Memcached vs Redis

How does performance of
Memcached compare to *Redis*
on a common feature set

Recap - Object cache

- Application running a large HashMap
- Store results of expensive/frequently accessed data
- Support for *get*, *set*, *mget*, *remove*

- Can be queried over the network
- Distributed
- Should handle large volume of requests

- Implementations: *Memcached*, *Redis, ...*

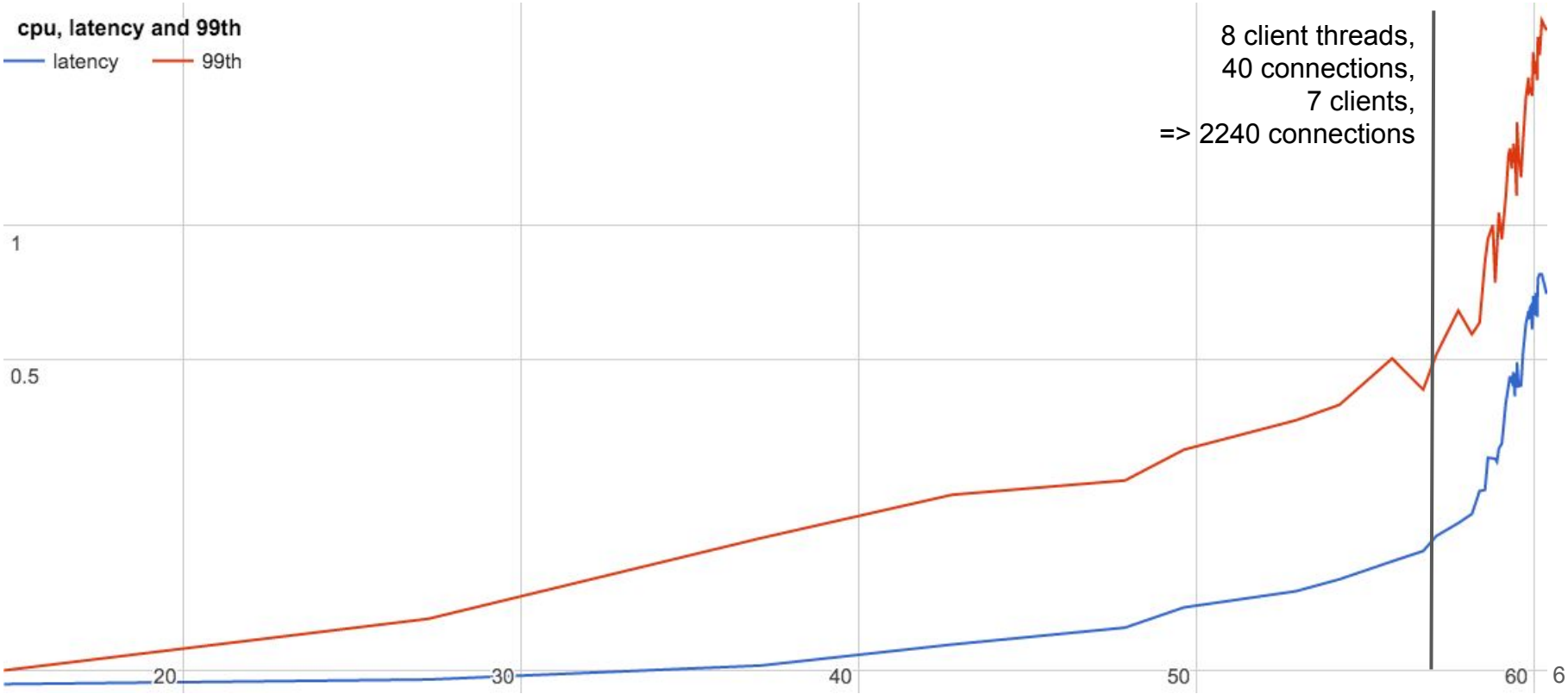
Work outline

- Stage 1
  - Configure testbed and install caches
  - Automate test execution and data collection
- Stage 2
  - Understand and tune memcached
- Stage 3
  - Run experiments on redis with the same configuration

