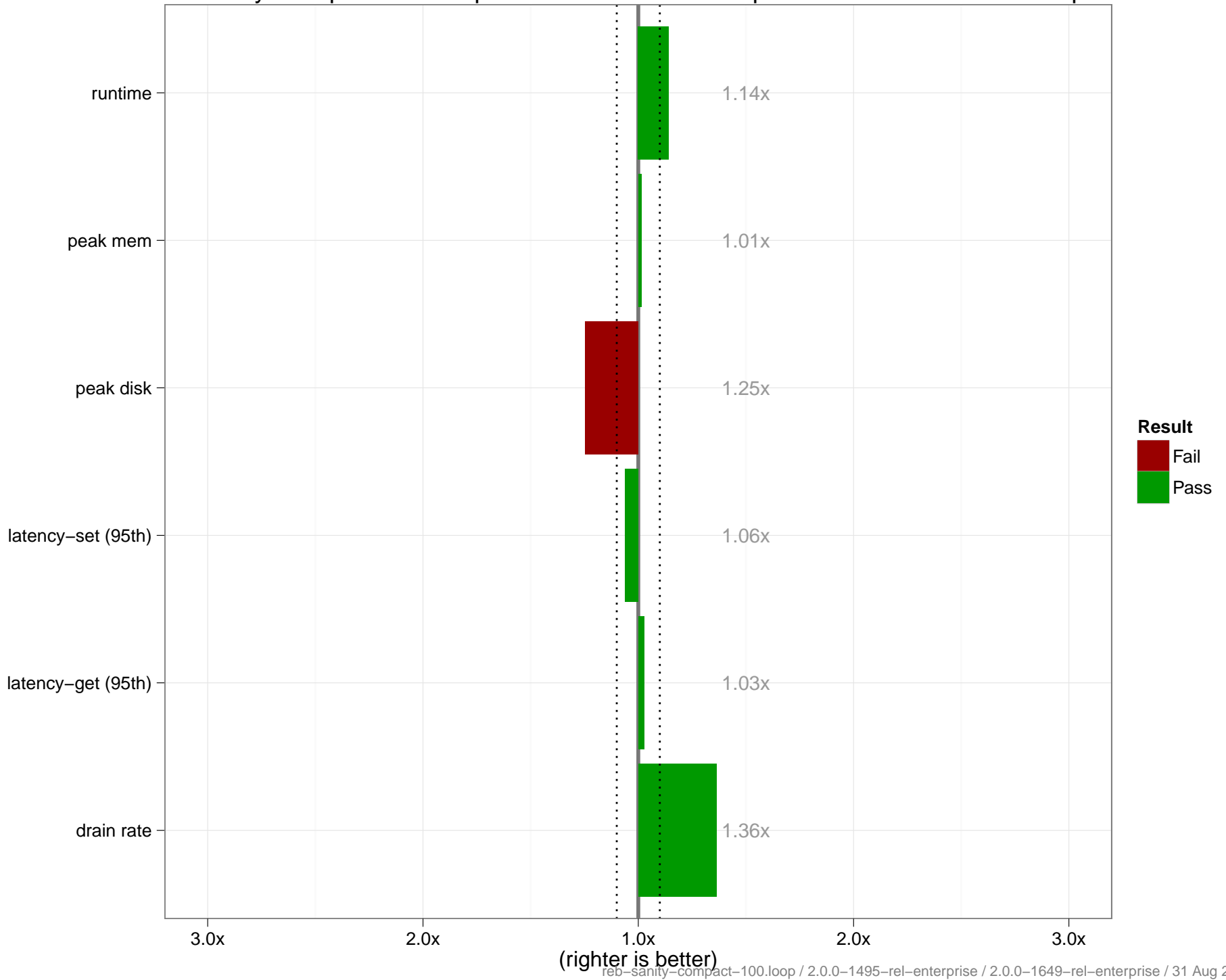
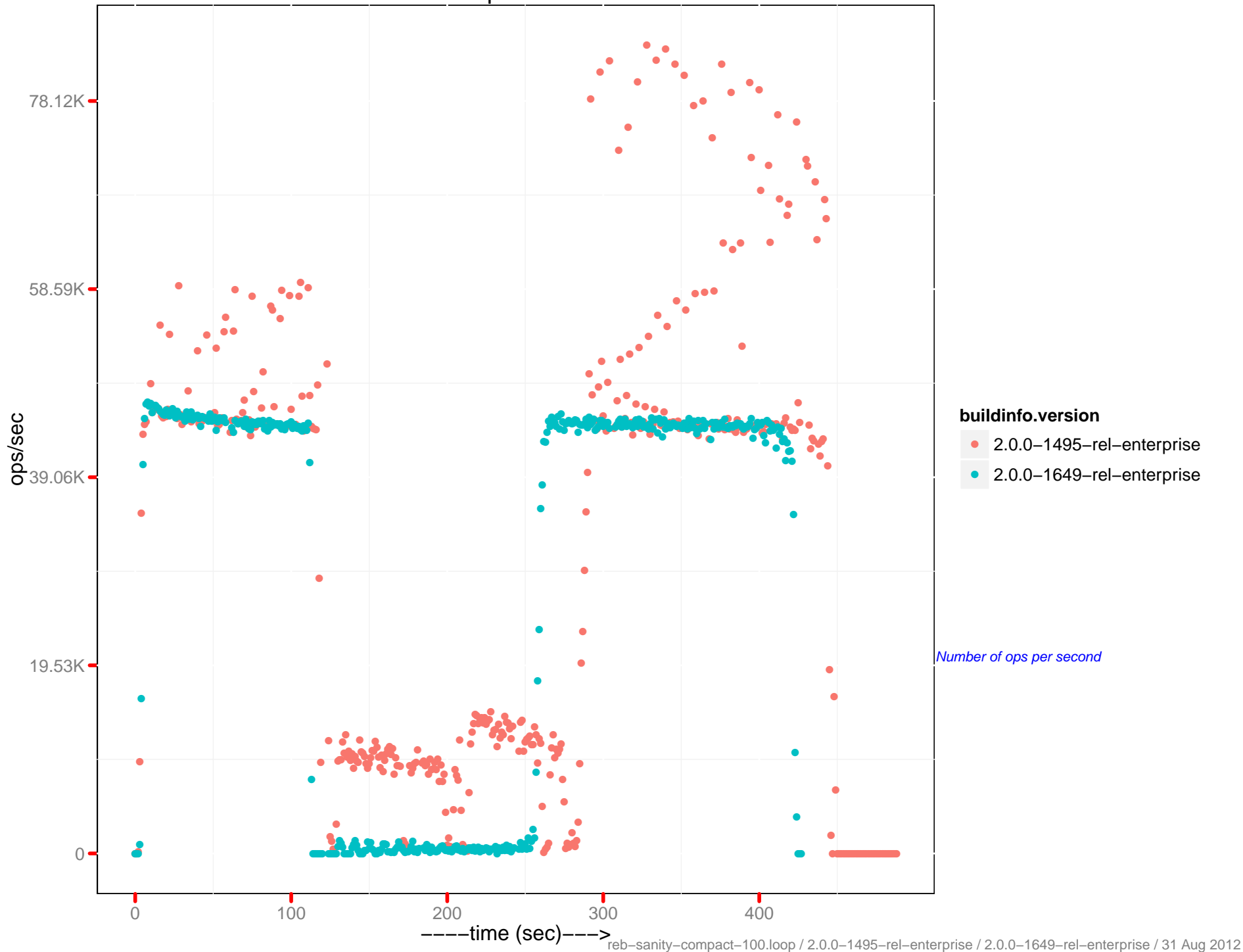


