DISTEL
Domain Name **S**erver **Te**sting **L**ab

Daniel Karrenberg
with
Alexis Yushin, Ted Lindgreen
Olaf Kolkman

Presentation Outline

Why DISTEL ?

What is DISTEL ?

How DISTEL works !

Some DISTEL Results !...?..!..??.!

This is Work in Progress !

Why DISTEL ?

- Regression testing of nsd
- Performance evaluation of root server components
- Simulation and analysis of abnormal query loads

Maybe sooner or later:

- General functionality testing and performance evaluation

What is DISTEL ?

- Present a reproducible load to a server
  - Synthetic
  - Observed (tcpdump traces)
  - @ varying speeds (load)
- Record the answers (!)
- Extract information
  - Performance
  - Functionality testing (compare with expected responses)
  - Regression testing (compare responses of different runs)

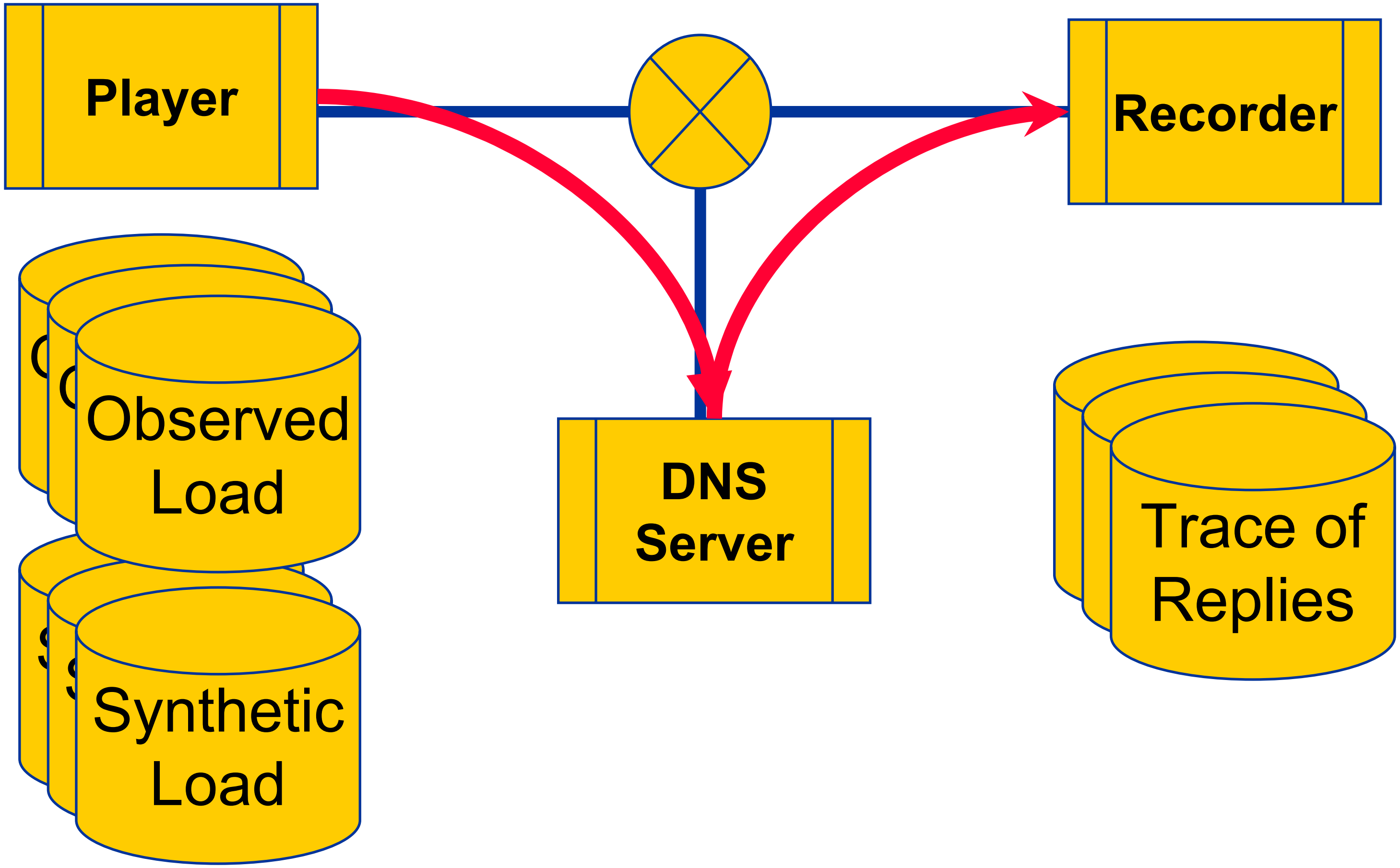
What is DISTEL ?

- 2 (3) machines running FreeBSD
  - Connected by two networks: test & control
- Some off-the-open-source-shelf tools:
  - tcpdump, perl, sed, diff, gnuplot, ethereal, sudo
- Some hacked tools:
  - tcpreplay
- Some special purpose software:
  - ~1500 lines of Perl, ~500 lines of C
  - Makefiles and other sundry scripts

What DISTEL is *not* !

- DISTEL is not Finished
- DISTEL is not a packaged set of software
  - Not finished
  - Set-up reasonably complex
  - Specialist knowledge required to operate
  - Packaging and documenting is a lot of work

How DISTEL Works !



