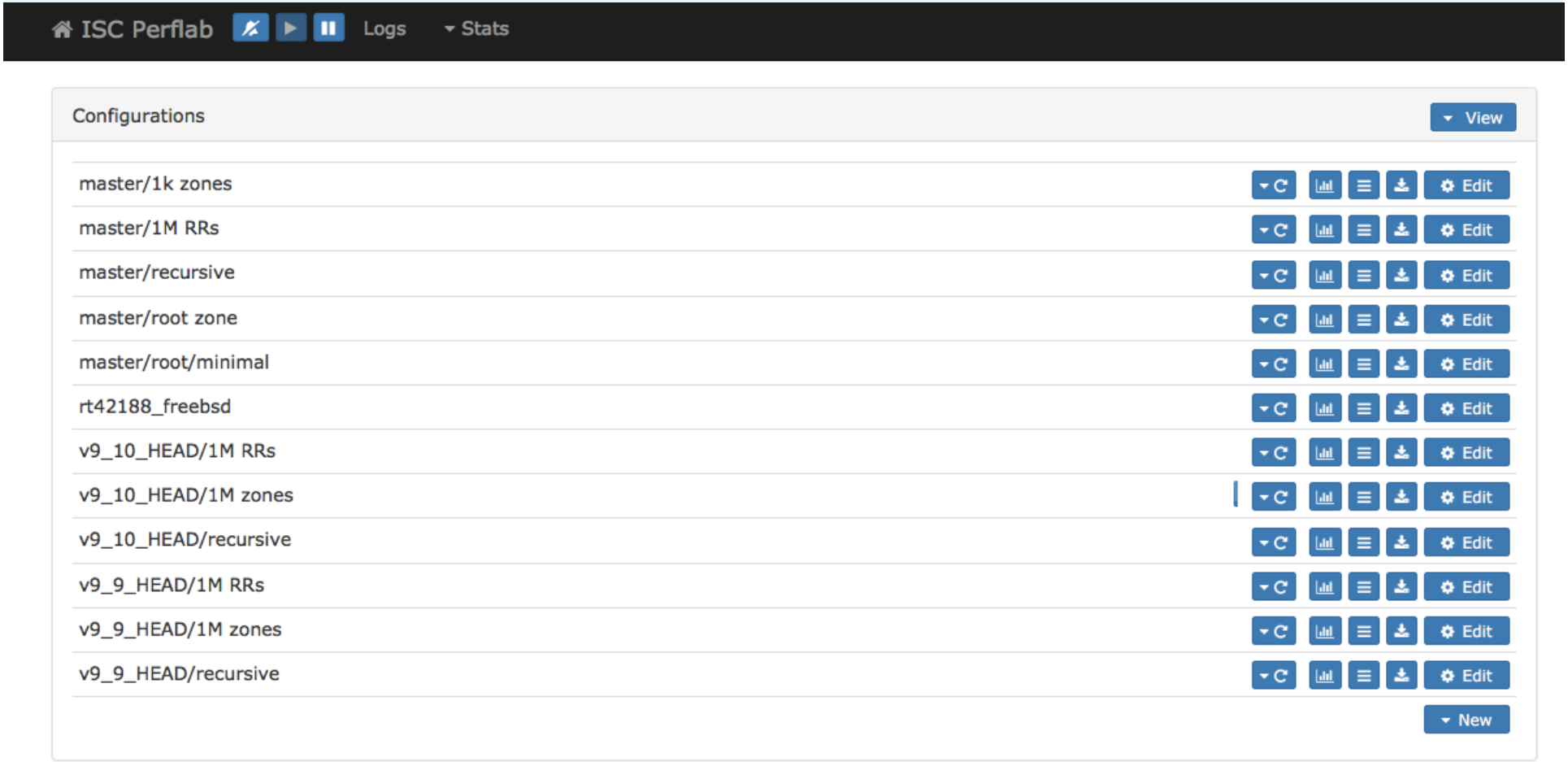

# development tool

- wrapper for build + scheduling dnsperf
- scheduled tests run continuously
- authoritative or recursive mode
- choose compile & command line options
- select zone configurations (e.g. many small zones, or fewer larger zones)
- select dnsperf query set

