PROVISIONING PERFORMANCE OF NAME SERVER SOFTWARE

cira

BUILDING A BETTER ONLINE CANADA

Presented by: Evan Thompson

evan.thompson@cira.ca

INTRODUCTION

- D-Zone Anycast DNS
  - Anycast cloud with 18 points of presence
  - BIND running in each location
    - Diversifying software this year

MOTIVATION

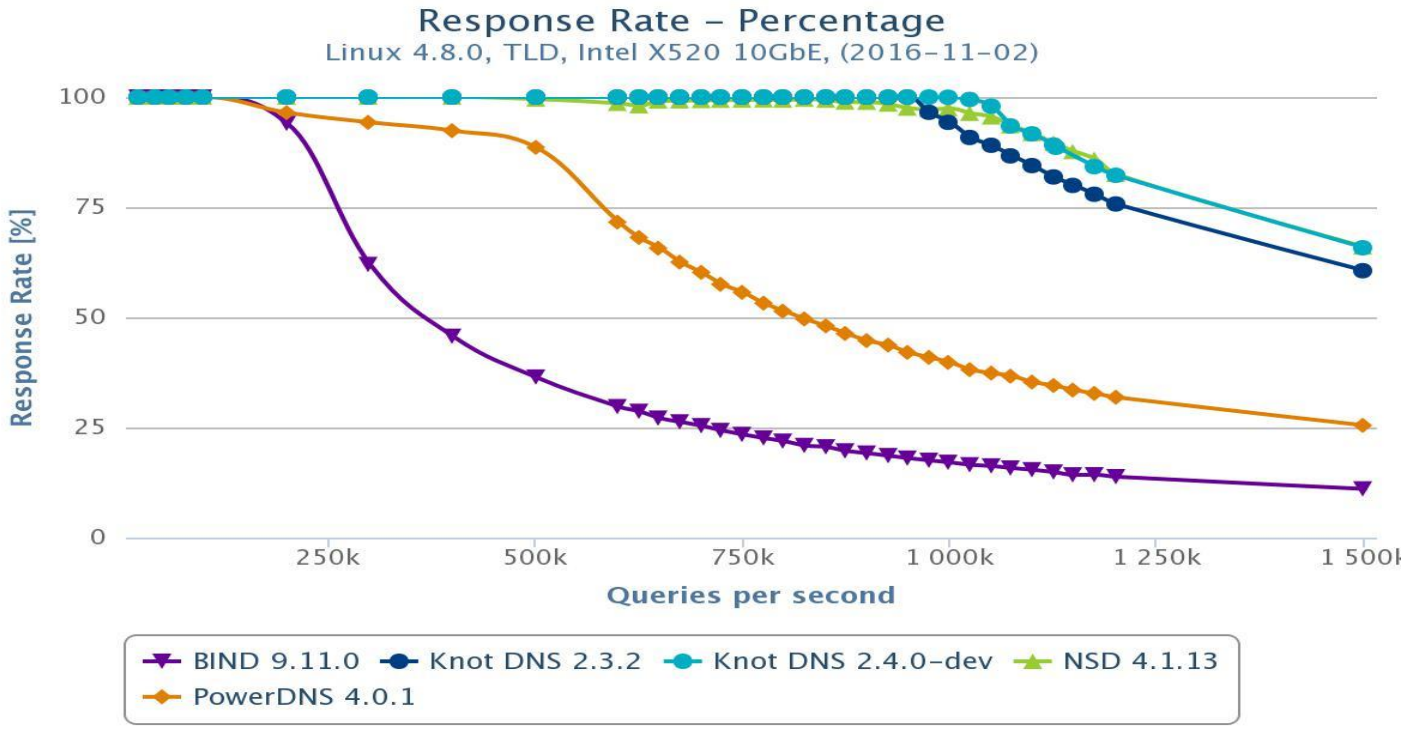
- D-Zone launched December 2013
  - Past year has seen substantial growth
  - 200+ customers
  - Requirement to scale to 1 million zones in the next year

DOES IT SCALE?

- Considerable effort underway ensuring performance of system at scale
  - Portal/API
  - Stats collection
  - Provisioning Engine
- BIND at scale?
  - Query performance
  - Provisioning performance

QUERY PERFORMANCE

- Plenty already out there



WHAT I WANTED TO KNOW

- Given a name server loaded with X zones:
  - How long does it take to add new zones?
  - Affect on query performance?
- Tested with:
  - BIND 9.11.0-P5
  - Knot 2.4.1-1
  - NSD 4.1.14

TEST SET UP

TESTING PHILOSOPHY

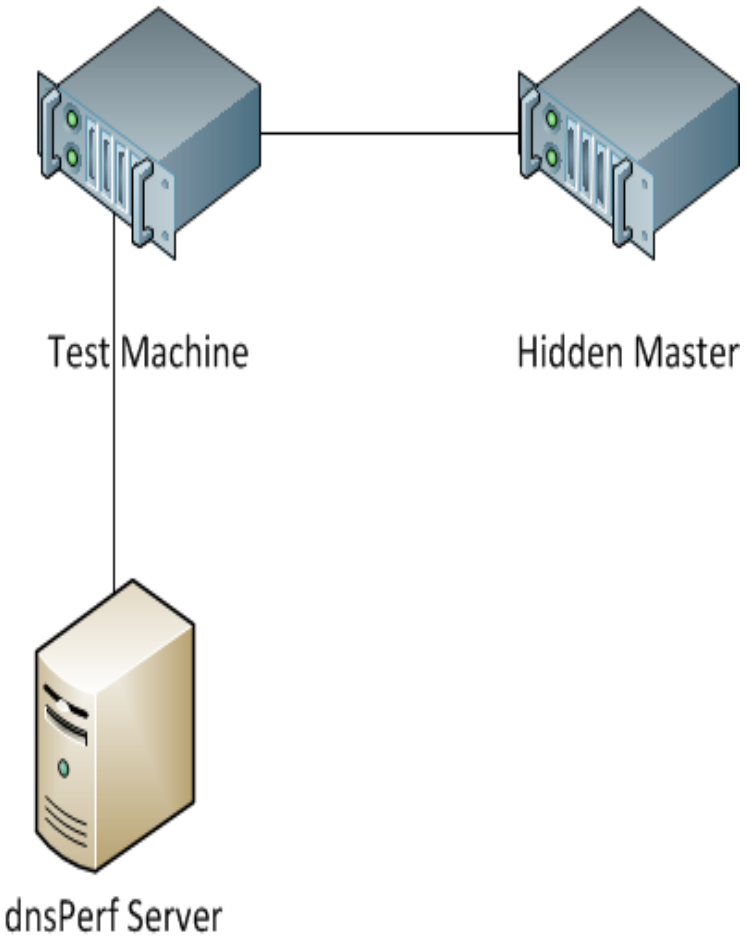
- Reproduce real life scenario:
  - New customer onboarding 10000 zones
- Replicate interactions with provisioning process
  - Zones added in batches
- Configuration of name servers would be **naïve**
  - Highlight lack of available information on this topic