reb-sanity-compact-100.loop : 2.0.0-1495-rel-enterprise : 2.0.0-1649-rel-enterprise



	2.0.0 – 1495	2.0.0 – 1649
<i>Runtime (in hr)</i>	0.14	0.12
<i>Avg. Drain Rate</i>	11.59K	15.80K
<i>Peak Disk (GB)</i>	12.09	15.05
<i>Peak Memory (GB)</i>	1.78	1.75
<i>Avg. OPS</i>	34.08K	30.93K
<i>Avg. mem memcached (GB)</i>	1.4	1.4
<i>Avg. mem beam.smp (MB)</i>	329.57	316.22
<i>Avg. CPU rate (%)</i>	34.72	37.86
<i>Latency-get (90th) (ms)</i>	29.49	29.39
<i>Latency-get (95th) (ms)</i>	34.07	33.2
<i>Latency-get (99th) (ms)</i>	48	45.32
<i>Latency-set (90th) (ms)</i>	29.34	30.71
<i>Latency-set (95th) (ms)</i>	33.46	35.51
<i>Latency-set (99th) (ms)</i>	47.66	52.78
<i>Latency-query (80th) (ms)</i>	NA	NA
<i>Latency-query (90th) (ms)</i>	NA	NA
<i>Latency-query (95th) (ms)</i>	NA	NA
<i>Latency-query (99th) (ms)</i>	NA	NA
<i>Latency-query (99.9th) (ms)</i>	NA	NA
<i>Avg. QPS</i>	0	0
<i>Rebalance Time (sec)</i>	179.22	149.41
<i>Testrunner Version</i>	c8c7b2d	c8c7b2d