# Continuous Benchmarking



# Ongoing Tests




ISC Perflab
Logs
Stats
Runs of configuration "master/recursive"
650000
600000
550000
500000
450000
400000
Queries per second
modified mutex
locking
21 Feb
28 Feb
06 Mar
13 Mar
20 Mar
27 Mar
03 Apr
10 Apr
17 Apr
Date / Time

# Opportunities

- recent release vs. master
- compare major trains
- before and after bug fix
- impact of configuration options

***monitor for regressions***
***validate improvements***

# Project Update

- Recent history
- BIND 9.11
- ISC Performance Lab
- **Community participation**

# Technical Contributions

- Starting with 9.10 we have made an increased effort to respond to and accept, patches
- Tony Finch and RedHat are top contributors
  - 18 patches contributed in past 12 months, 12 accepted, 4 pending, 2 rejected
- Sebastian Castro, .NZ - dnssec-keymgr script
- The AFL tool and Hanno Böck (http://lcamtuf.coredump.cx/afl/)
- Robert Edmonds, Farsight - dnstap
- Petr Spacek, RedHat - dyndb

# Financial Contributions

~100 financial supporters

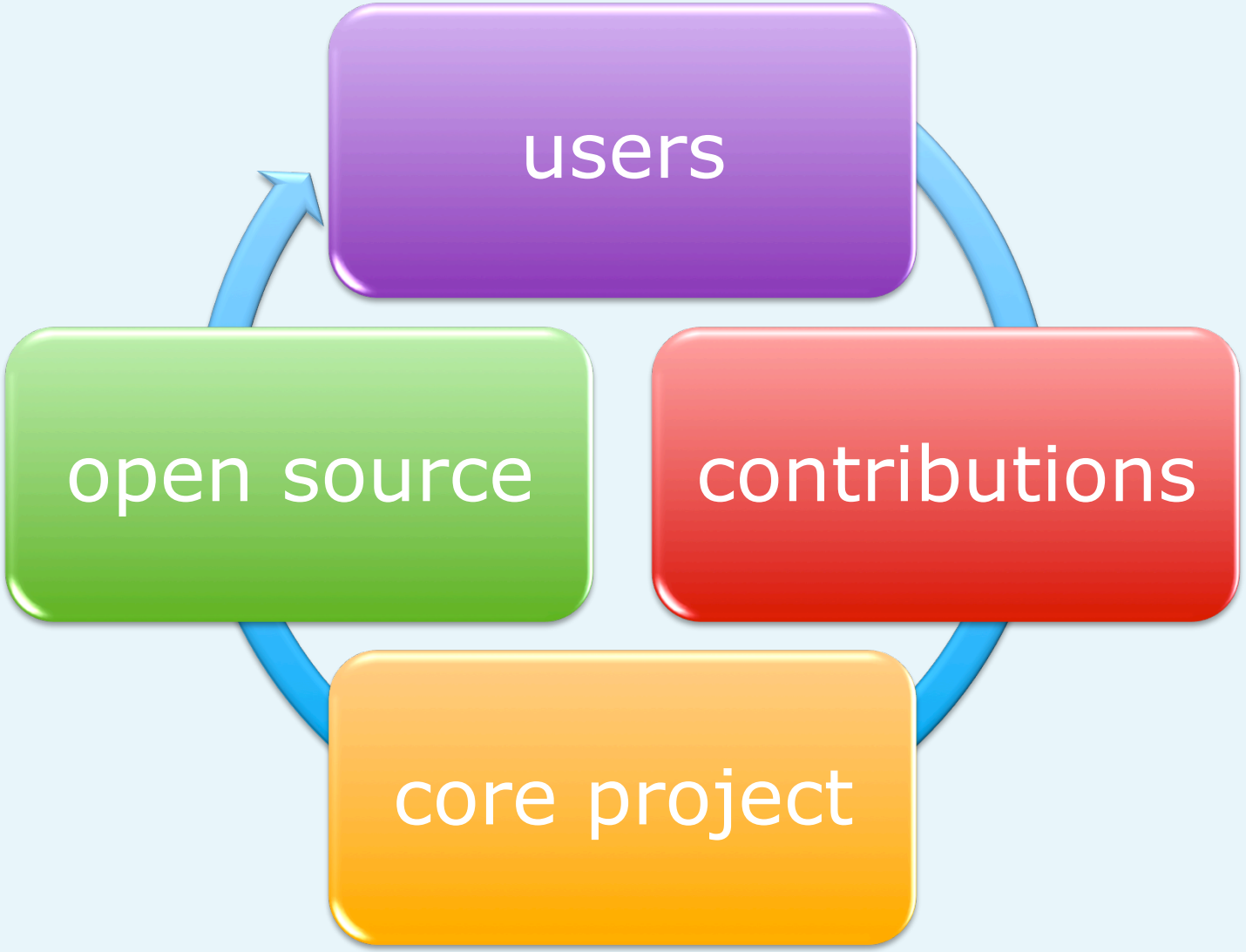
**20% of our supporters are attendees at this RIPE meeting**