TESTING PHILOSOPHY

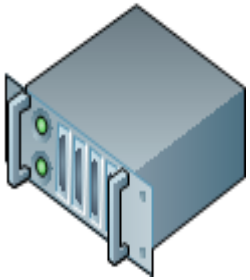
- Query performance would be tested, but it would not be the focus
  - Testing of query performance would not be thorough
  - Tests used to establish basic baseline for query performance in test set up
  - Same tests would be rerun while zones were added to observe any change

TEST PROCESS

- Start with *X* zones on test machine
  - *X*: 100K, 500K, 1M
- Add 10000 new slave zones in batches
  - Batch sizes: 100, 500, 1000, 5000, 1000
- Test both written configs and dynamic adds
- Captured how long:
  - Each add command took
  - Adding the 10000 zones took
- dnsPerf run against test machine



TEST HARDWARE



- Test Machine/Master server
  - CISCO UCSC-C220-M3L
  - 32GB memory
  - 241GB storage
  - Quad core
    - Intel(R) Xeon(R) CPU E5-2609 0 @ 2.40GHz

TEST HARDWARE

- dnsPerf server
  - Virtual Machine
  - 2GB memory
  - 58GB storage
  - Single core
    - Intel(R) Xeon(R) CPU X5670 @ 2.93GHz

TESTING NOTES – ZONES

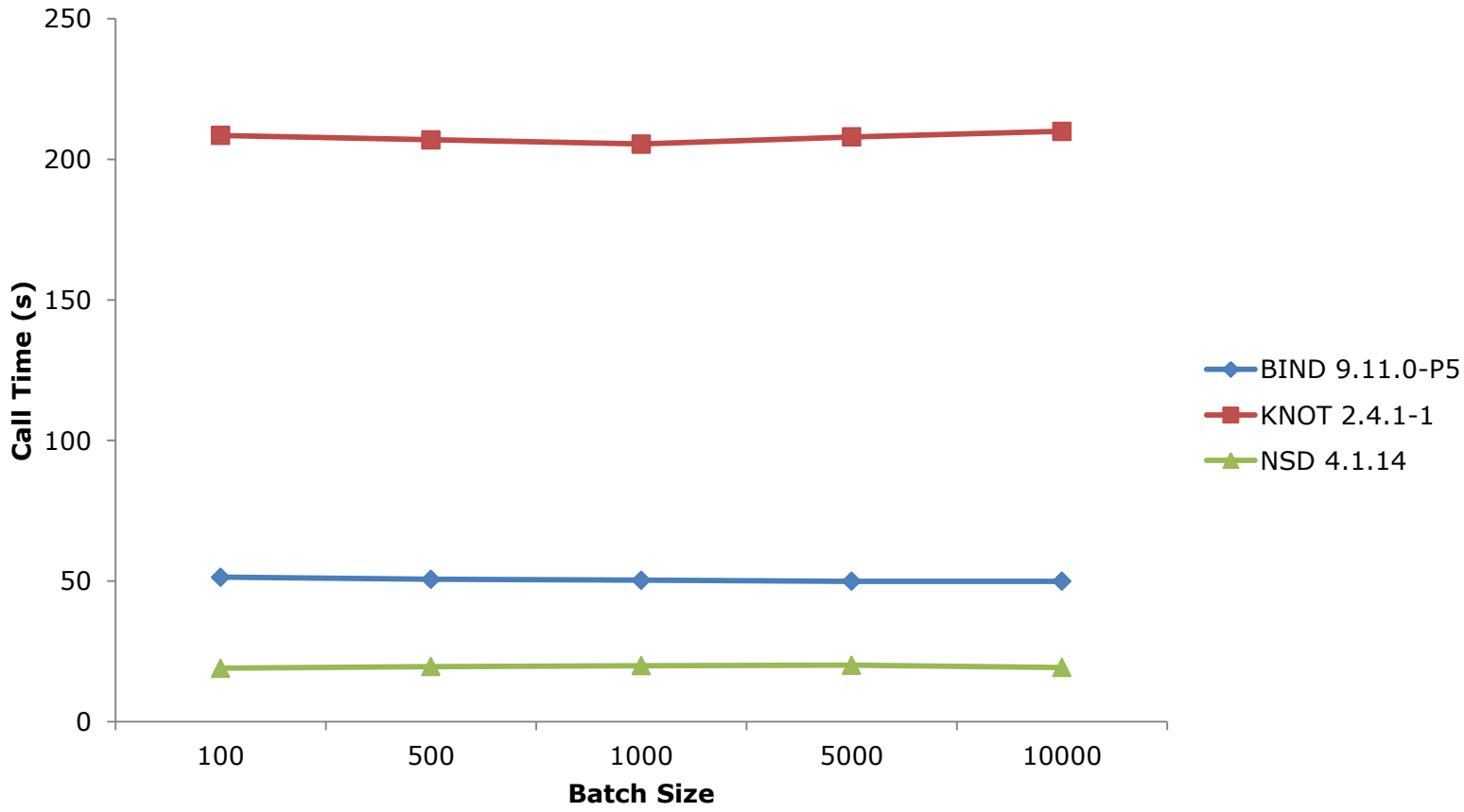
- Names generated as unique permutations of 12 words
  - 6-10 characters in length
- Mixture of A, AAAA, NS, TXT and MX records
- Concerned with size over content
  - 0.25% 2.2M
  - 34.5% 12K
  - 65.25% 500b