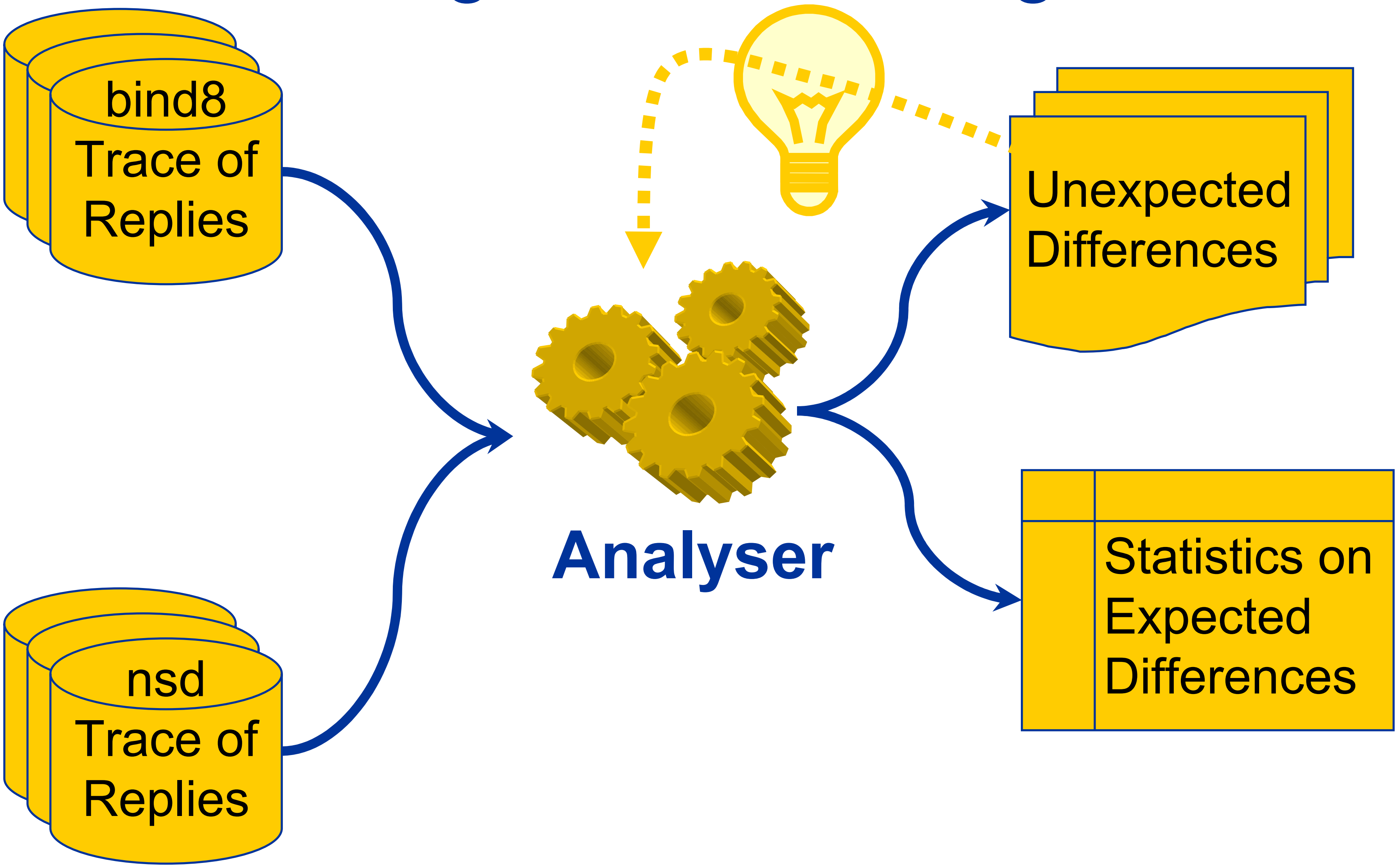
Some Details

- Player controls test-runs
  - Starts recorder & collects recorded answers
- Adapt destination addresses of load
  - MAC & IP / Checksums
- Log Experimental conditions
  - OS parameters / Software versions / Arguments of Player, Recorder and cooperating Target Server
- Replay Load & Record
  - Use “tcpreplay” & tcpdump
  - Timing!

Regression Testing

- Compare Responses of Different Runs
  - After modifications to software
  - Different Implementations
  - …
- High Volume
  - Typically O(900k) responses per run
  - Cannot compare manually
  - Need to categorise differences
  - Note unforeseen differences

Regression Testing



bind8 Trace of Replies

nsd Trace of Replies

Analyser

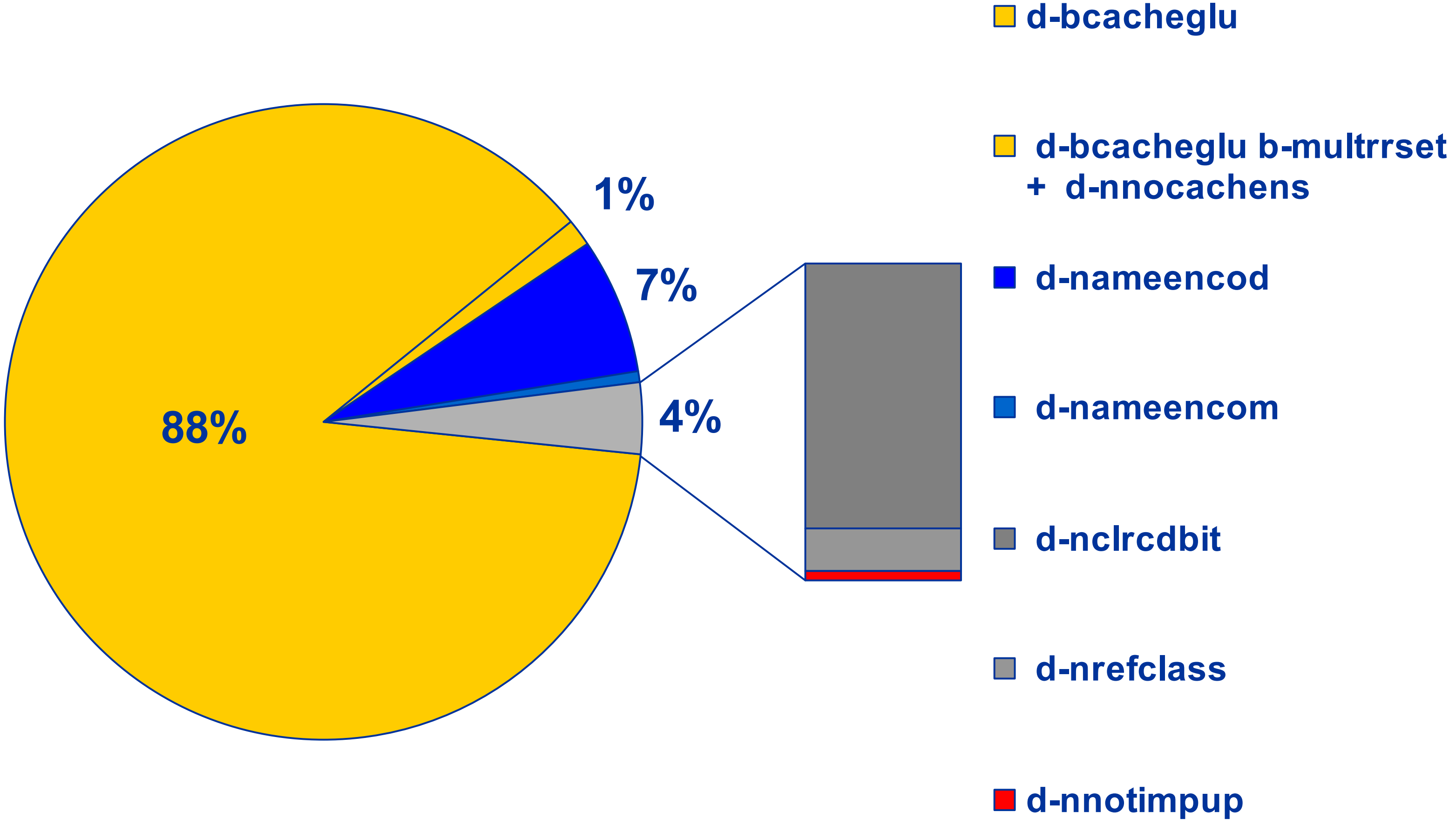
Unexpected Differences

Statistics on Expected Differences

```
                    --------------------------------------------------------------------------
                    k-root bind-8.3.3 / nsd-1.0.1      Total Answers: 899990
                    --------------------------------------------------------------------------
                                              d-bcacheglu  47182 /  5.24%
                                             d-nameencod   3779 /  0.42%
                                               d-nclrcdbit   1619 /  0.18%
                                 d-bcacheglu b-multrrset    628 /  0.07%
                                             d-nameencom    340 /  0.04%
                                               d-nrefclass    254 /  0.03%
                                              d-nnotimpup      55 /  0.01%
                                              d-nnocachns      17 /  0.00%
                                               d-nnotimpny       4 /  0.00%
                                     b-rootdot b-nonxdom       3 /  0.00%
                                               d-bindchaos       2 /  0.00%
                    --------------------------------------------------------------------------
                                        Total Different Responses  53883 /  5.99%
==================================================================================================

       b-multrrset -   bind puts same RRSet in multiple sections:    628 /  1.15%
      b-nonxdom   -      bind misses NXDomain when no zone cut:     3 /  0.01%
           b-rootdot   -            bind answers queries for ROOT.:     3 /  0.01%
         d-bcacheglu -            bind answers with chached glue:  47810 / 87.70%
           d-bindchaos -              bind answers to CHAOS *.bind.:     2 /  0.00%
           d-nameencod -                    different name encoding:   3779 /  6.93%
                                                 Additional bytes:  13711
     d-nameencom -  name enc. causes omission of additional RR:    340 /  0.62%
                                                      Omitted RRs:    340
           d-nclrcdbit -              nsd clears CD bit in response:   1619 /  2.97%
        d-nnocachns -     ns returns no non-authoritative answers:     17 /  0.03%
         d-nnotimpny -       nsd returns NotImp on notify requests:     4 /  0.01%
        d-nnotimpup -      nsd returns NotImp on update requests:     55 /  0.10%
      d-nrefclass -   nsd returns Refused on unknown class/type:    254 /  0.47%
                    --------------------------------------------------------------------------
                                             Total Differences  54514 /100.00%
==================================================================================================
```