ops/sec

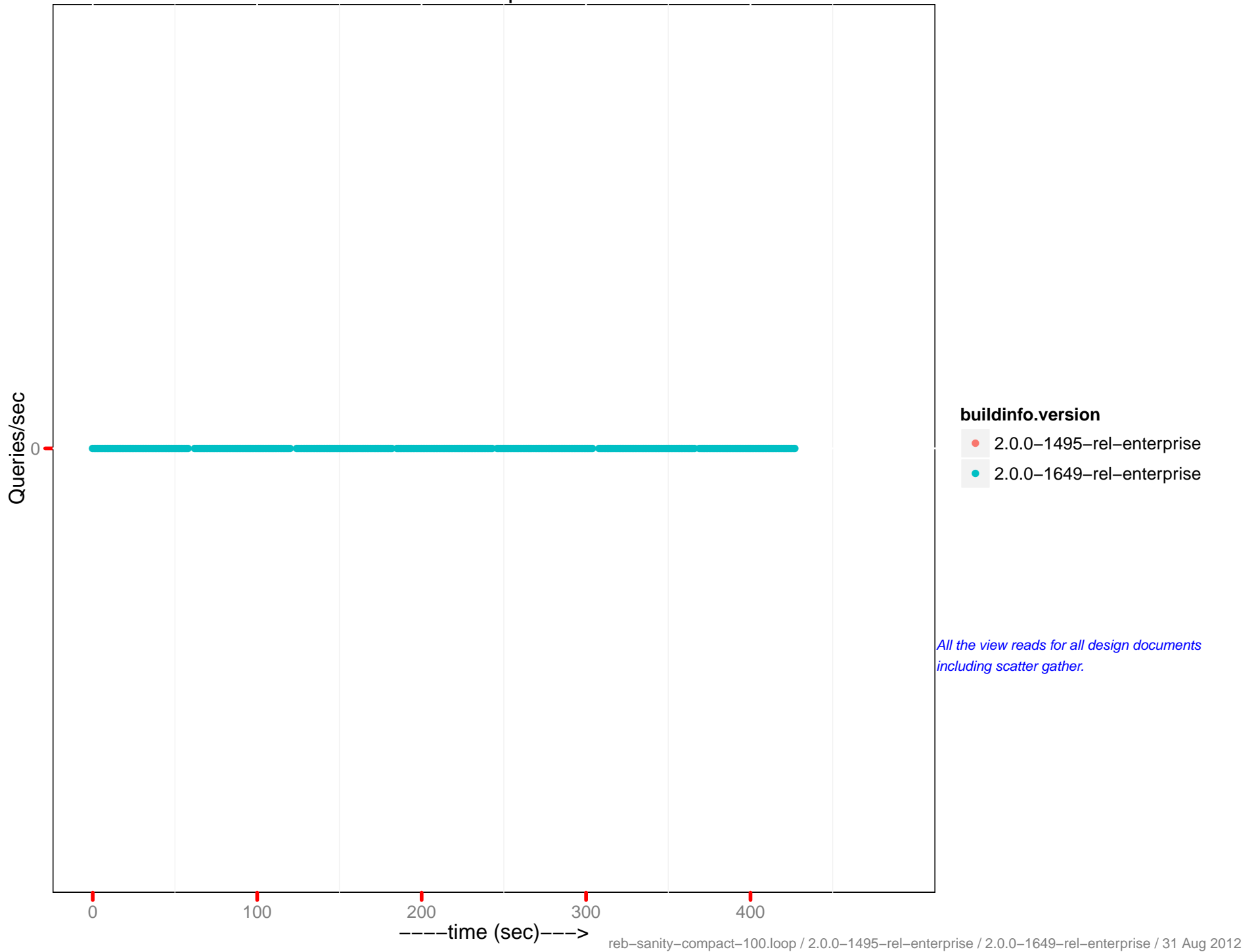


buildinfo.version

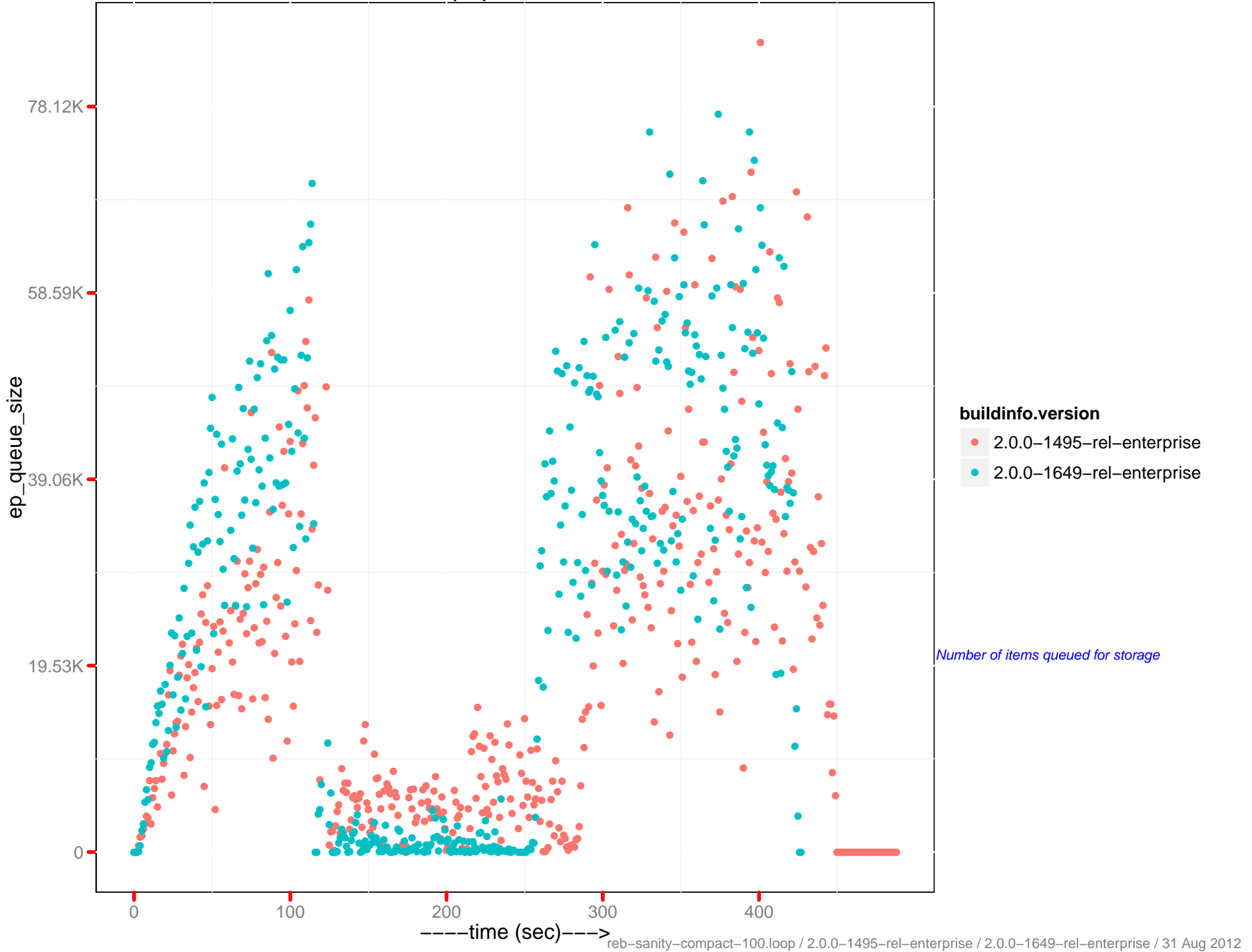
- 2.0.0-1495-rel-enterprise
- 2.0.0-1649-rel-enterprise

Number of ops per second

View read per sec.