TEST RESULTS

1M Zones – Written Config

ADD COMMAND TIMINGS

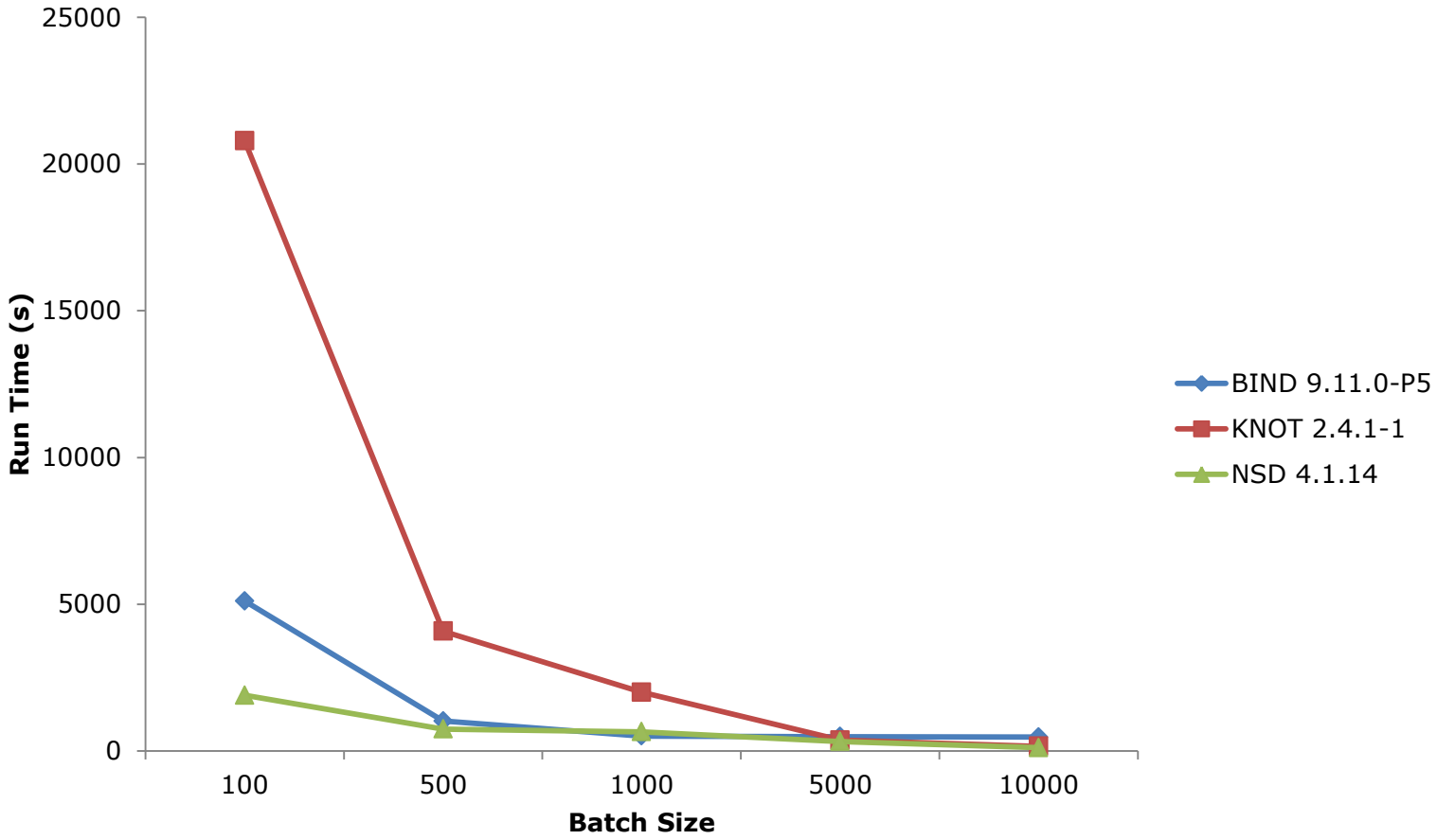
Average time taken to run commands used for adding a single batch of zones with written configs



*Knot recommends using binary database for large configs

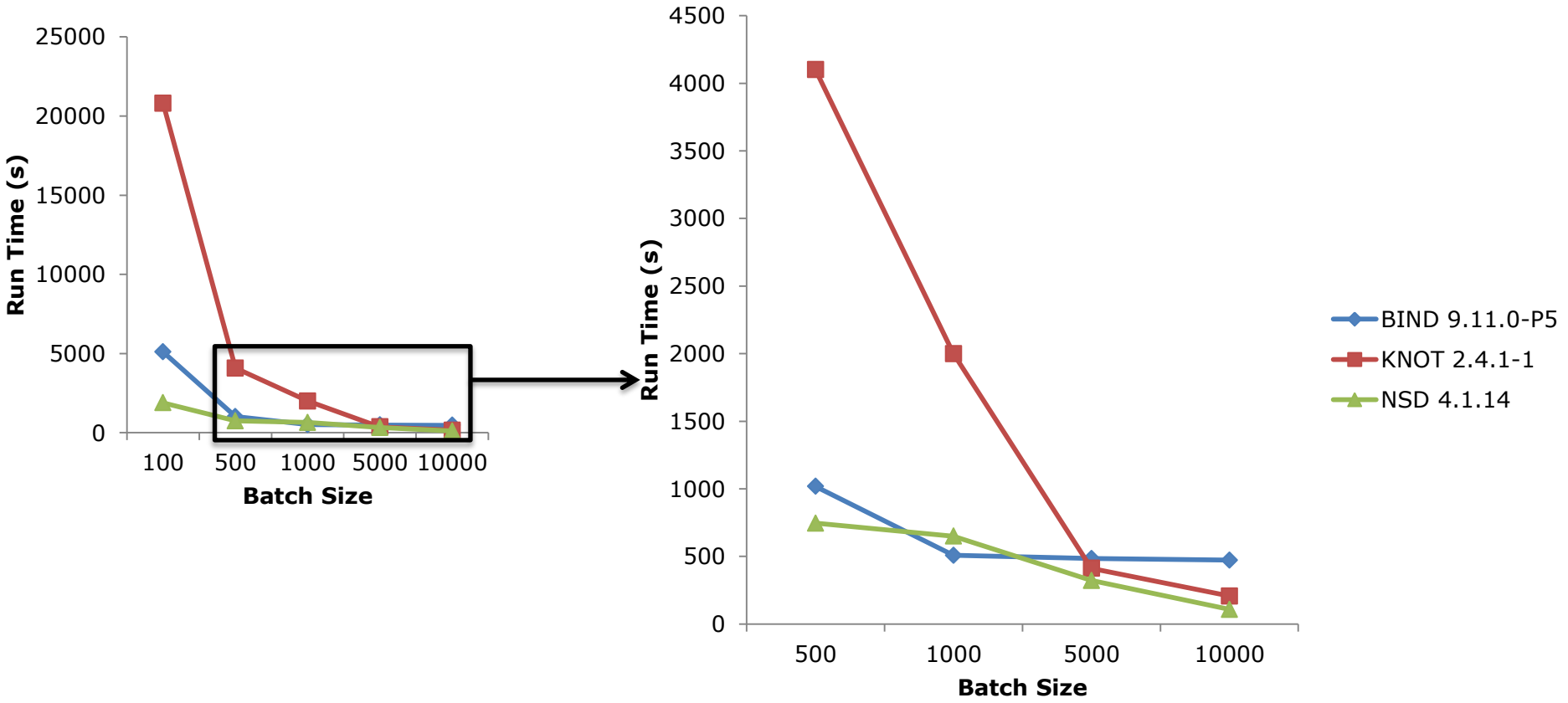
TIME TO ADD 10000 ZONES

Average time taken to add 10000 zones using written configs in batches.