bind-8.3.3 / nsd-1.0.1 root

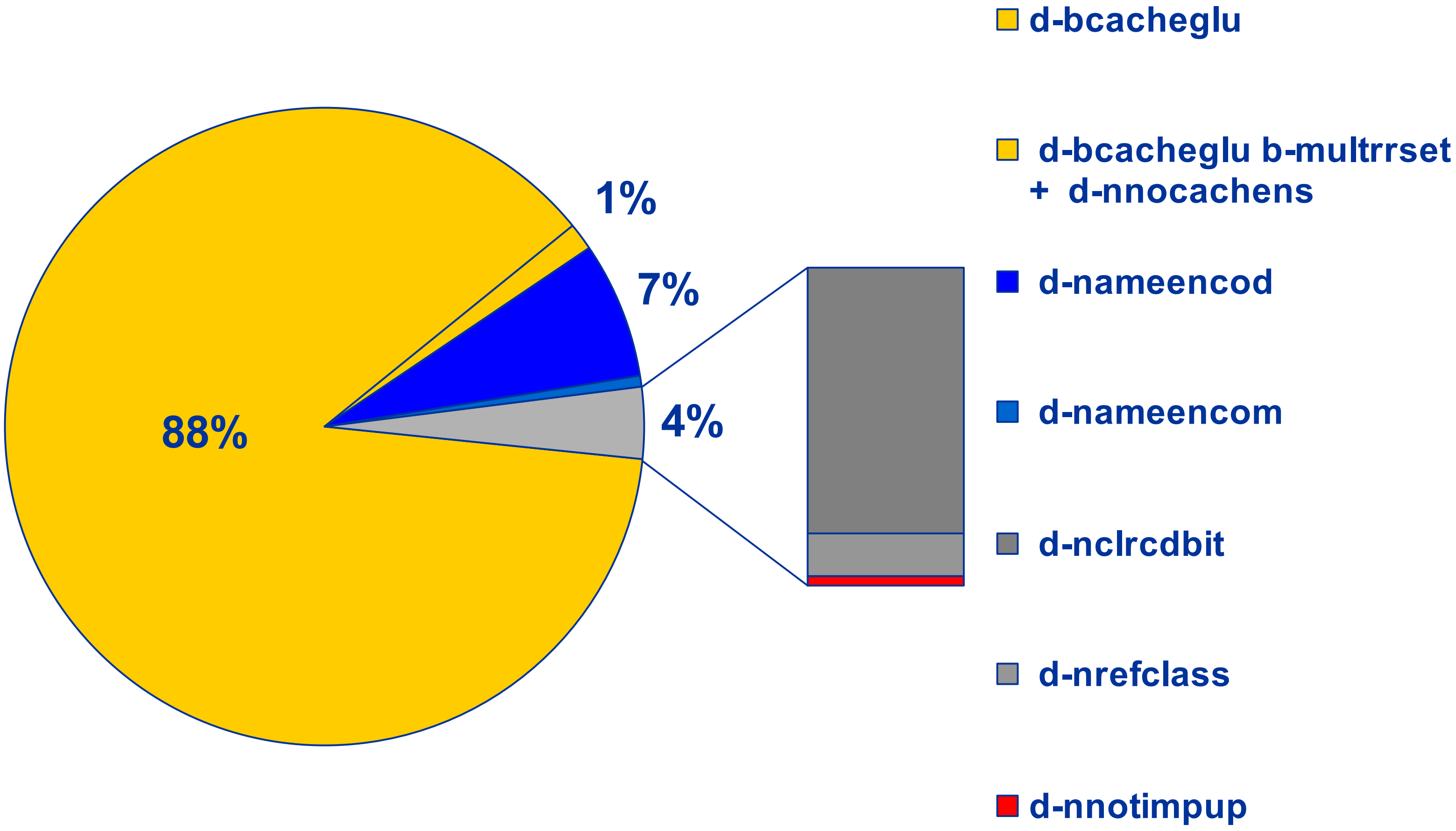


Total Differences 5.99%

nsd Name Encoding

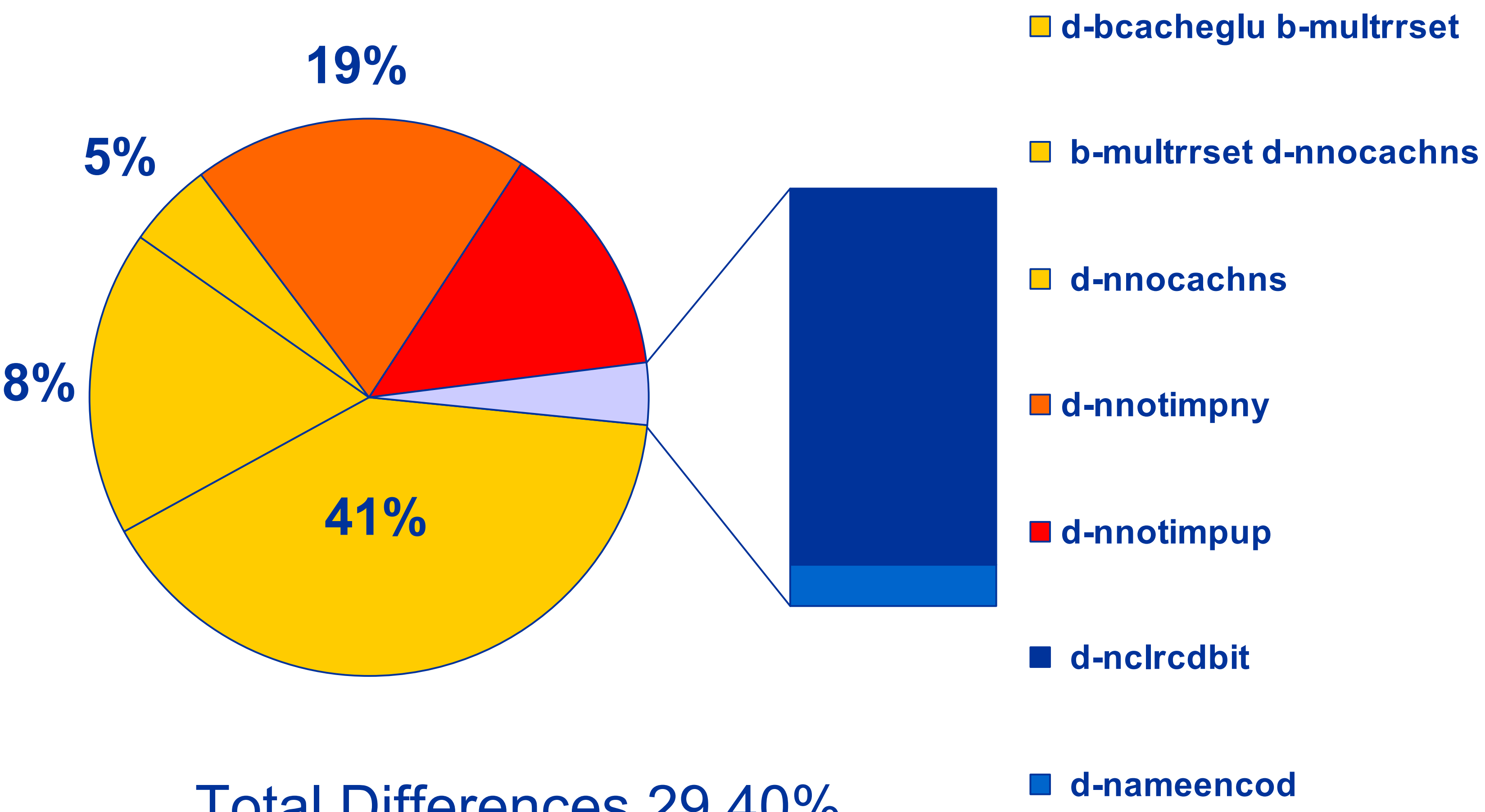
- Same **Response** / Different Encoding
  - 0.42%
  - Output bandwidth at IP level +0.04%
- Same **Answer** / Different Additional Info
  - 0.04%
  - **1** RR omitted from **additional** section
  - All of these queries for very long names
  - Almost all query names contain “**`._msdcs.`**” most contain “**`.Default-First-Site-Name.`**”
- No Answer Truncations (in any Test Run)

