The development of BIND, tracking the growth of the DNS and DNS standards over 30 years

Brian Reid

ISC

13 July 2018

Musty old software?

- BIND 9 first released October 2000
- Linux first released September 1991

Think about the evolution of the requirements for those systems.

And the concerns that implementing some new requirement might add complexity

BIND implemented DNS

- BIND implemented DNS. That was its identity. UC Berkeley
- BIND ended up in the custody of my laboratory at DEC (long boring story)
- P.Vixie insisted that it be the “reference implementation” (never mind that it was the only)
- He took BIND with him when he left.

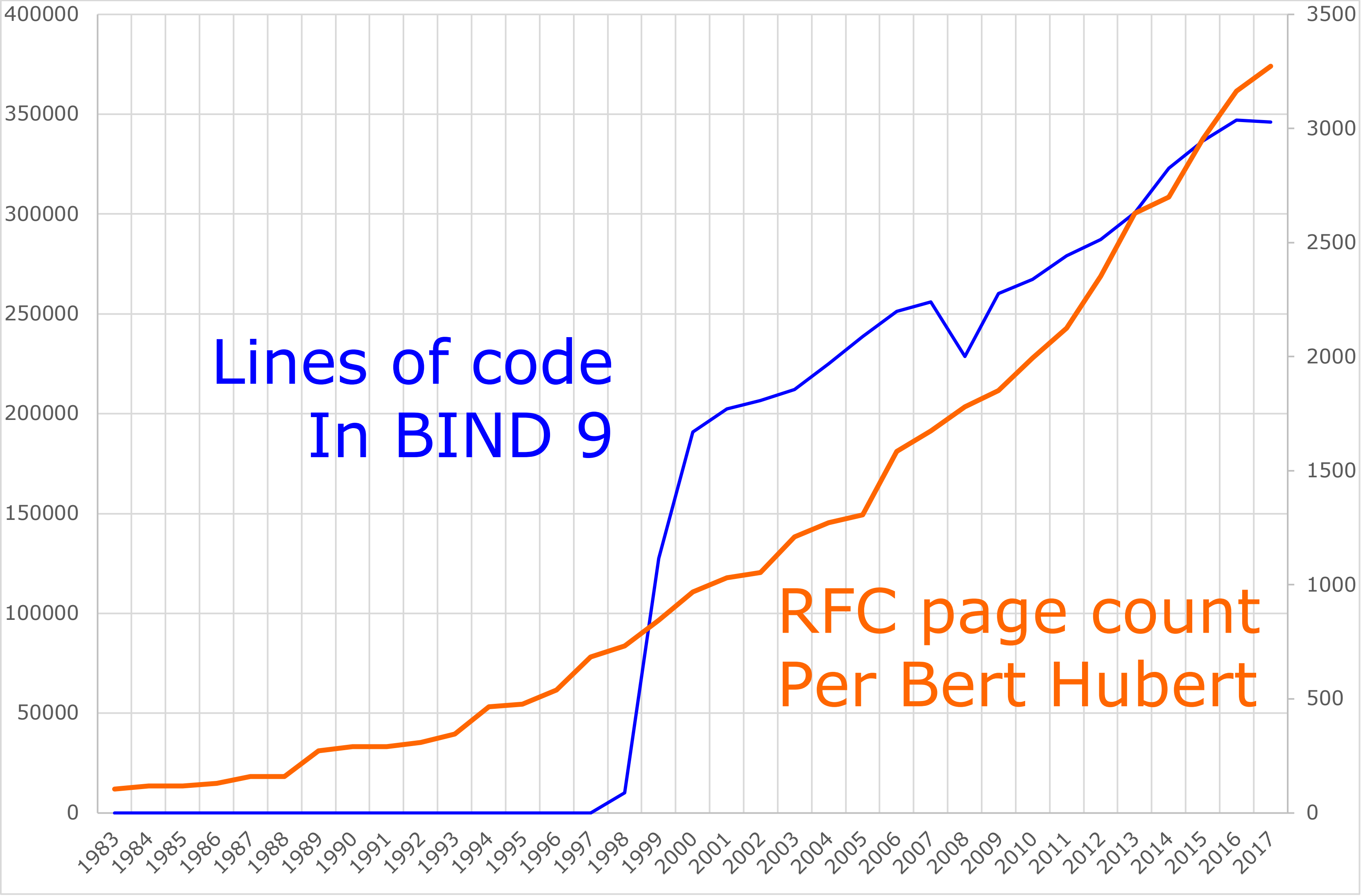
”Plan to throw one away, because you will anyhow”