**THANK YOU to these people**

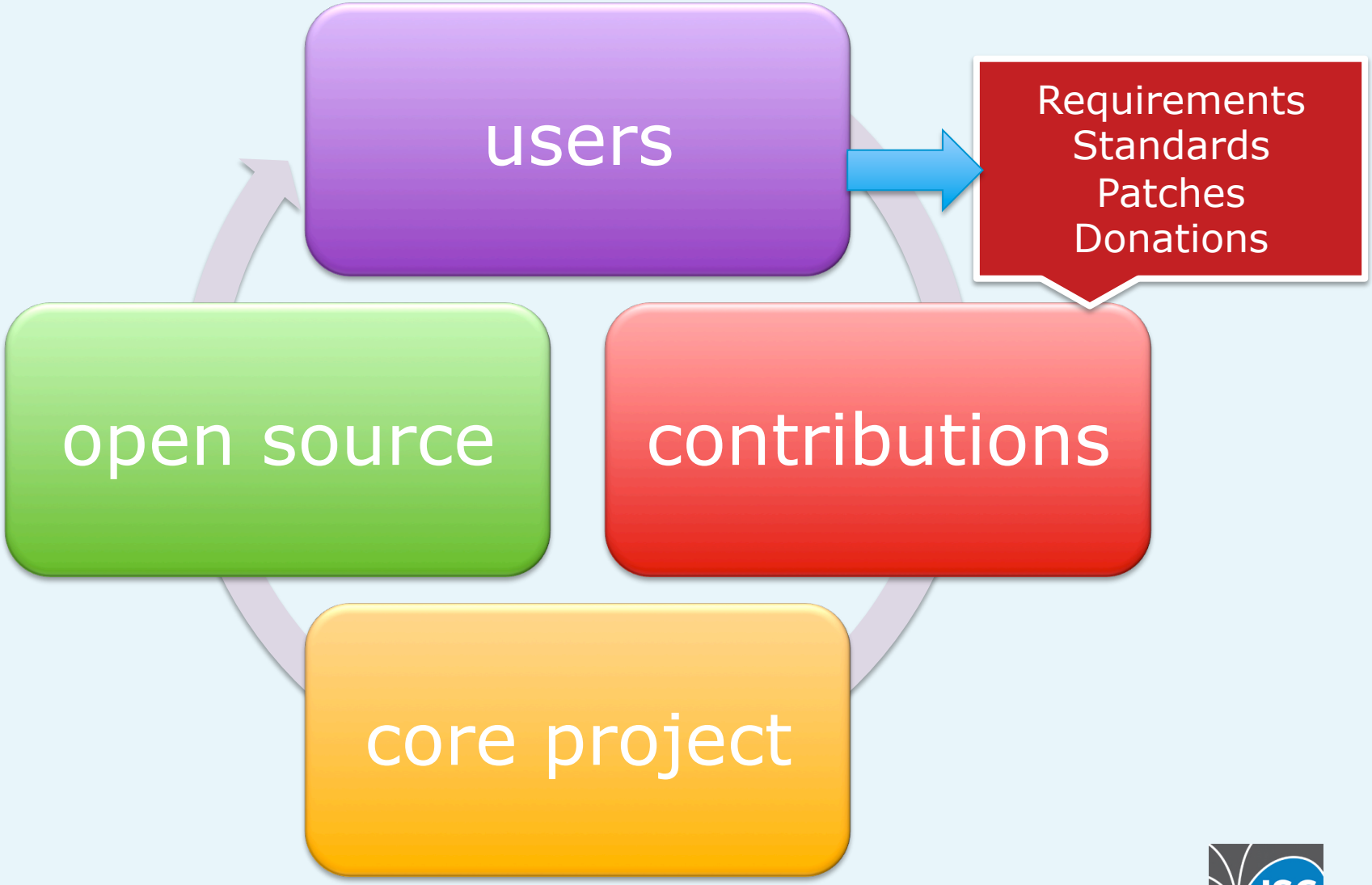
# HOW CAN WE BROADEN THE BASE?