bind-8.3.3 / nsd-1.0.1 root



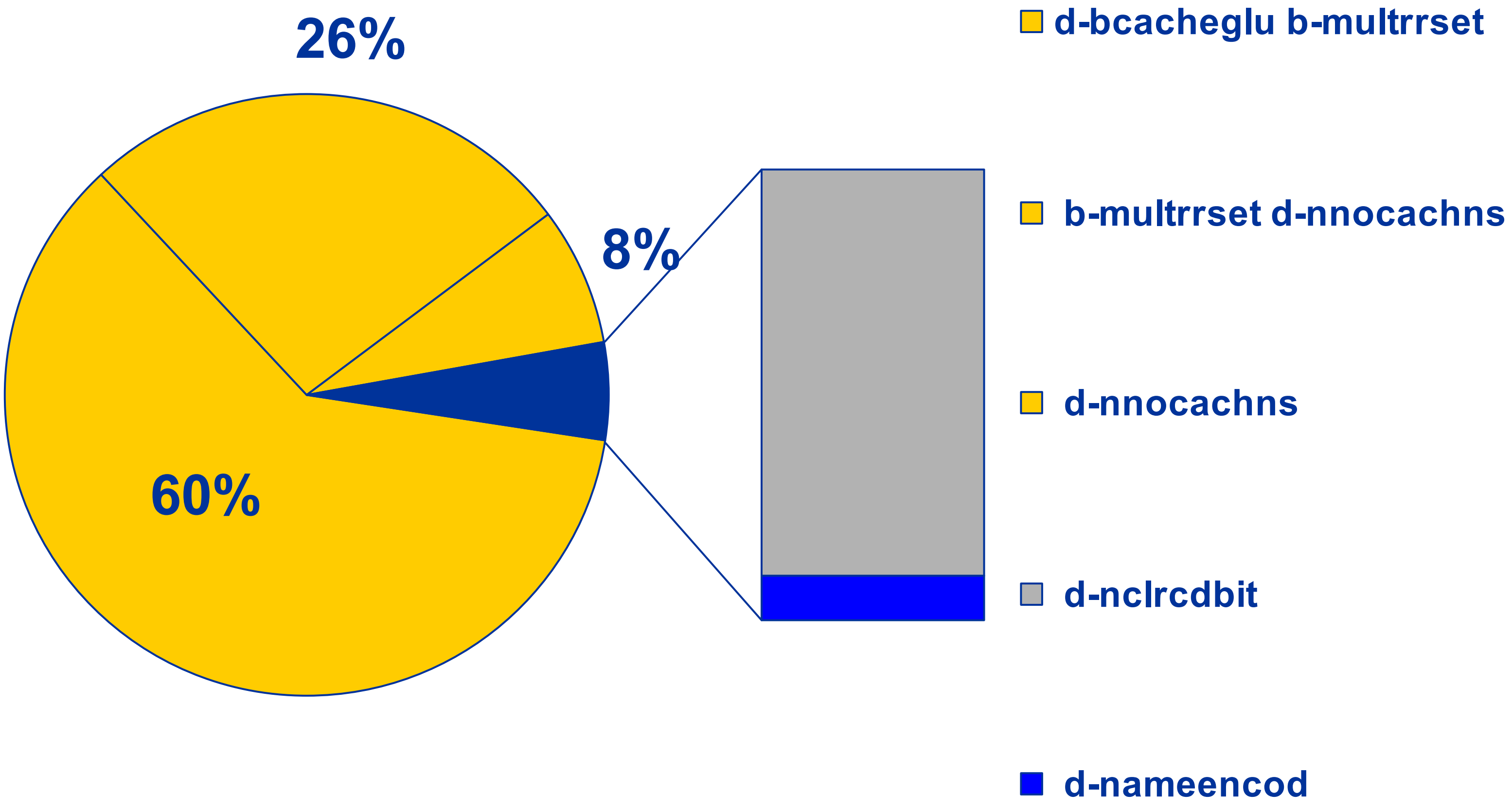
Total Differences 5.99%

bind-8.3.3 / nsd-1.0.1 .NL



Total Differences 29.40%

Normalised .NL



Total Differences 19.6%

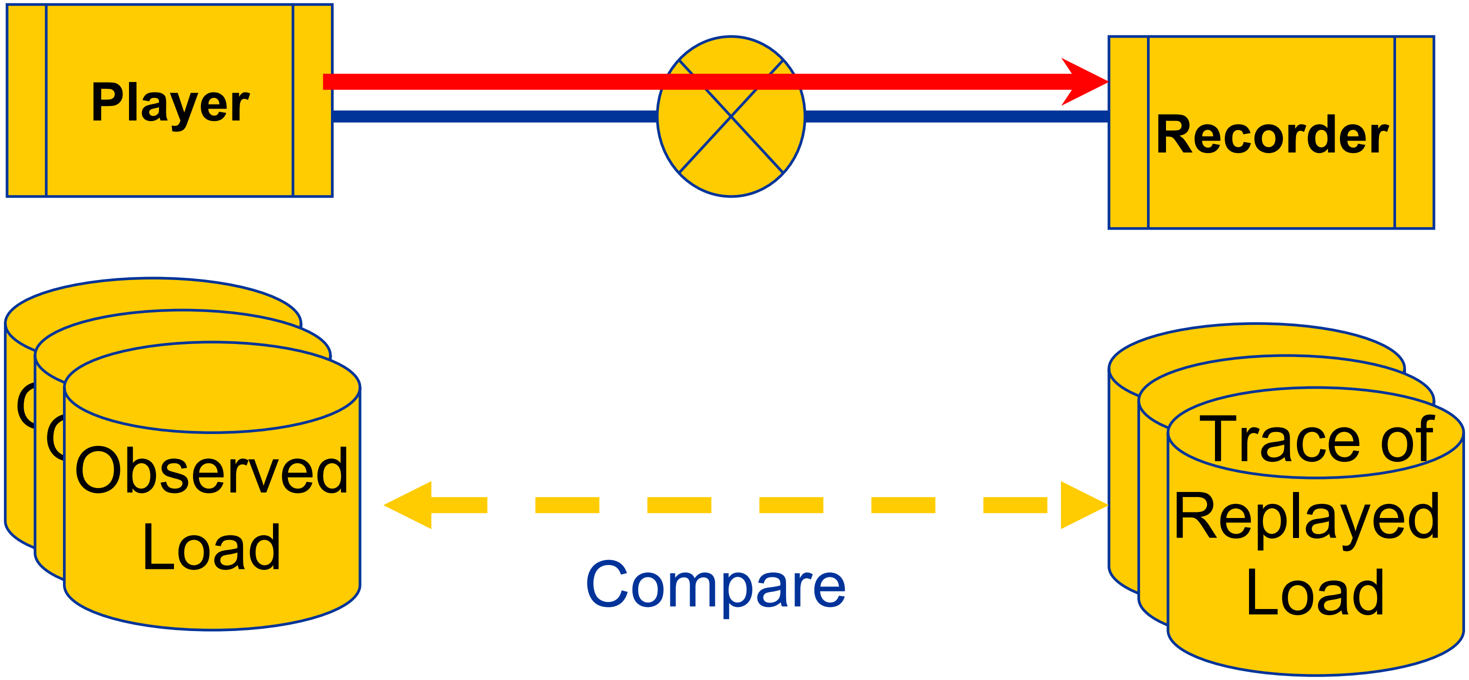
Differences Evaluated

- None of these differences will be noticed by resolvers conforming to the Internet standards.
- Extensive testing and documenting the differences is part of our very conservative and very extensive testing effort.