- At Berkeley, BIND was BIND 4.
- Vixie realized need to start over.
- BIND 8 followed BIND 4 to mimic Sendmail, whose v8 followed v4
- BIND 8 accumulated many CVEs.
- Maybe Fred meant “Plan to throw two away…”

Requirement changes forced re-architecting BIND

- DNSSEC
- IPv6
- Multiprocessor support
- Remote management
- TSIG
- Dynamic update (DDNS)
- *etc*

RFC pages vs BIND code size



One feature over the line?

We asked people what they wished we had never put into BIND

- Views
- DNSSEC
- Universality (auth and recursive)
- DNAME
- Views
- NSEC3
- Did I say Views?

All power tools can kill

- Sometimes an RFC defines something that seems useful
- If standards-track, we put it into BIND
- Later, bad people discover how to abuse it or attack with it
- RFCs don’t advise how to control the monster, just how to hatch it
- My favorite example is IPv6

filter-aaaa-on-v4
filter-aaaa-on-v6

- BIND 9.12 option documentation:

  This option is intended to help the transition from IPv4 to IPv6 by not giving IPv6 addresses to DNS clients unless they have connections to the IPv6 Internet. This is not recommended unless absolutely necessary. The default is **no**. The **filter-aaaa-on-v4** option may also be specified in **view** statements to override the global **filter-aaaa-on-v4** option.

IPv6-related interactions

- CVE-2017-3135:

  Under some conditions when using both DNS64 and RPZ to rewrite query responses, query processing can resume in an inconsistent state leading to either an INSIST assertion failure or an attempt to read through a NULL pointer.

Accommodating operations

- From BIND 9.12 release notes:

  When acting as a recursive resolver, **named** can now continue returning answers whose TTLs have expired when the authoritative server is under attack…. This is controlled by the **stale-answer-enable**, **stale-answer-ttl** and **max-stale-ttl options**

- From BIND 9.11 release notes

  named will no longer start or accept reconfiguration if **managed-keys** or **dnssec-validation auto** are in use and the managed-keys directory … is not writable by the effective user ID