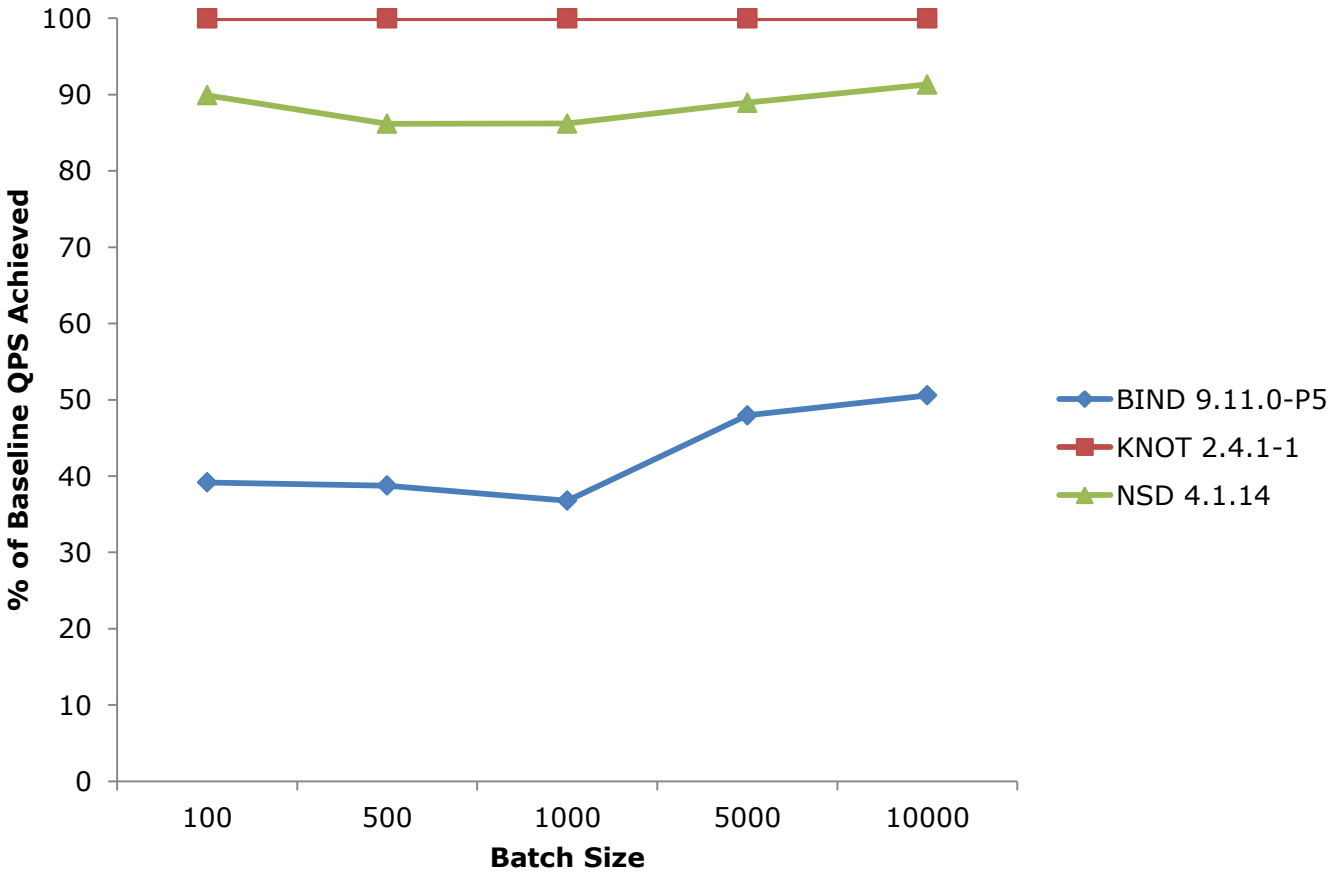
TIME TO ADD 10000 ZONES

Average time taken to add 10000 zones using written configs in batches.



QUERY PERFORMANCE

% of baseline QPS achieved while 10000 zones were added using written configs in batches.