Even slightly

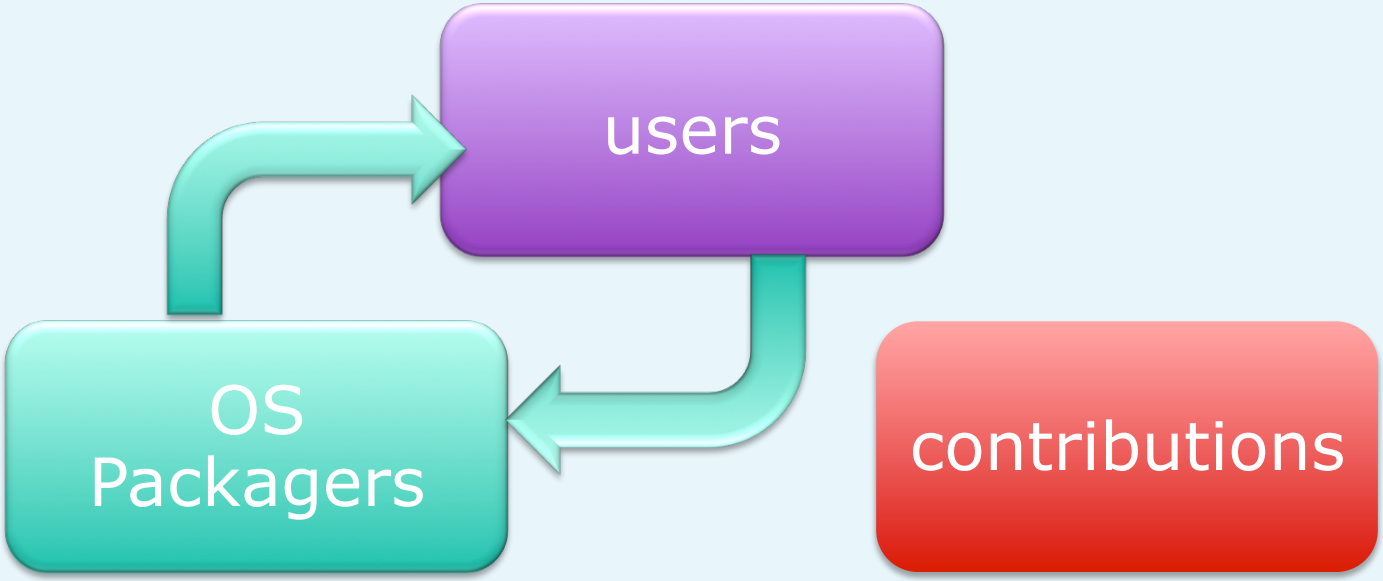
# Originally, a Virtuous Cycle



# Multiple ways to contribute



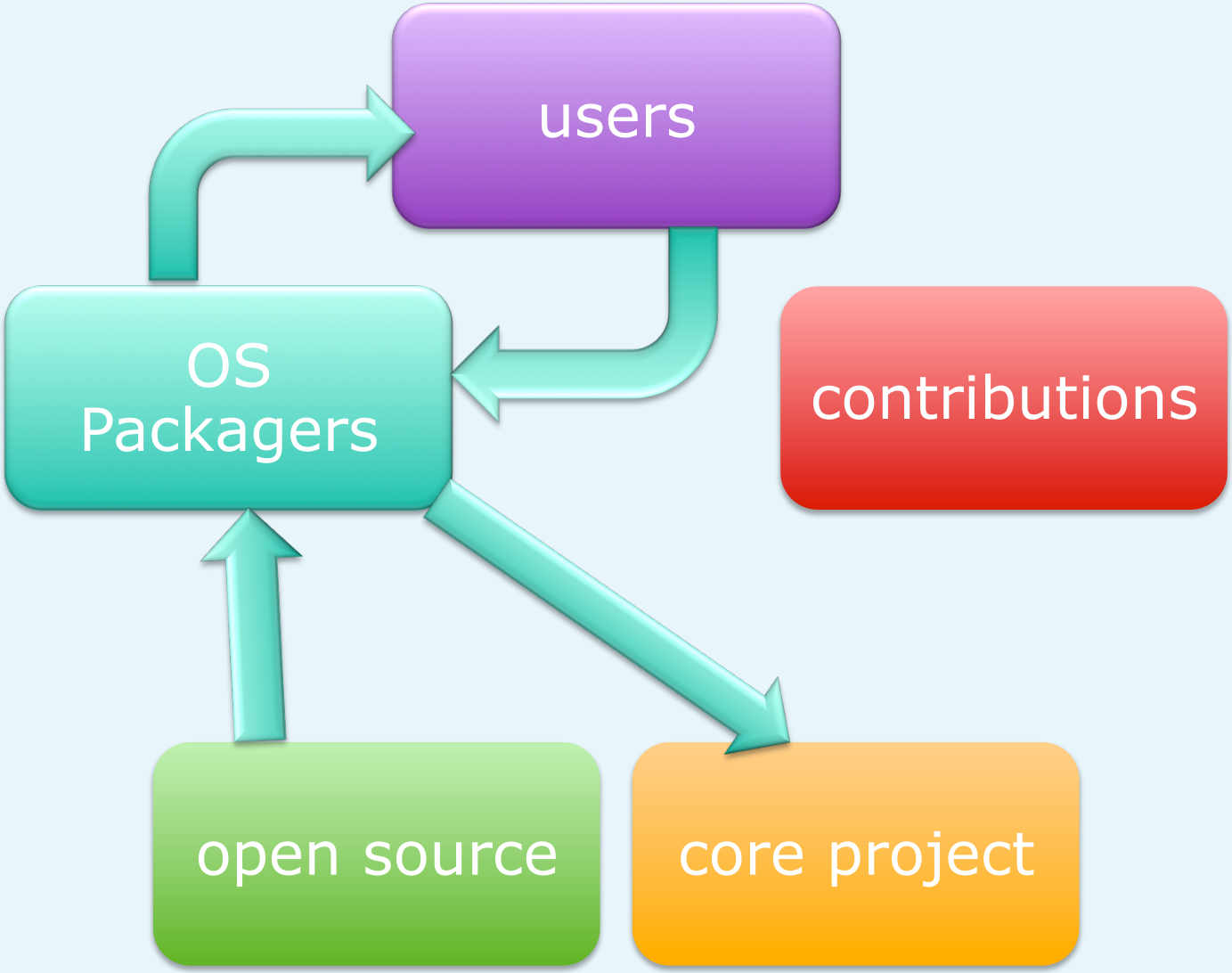