- We know of no other published testing going to this level of detail and openness.
- None of these differences will be noticed by resolvers conforming to the Internet standards.

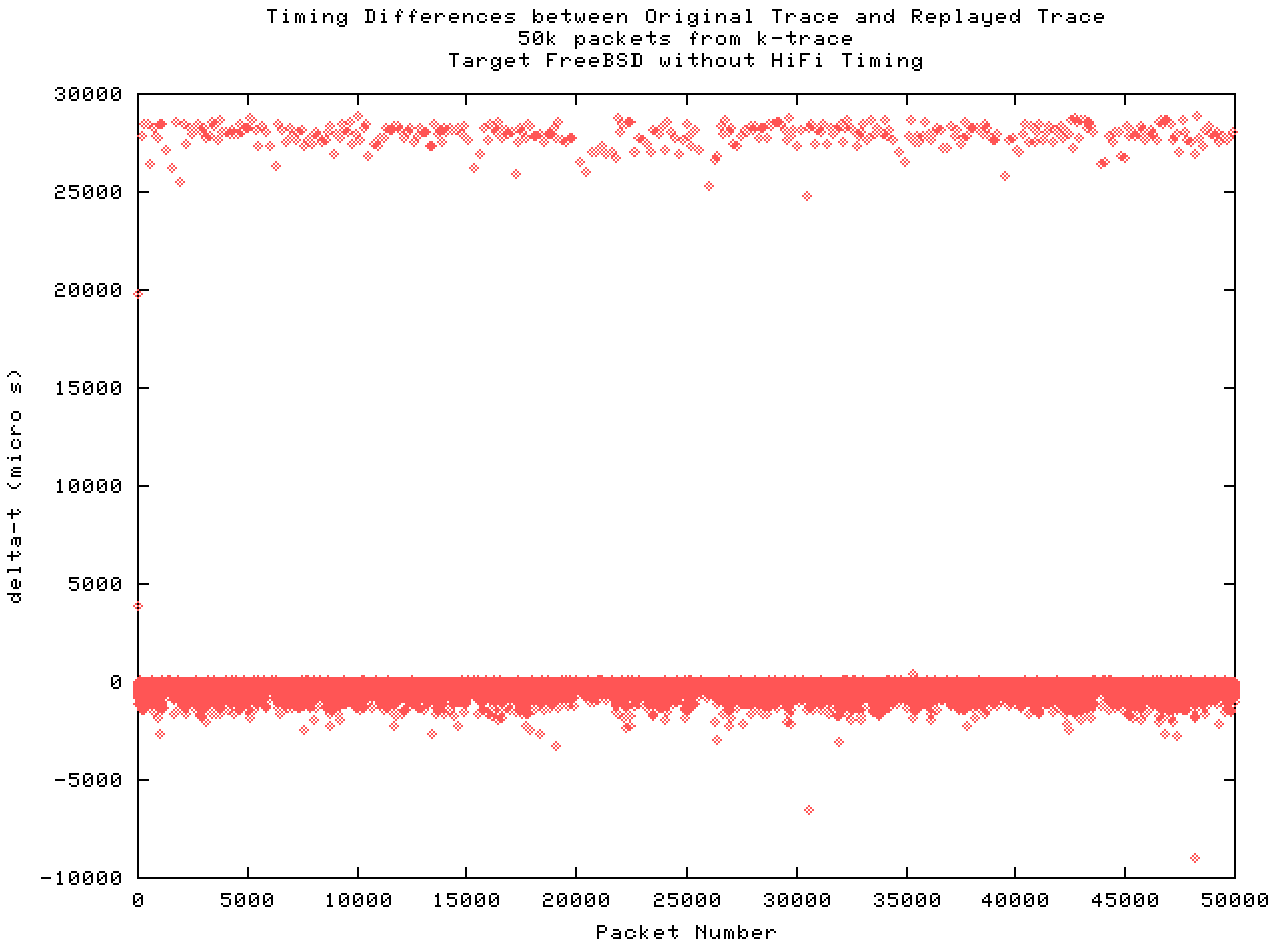
Performance Testing

- Play a test load at various speeds
  - Check if responses are correct
  - Count how many responses are received
- Future
  - Measure reponse timing ?
  - More variations in load characteristics
    - Burstiness
    - Anomalies (DDoS)
- Examples are older runs