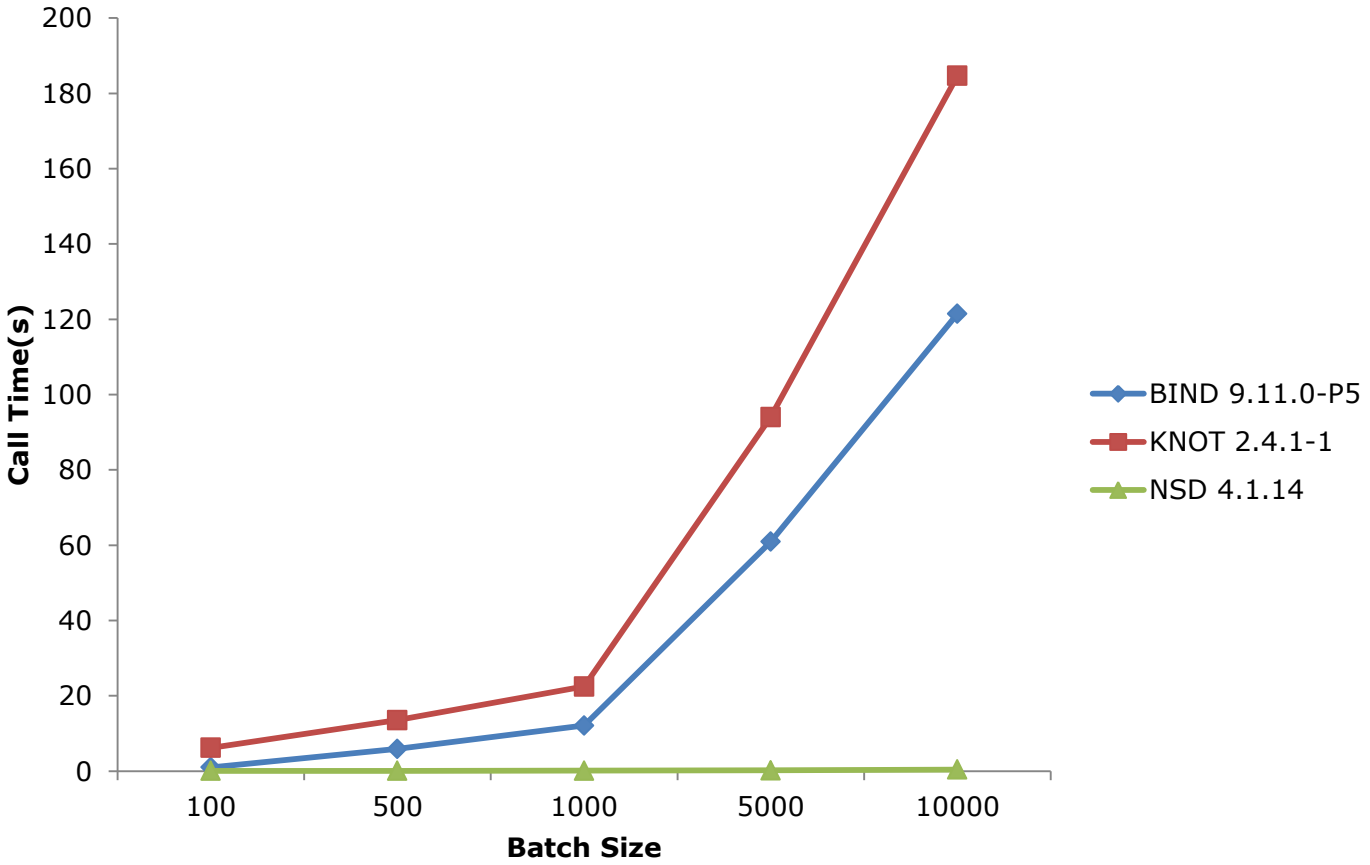


TEST RESULTS

1M Zones - Dynamic Adds

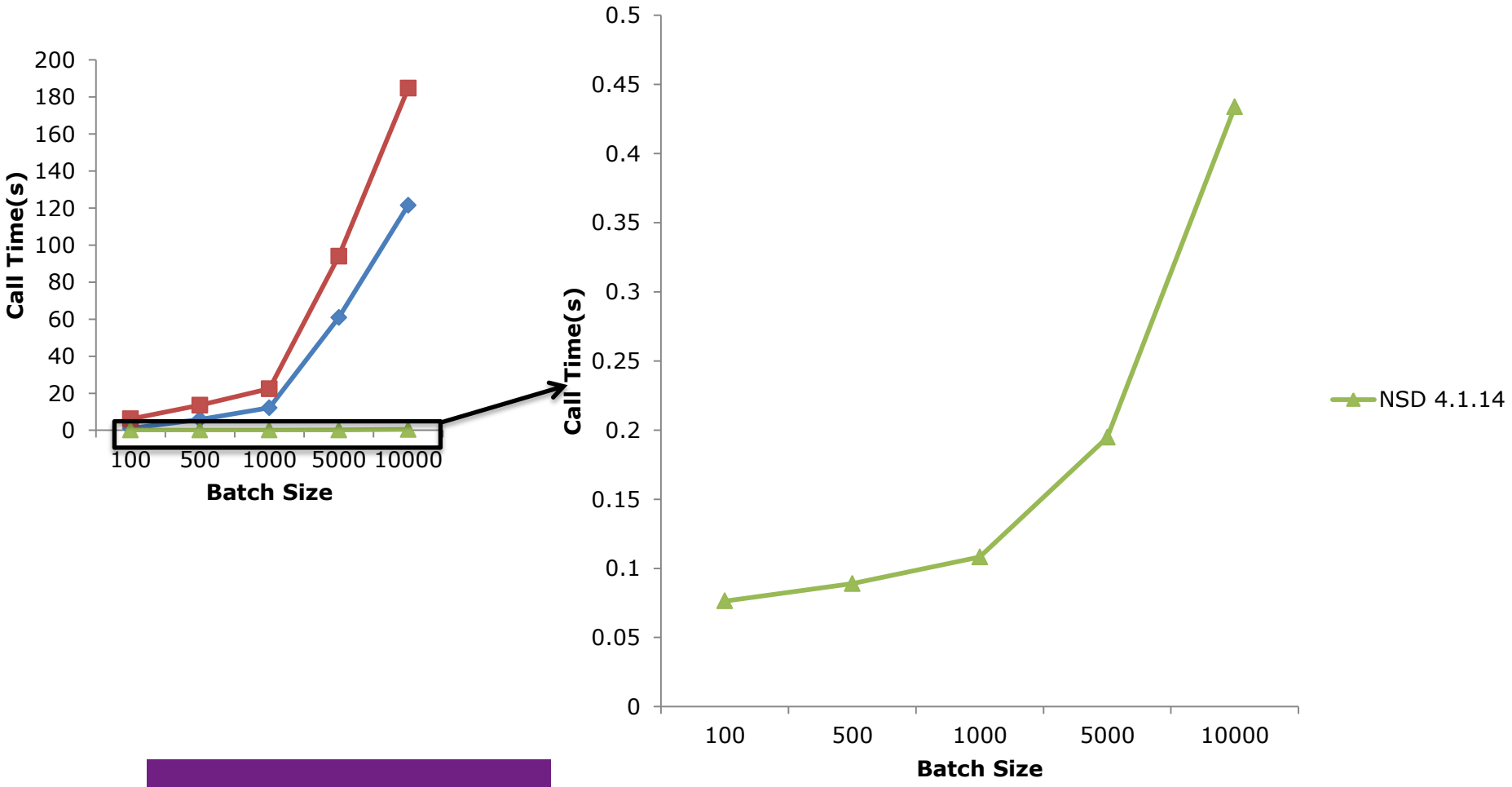
ADD COMMAND TIMINGS

Average time taken to run commands used for adding a single batch of zones dynamically.



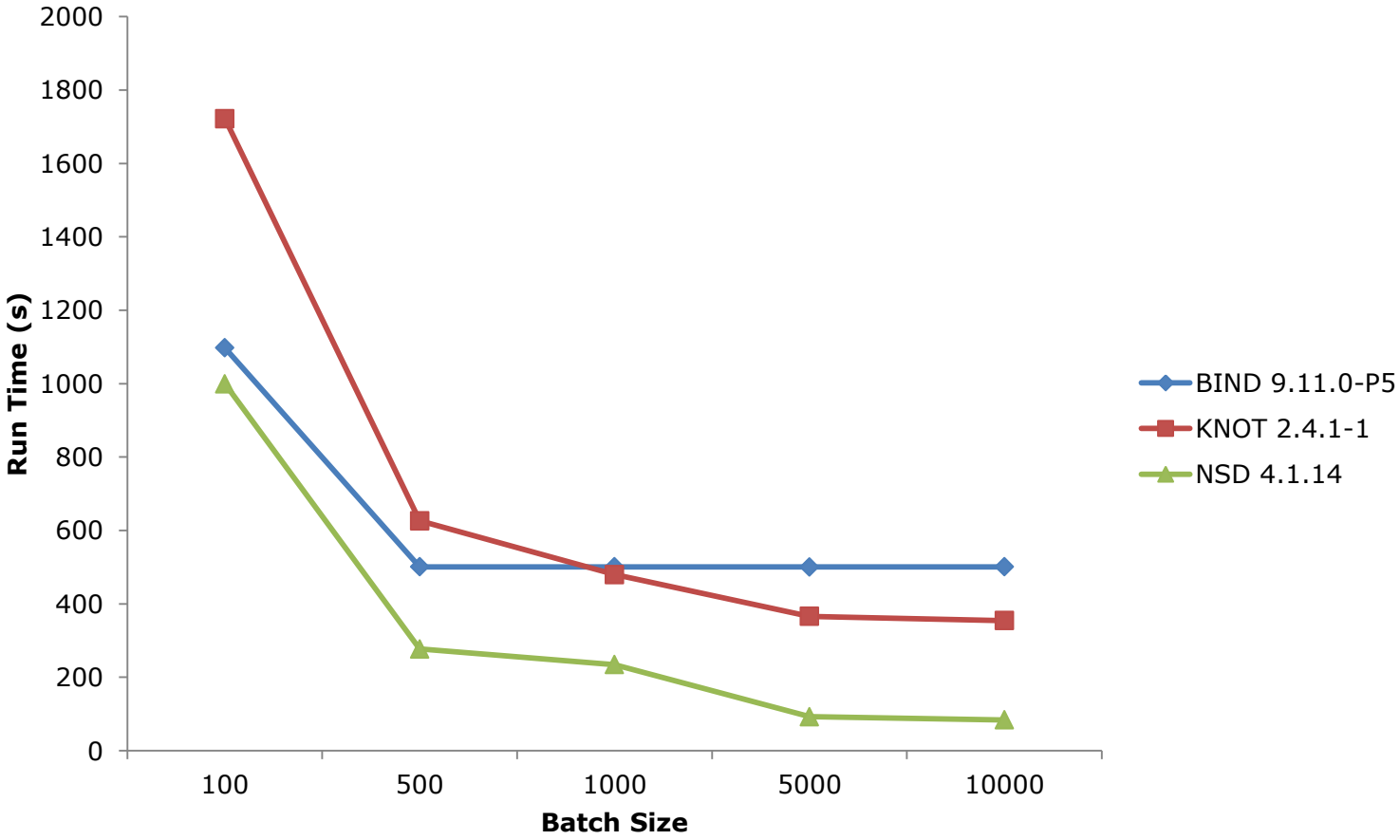
ADD COMMAND TIMINGS

Average time of system call used to add zones dynamically by batch size.