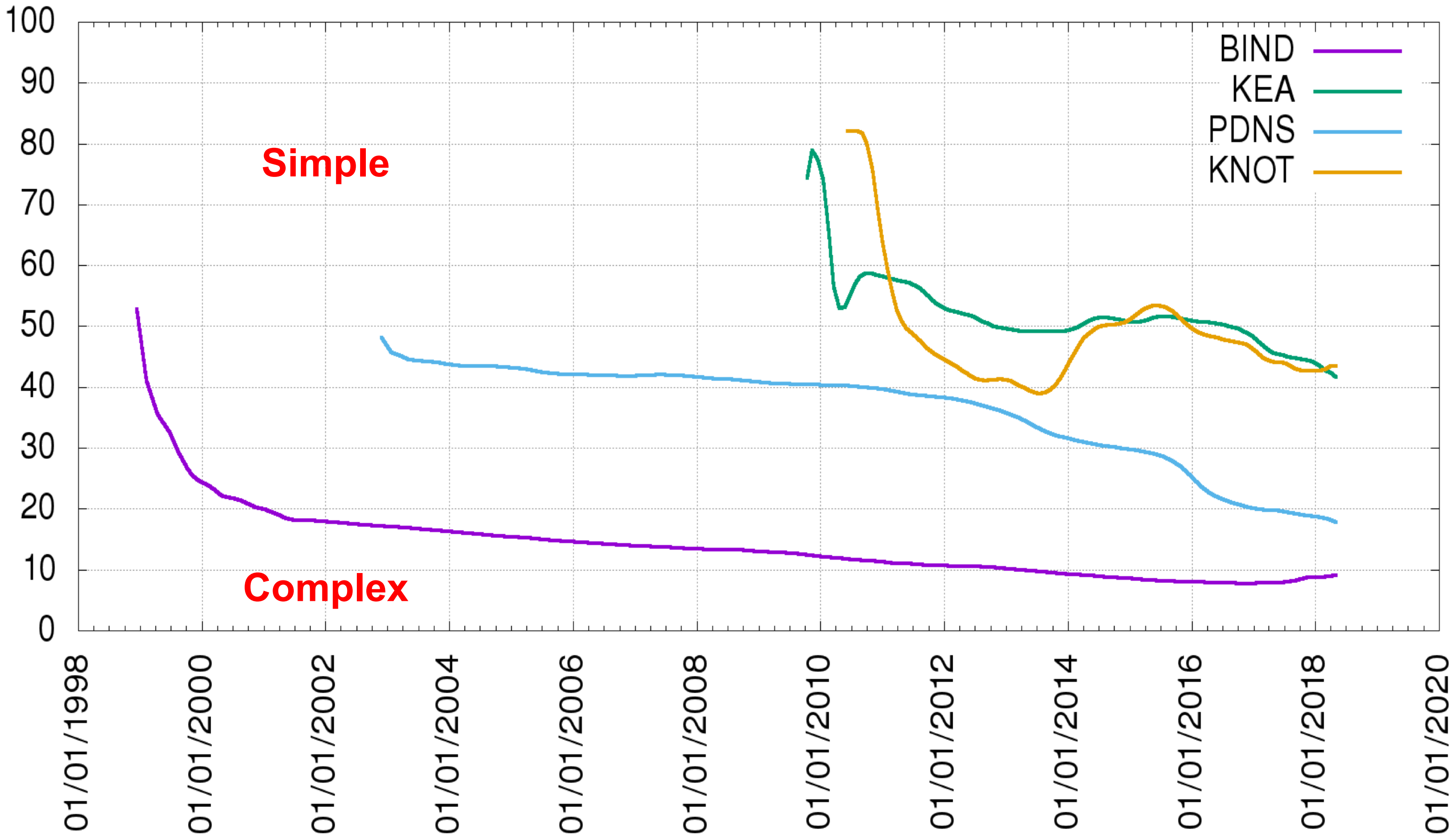
Between stasis and rewrite

- BIND has been rewritten twice
- 2500 pages of RFC since last rewrite (data from Bert Hubert’s Camel)
- Current practice is to refactor, not necessarily rewrite
- Refactoring is a partial rewriting of parts that need it most
- Remove complexity when you can
- Remove code when you can

Complexity and metrics

- For decades, “McCabe Cyclomatic Complexity Index” has been a metric
- Just as with College admission test scores: not the whole story
- ISC’s Witold Kręcicki devised a maintainability index by combining metrics (including that one)
- See https://www.isc.org/blogs/bind-9-refactoring/