# Users began using packages



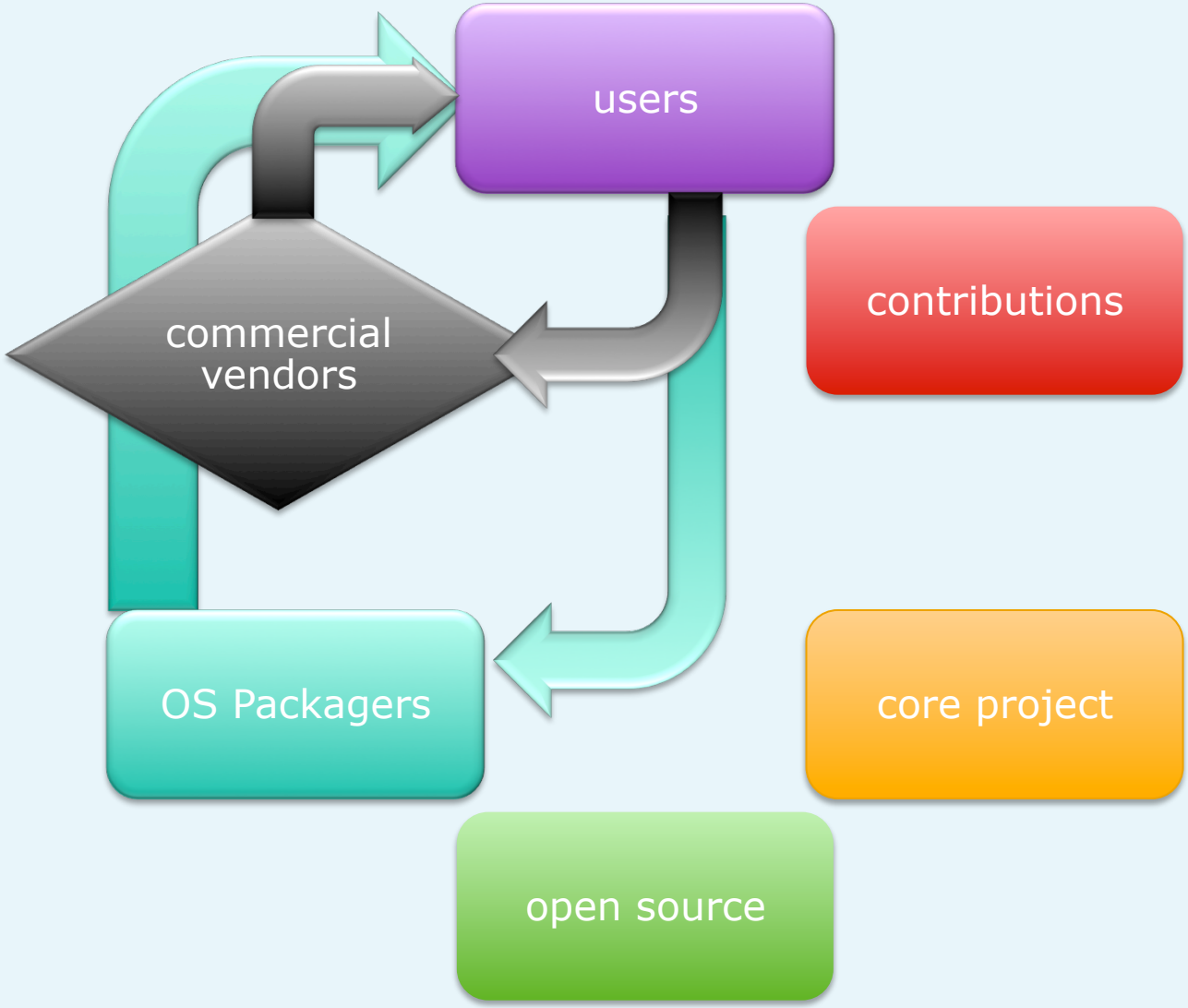

open source

core project

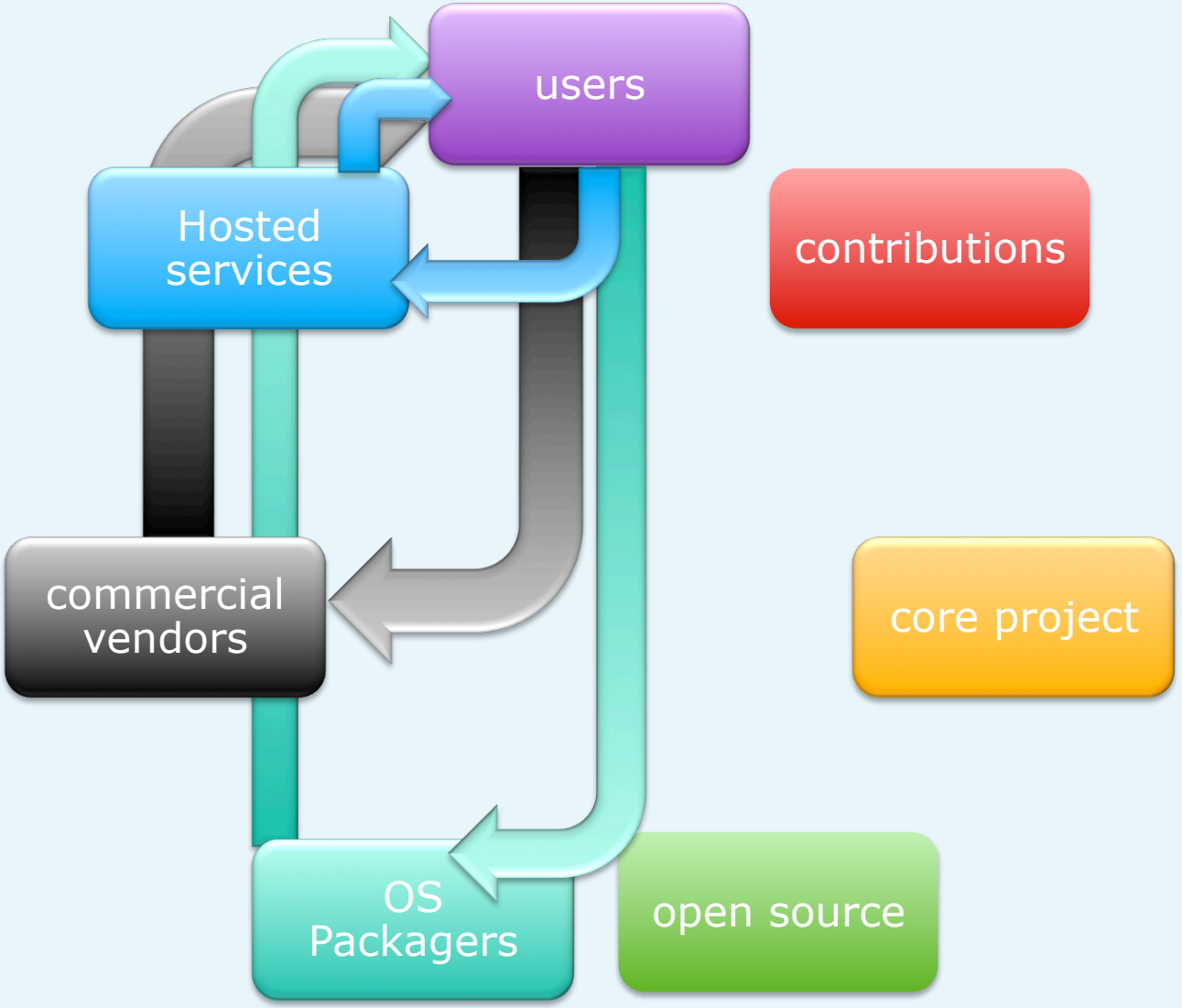
# Packagers interface with core