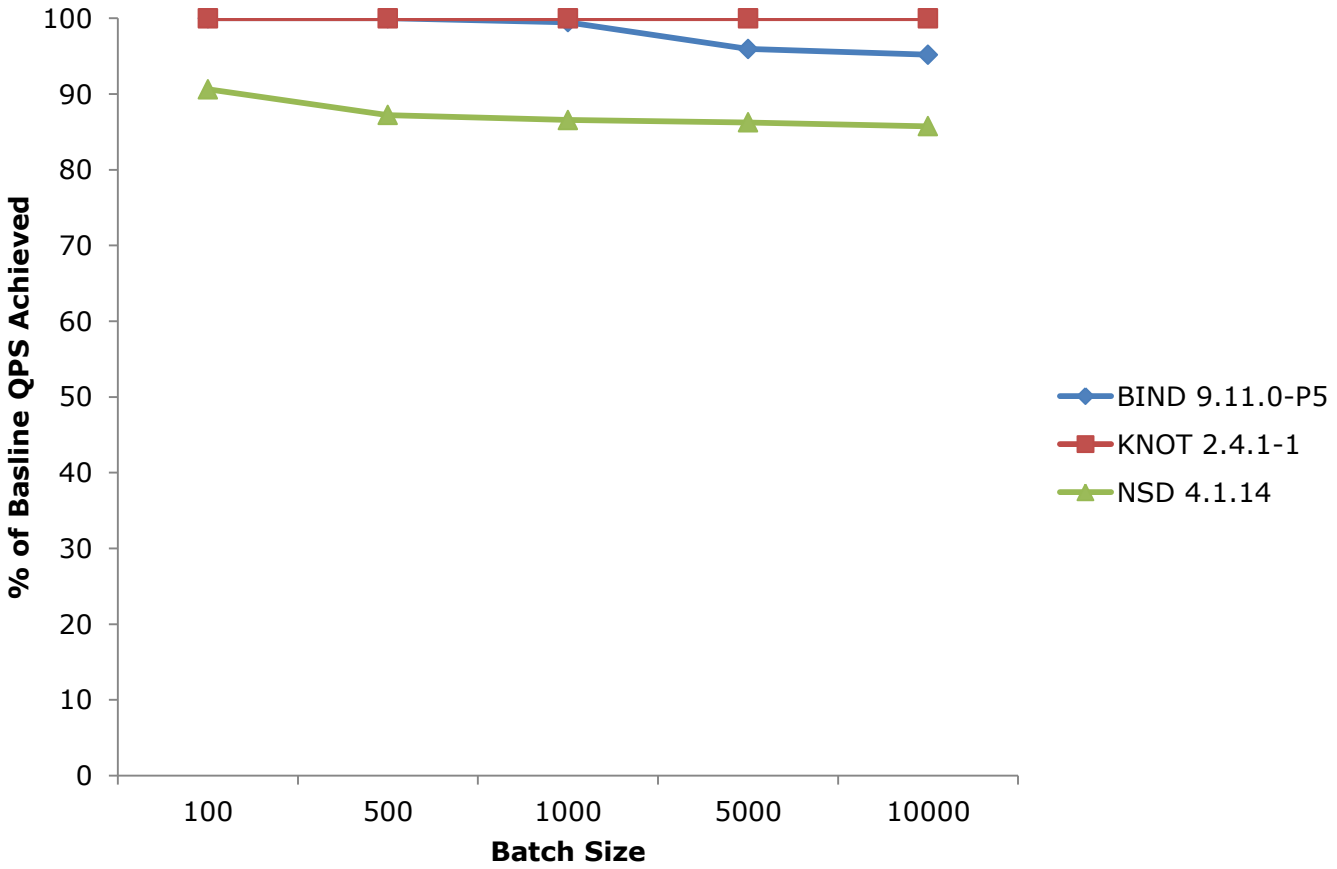
TIME TO ADD 10000 ZONES

Average time taken to add 10000 zones dynamically in batches.



QUERY PERFORMANCE

% of baseline QPS achieved while 10000 zones were dynamically added in batches.