Witold’s maintenance index

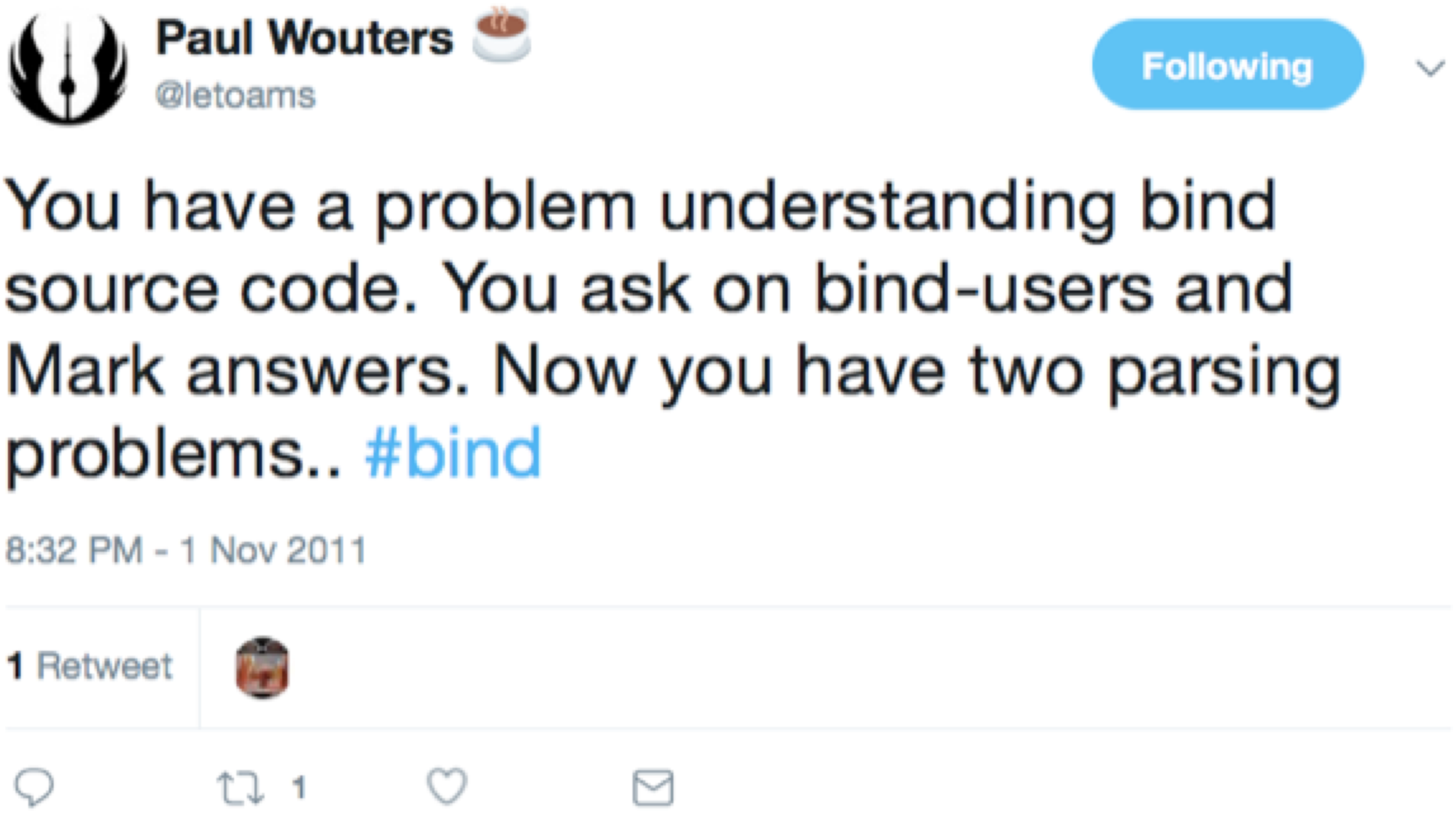


The challenge is simple
Accomplishing it is not

- Implement and test changes required by new RFCs
- Complexity will increase
- Can reset baseline with full re-implementation
- Or can identify problem spots and try to refactor

Growing complexity hurts

- More complex code is more likely to have problems
- Especially when changed
- Table of CVE *vs* code complexity is in Witold Kręcicki’s aforementioned blog
- Old quote from US TV advert: “You can pay me now, or you can pay me later”



Paul Wouters ☕
@letoams

You have a problem understanding bind source code. You ask on bind-users and Mark answers. Now you have two parsing problems.. #bind

8:32 PM - 1 Nov 2011

BIND 9.0

BIND 9.13