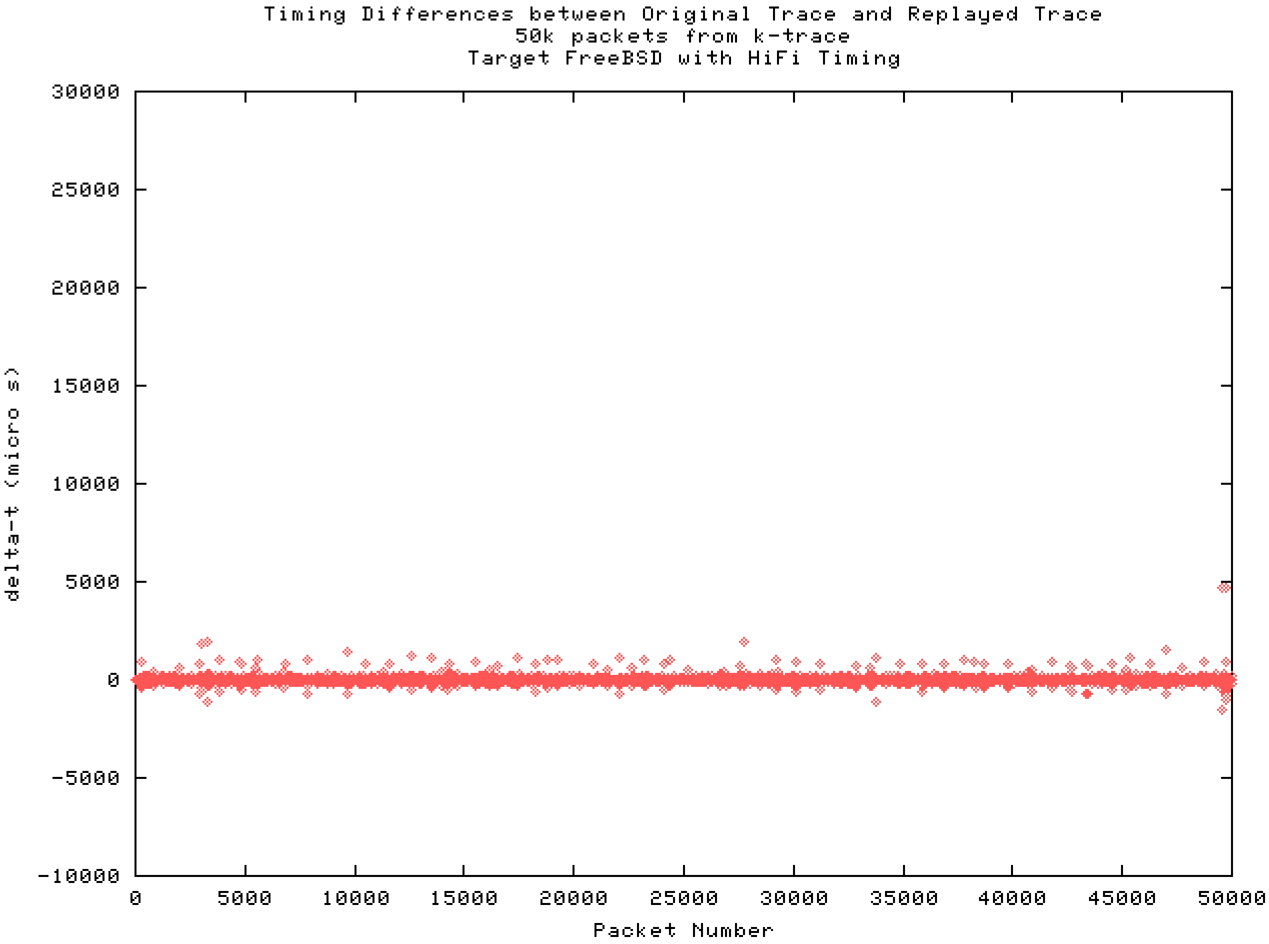
Verifying Player Timing



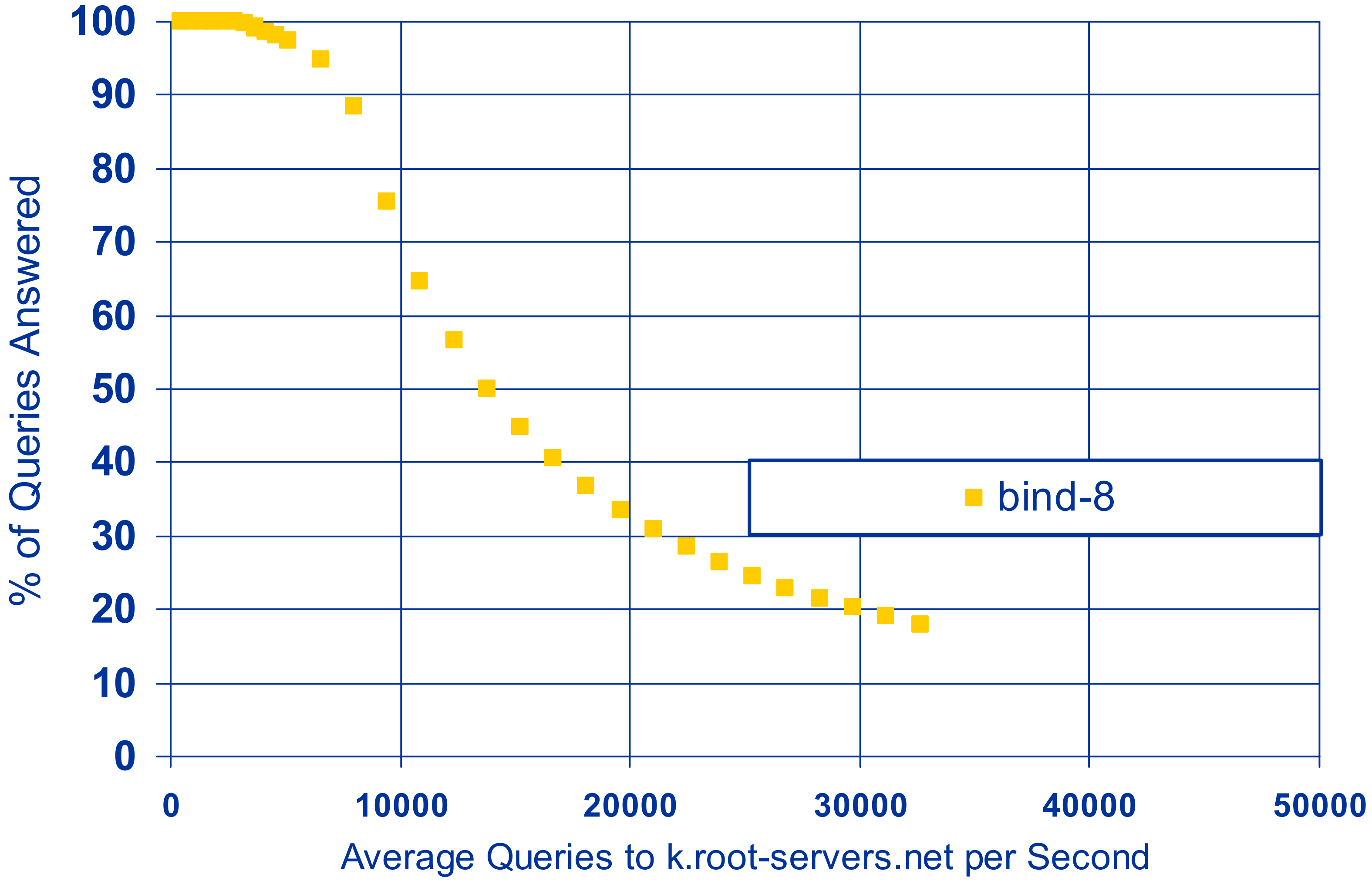
Original Player Timing



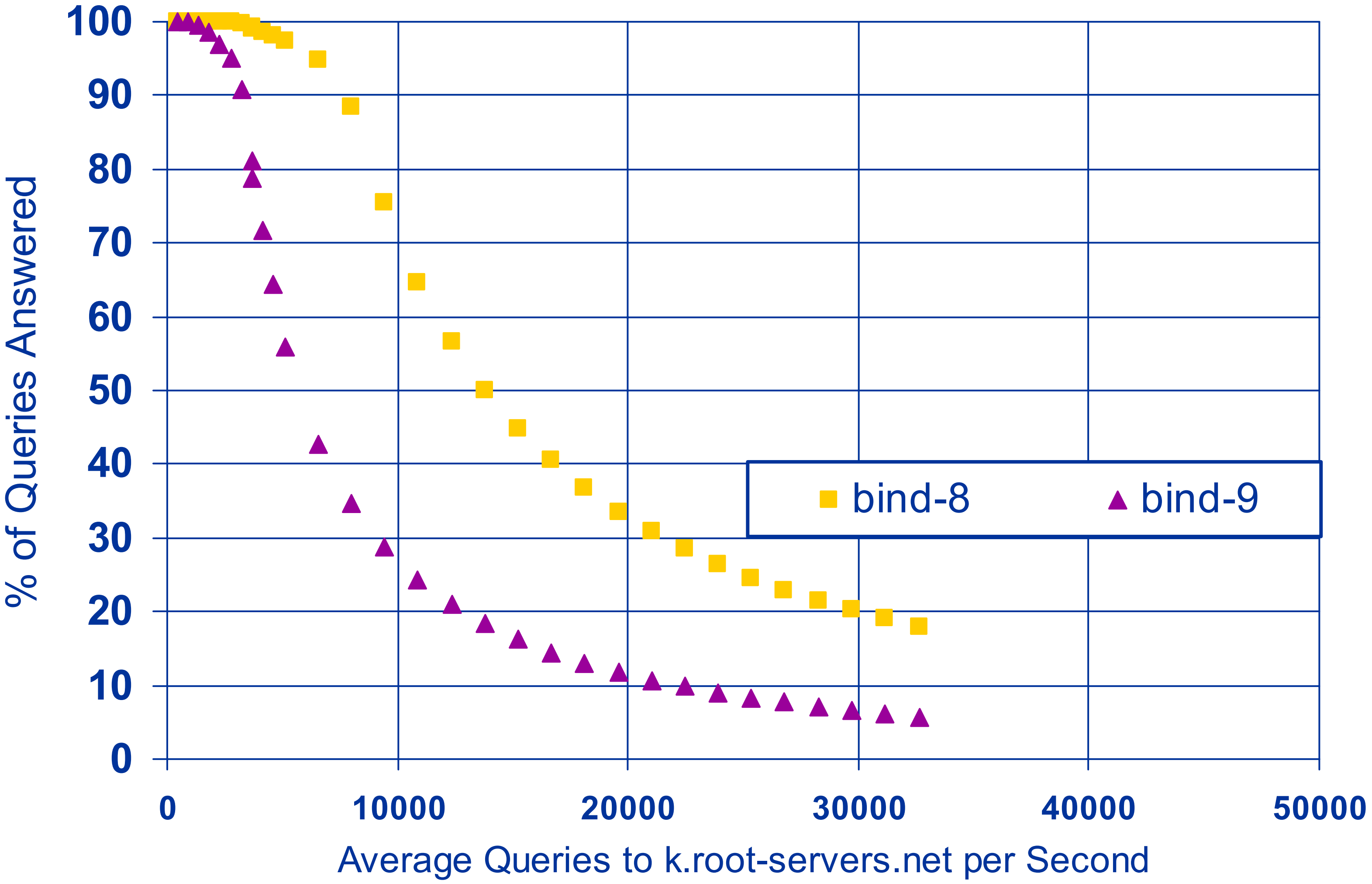
Better Player Timing



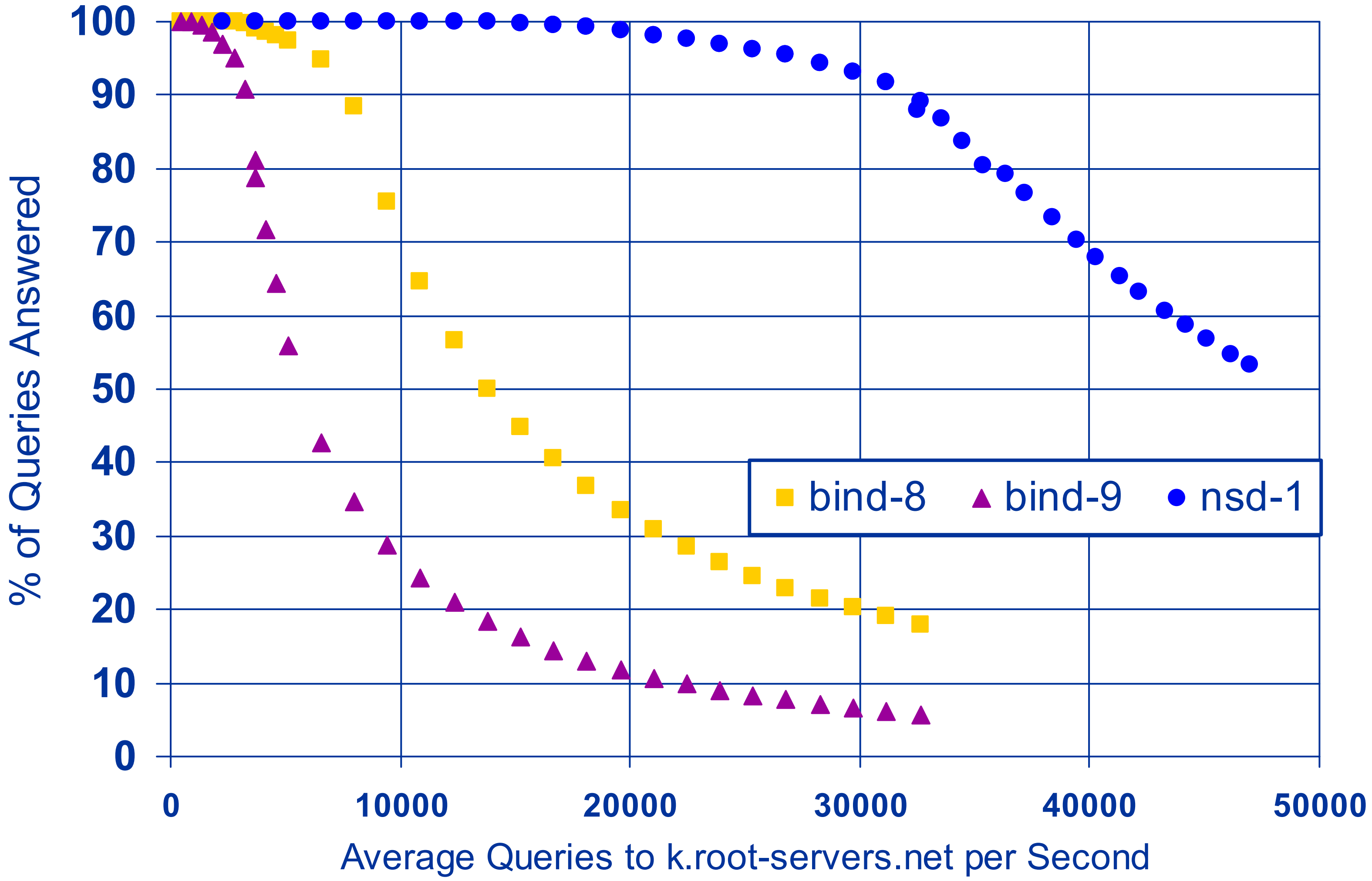