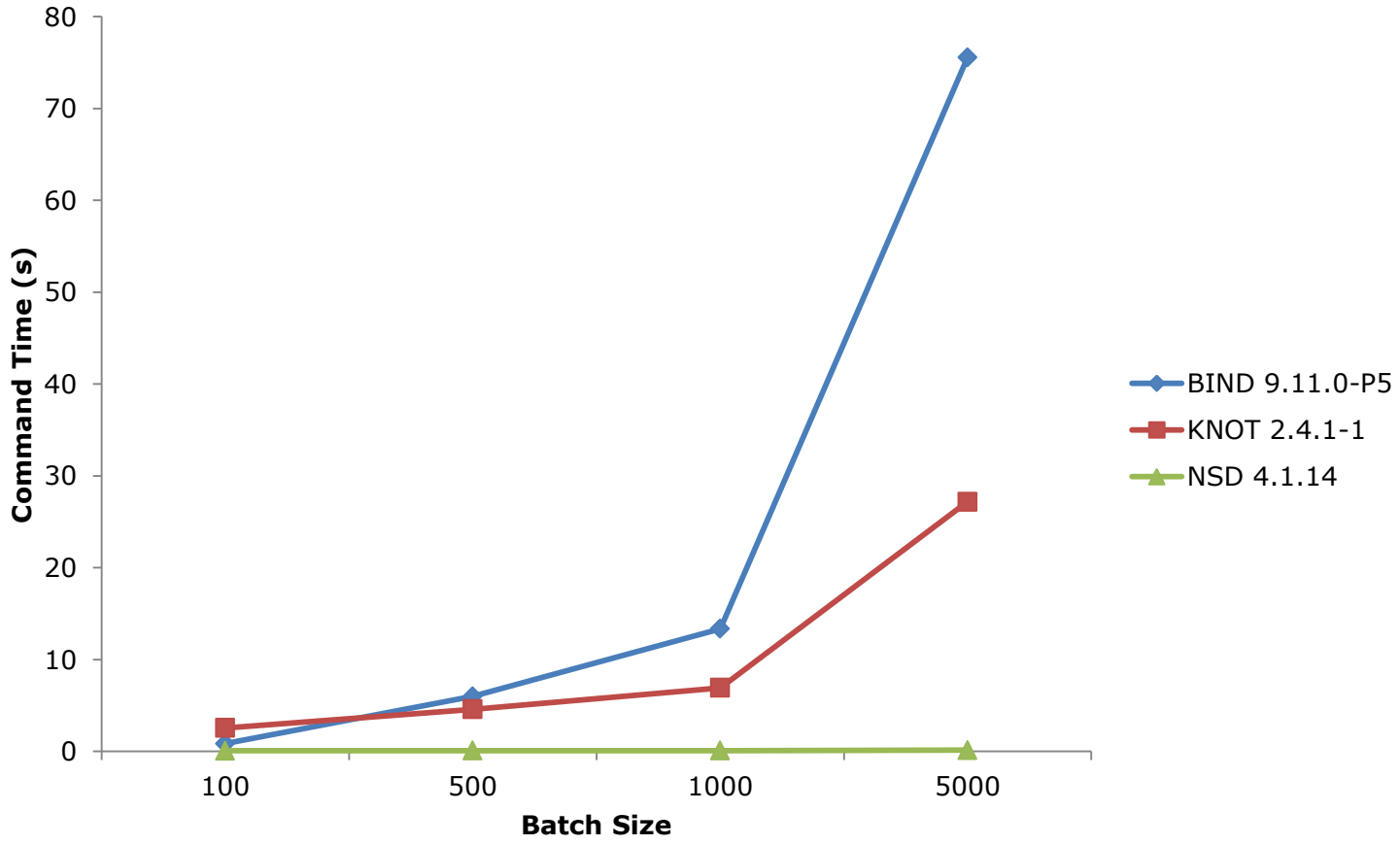
DELETE TESTS

DELETE TESTS

- Started with 500K master zones
- Dynamic deletes to remove 5000 zones
  - Batches of: 100, 500, 1000, 5000
- Monitored:
  - Length of each delete call
  - How long it took for each batch to be deleted
  - Query performance

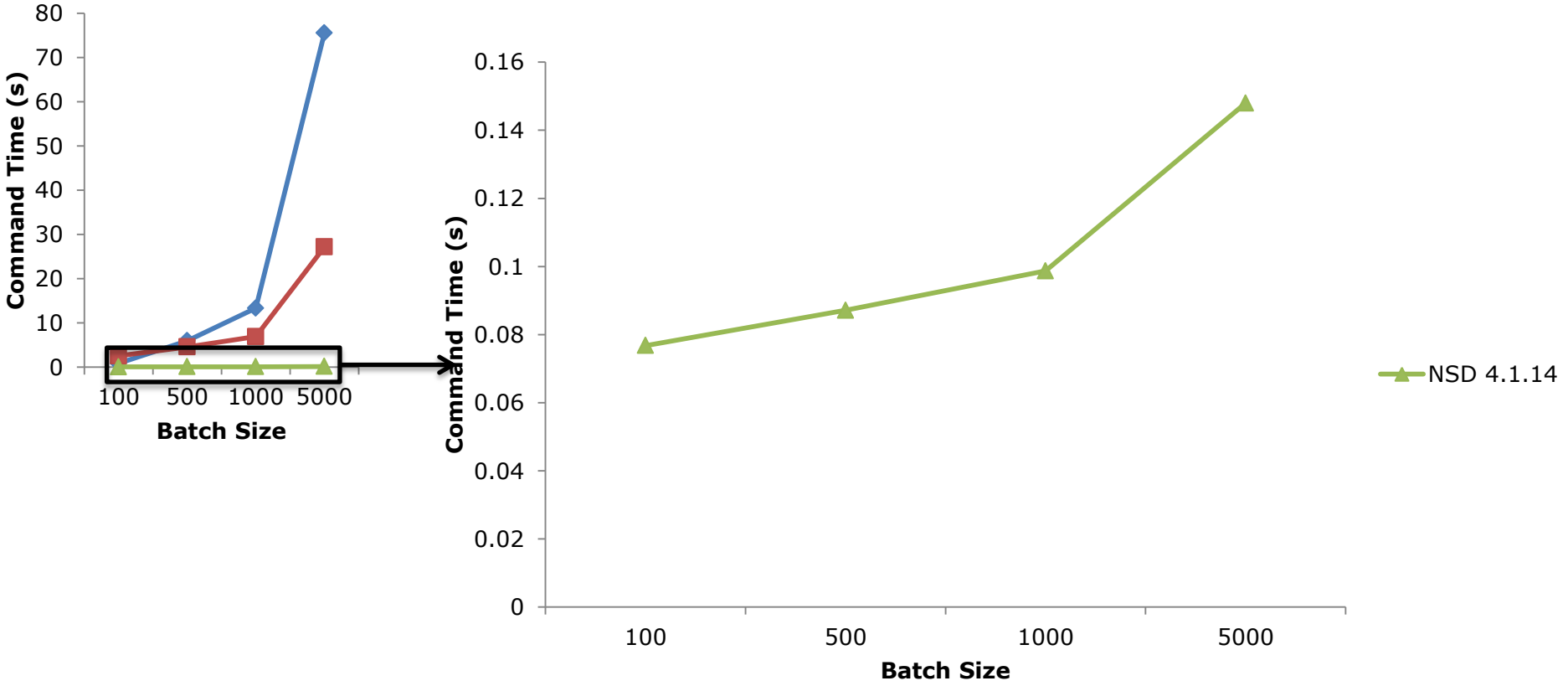
DELETE COMMAND TIMINGS

Average time taken to run commands used for deleting a single batch of zones dynamically.