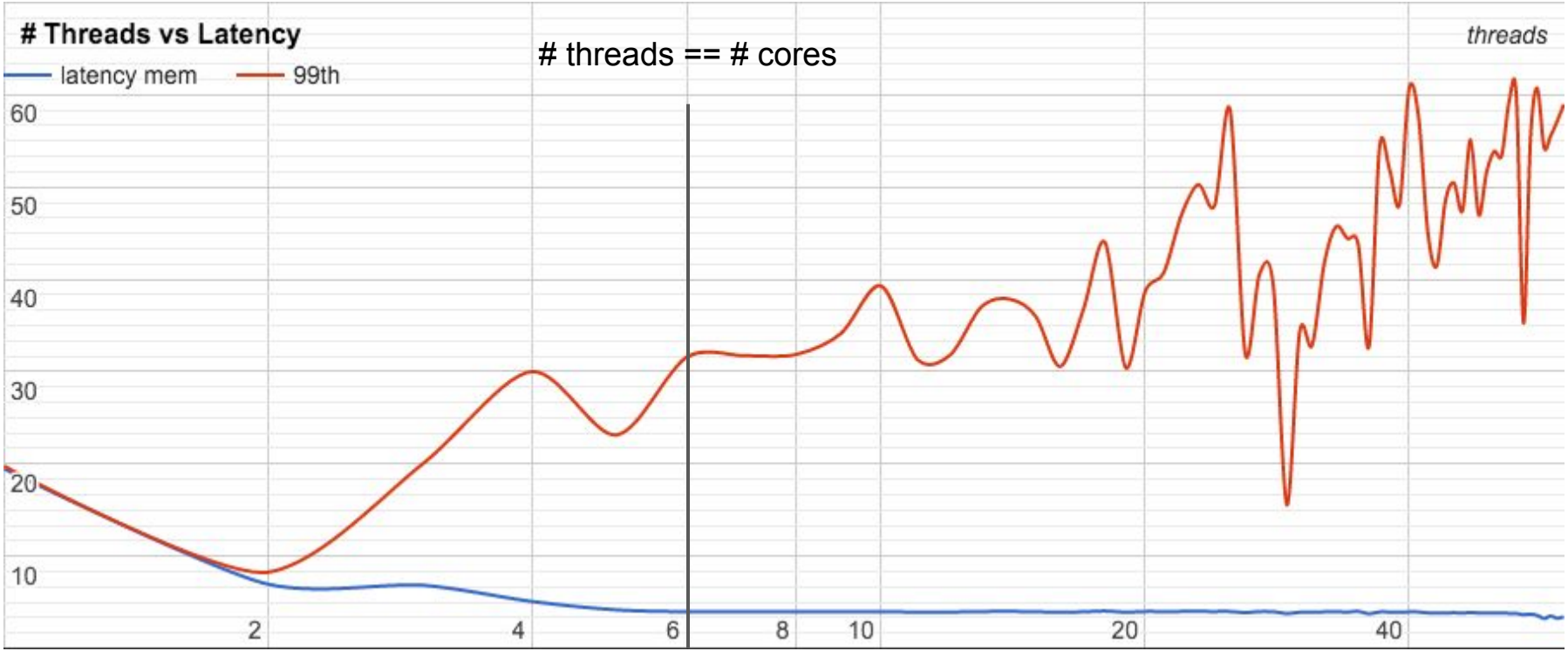
Stage 1

- 8 hosts, 1 rack, 1 Gbps link
  - 1 server
  - 7 clients
- Parallel SSH and Data Collection
  - Server - CPU, Memory, Network throughput
  - Client - get/set latency, mean latency, operations per second, request latency distribution

Stage 2 - CPU vs Latency (4 memcached threads)



Stage 2 - Scaling memcached threads



Stage 2 - CPU cycles

Stage 2 - Next steps

- Experiments with packet queuing vs throughput and latency
- Effect of *value* size on throughput
- Skewed distribution of keys
- Effect of TCP vs UDP connections

Stage 3

- Re-run experiments on redis
- Compare performance
- Analyse how additional features of redis impact performance

Expectation:

- Redis will perform worse at scale
- Redis performance will further degrade with features such as saving data to disk

Timeline

- Stage 1 (setup, configuration, parallel benchmarking and data collection):
  - Completed in weeks 1-4
- Stage 2 (tuning memcached, system stability)
  - Server utilization vs throughput (completed)
  - Scaling memcached across threads/processes (completed)
  - Validation of data and establishing a baseline for Redis (in-progress)
  - Skewed distributions, TCP vs UDP (to-do)
  - Aiming to complete by December 14
- Stage 3 (tuning redis, benchmarking and comparison)
  - Aiming to run tests and record results by December 31
- Further
  - Analysis of the data further and iteration on findings

Q & A