Performance Results



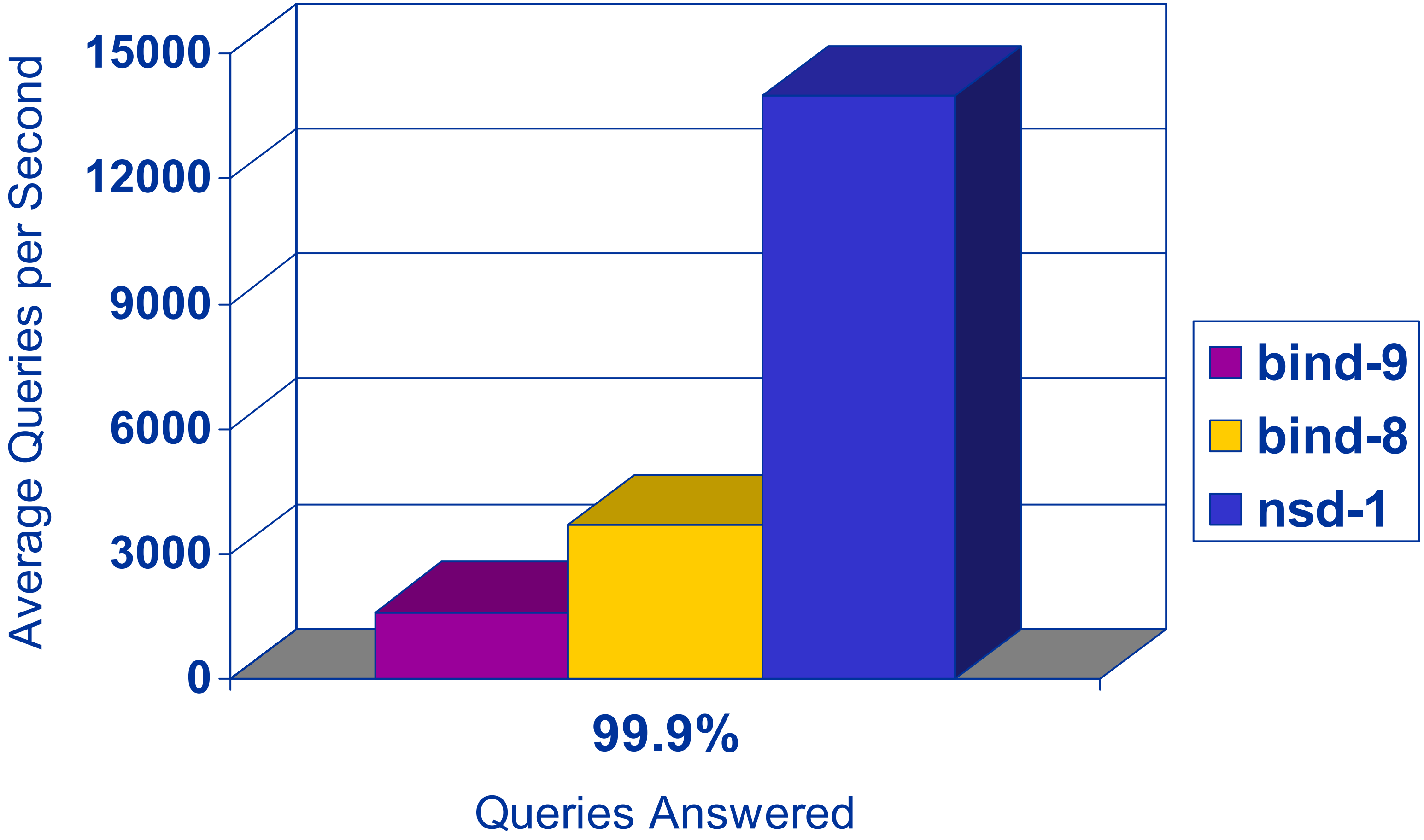
Performance Results



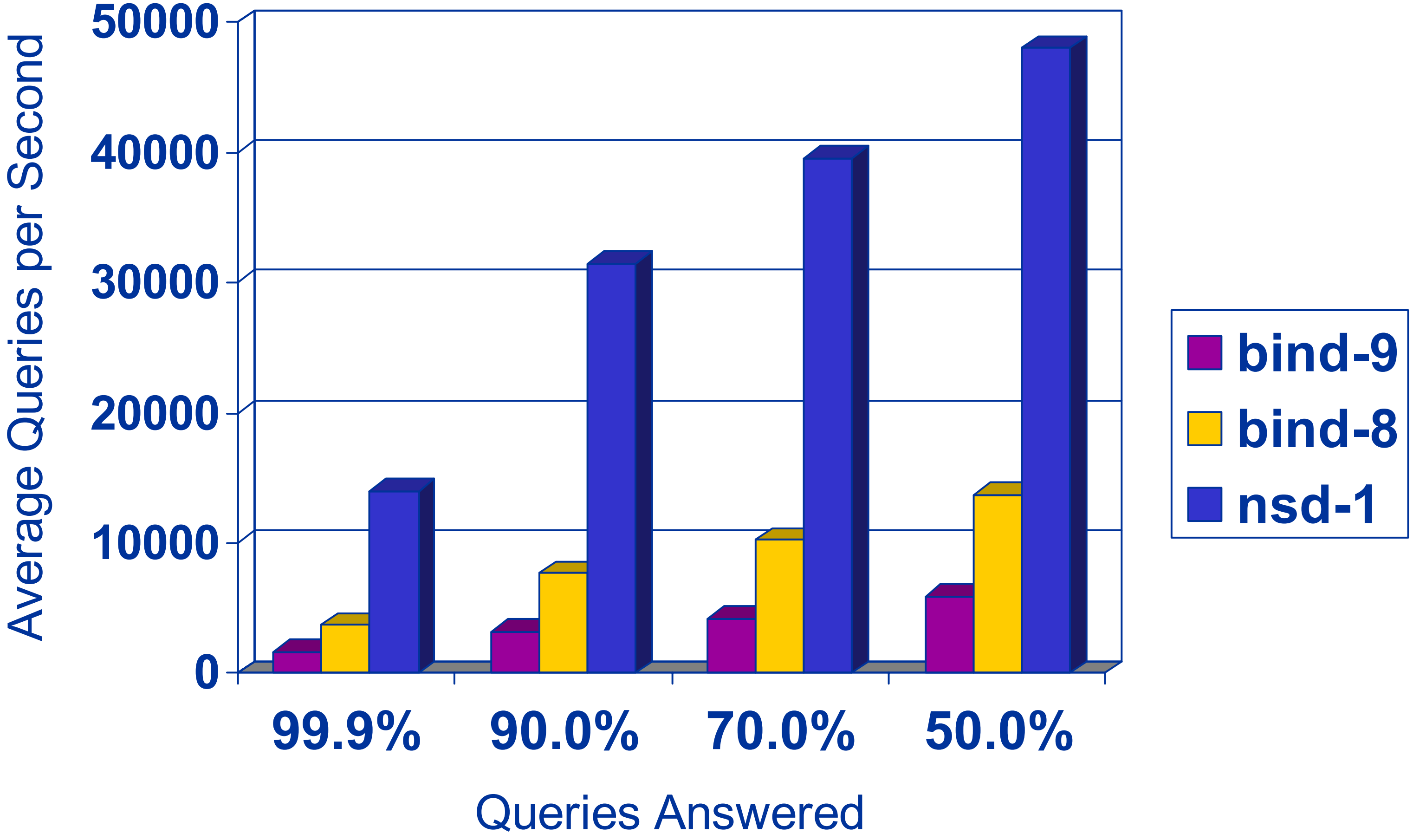
Performance Results



Performance Results “Marketing Version”



Performance Results
“Load Sharing”



Questions???

- Slides and other information will be available from http://www.ripe.net/

Documentation of observed differences and performance is in the nsd-1.0.1 distribution.

Interest in DISTEL as-is? Talk to me.