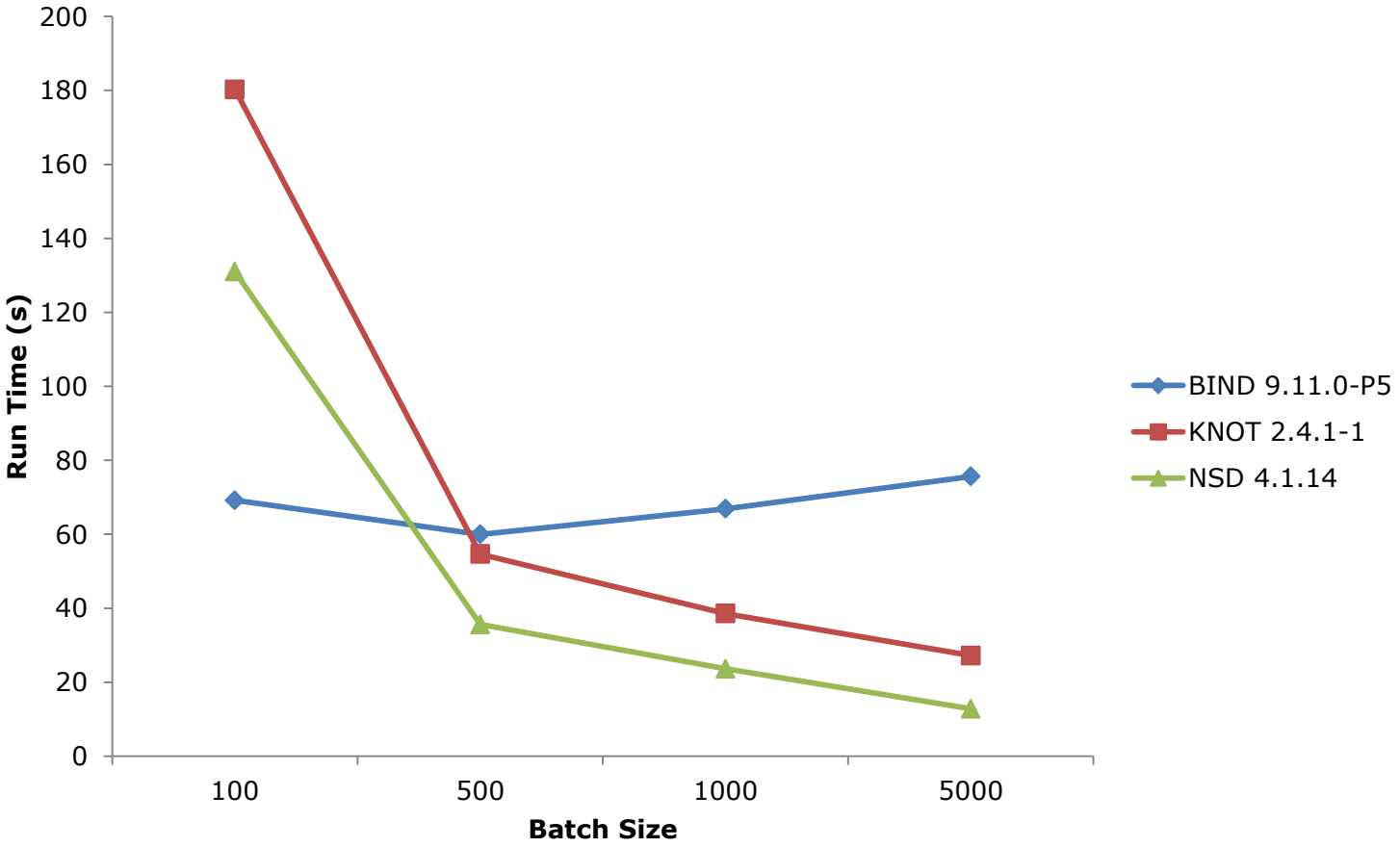
DELETE COMMAND TIMINGS

Average time of system call used to delete zones dynamically by batch size.



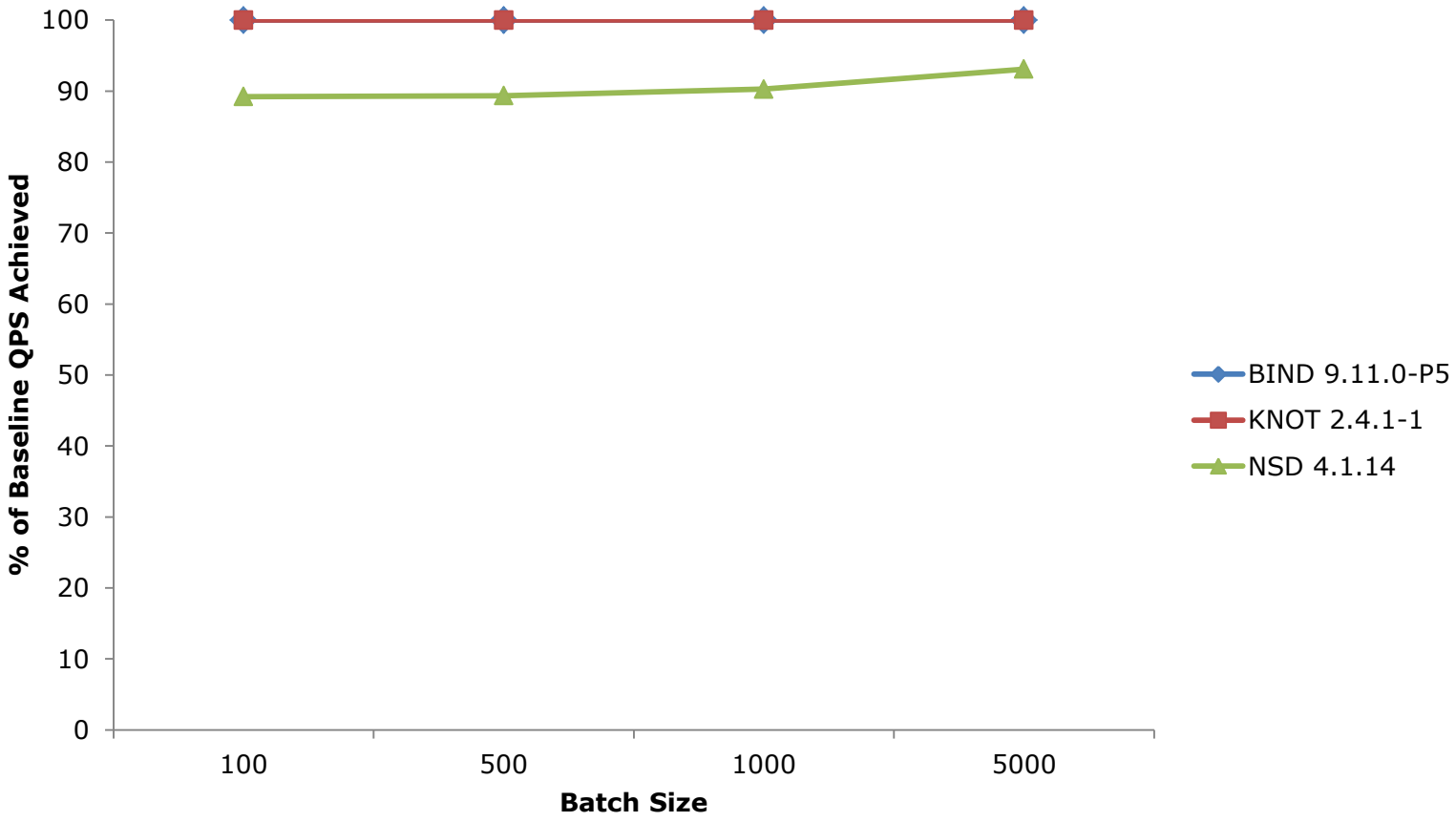
TIME TO DELETE 5000 ZONES

Average time taken to delete 5000 zones dynamically in batches.



QUERY PERFORMANCE

% of baseline QPS achieved while 5000 zones were dynamically added in batches.



CONCLUSION

TAKE AWAY

- Dynamic > Written configs:
  - Speed
  - Query impact minimized
  - Less “custom” IO for provisioning
- Written configs:
  - Larger batch adds when possible

NEXT STEPS

- Continue the conversation:
  - Have experience/questions, talk to me about it
- Further testing:
  - Re-run tests with performance tuned configuration
  - In depth look at performance for master configuration
  - Test other name server software
- D-Zone Anycast DNS:
  - Diversification of name server software

THANK YOU

QUESTIONS/COMMENTS

evan.thompson@cira.ca