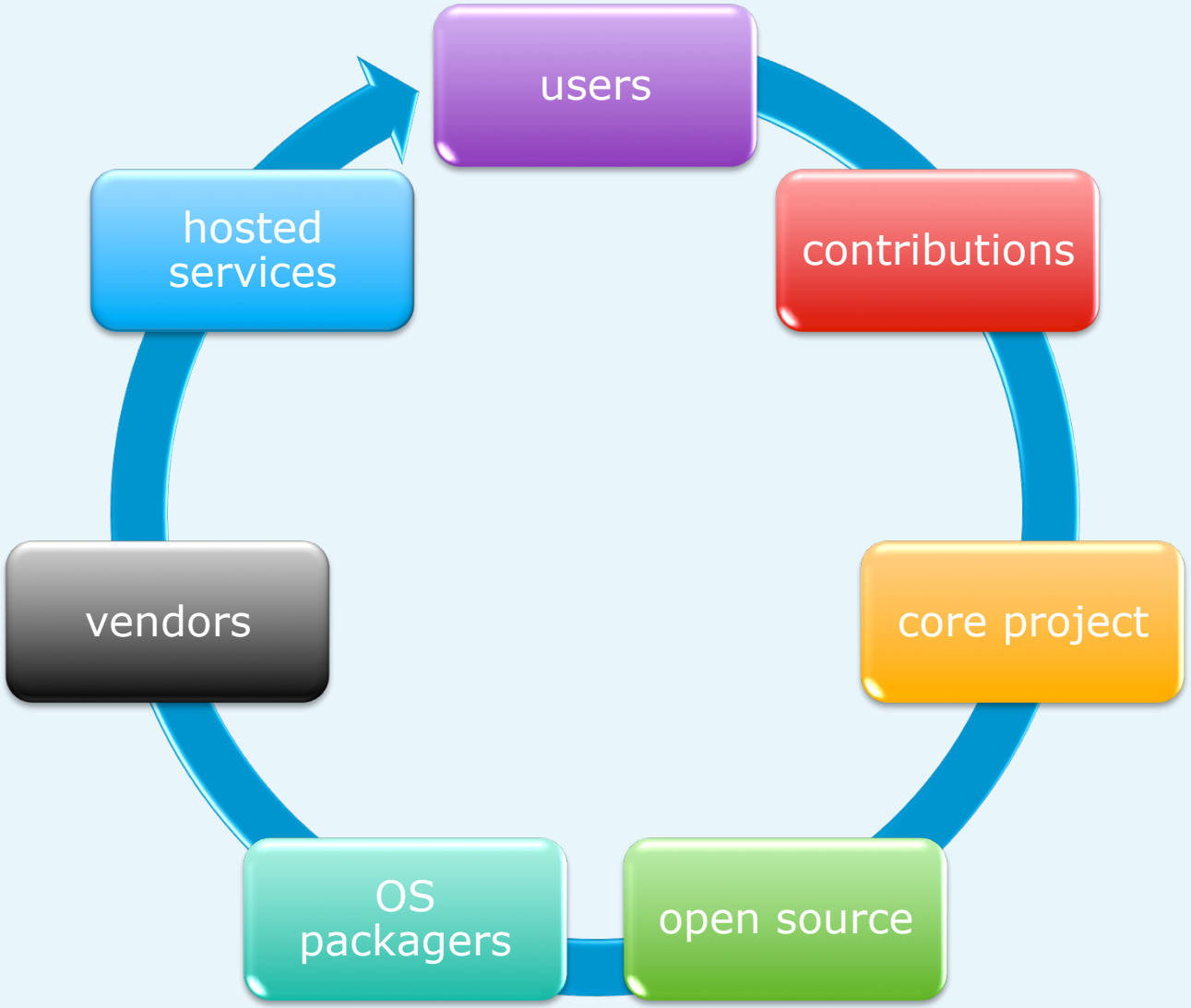
# Commercial apps captured users



# And now SaaS ...



# Can the circle be unbroken?



# POSSIBLE SOLUTION – MORE RESTRICTIVE OPEN SOURCE LICENSE

# Code for the common good

Preserve the original intent

- Encourage re-use and improvement
- Standards-based
- Transparent

Require commercial users to share their improvements or support the core team

# Summary

Considering a new open source license for BIND

- *Slightly more restrictive*. MPL2.0?

**We are looking for feedback**