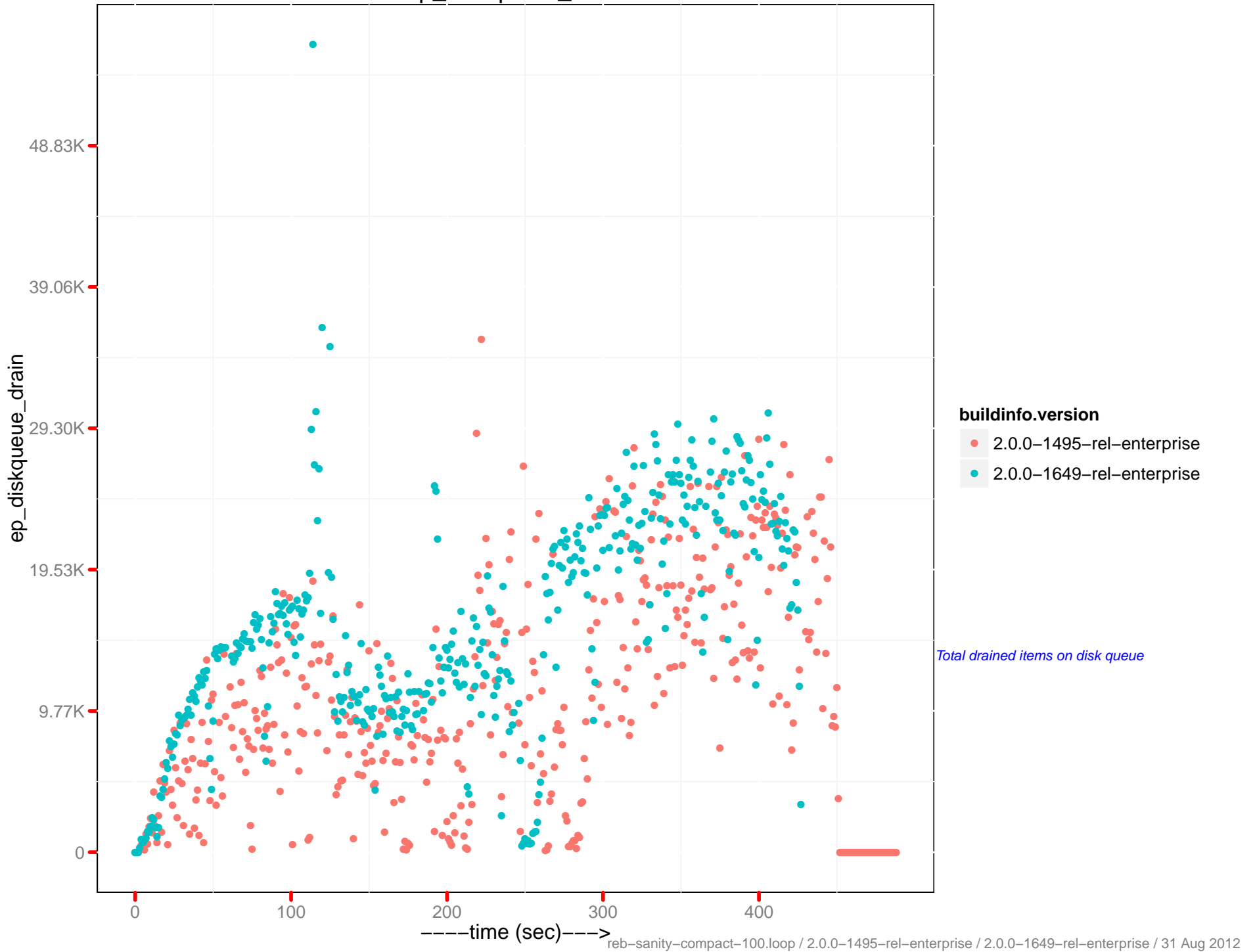
ep queue size



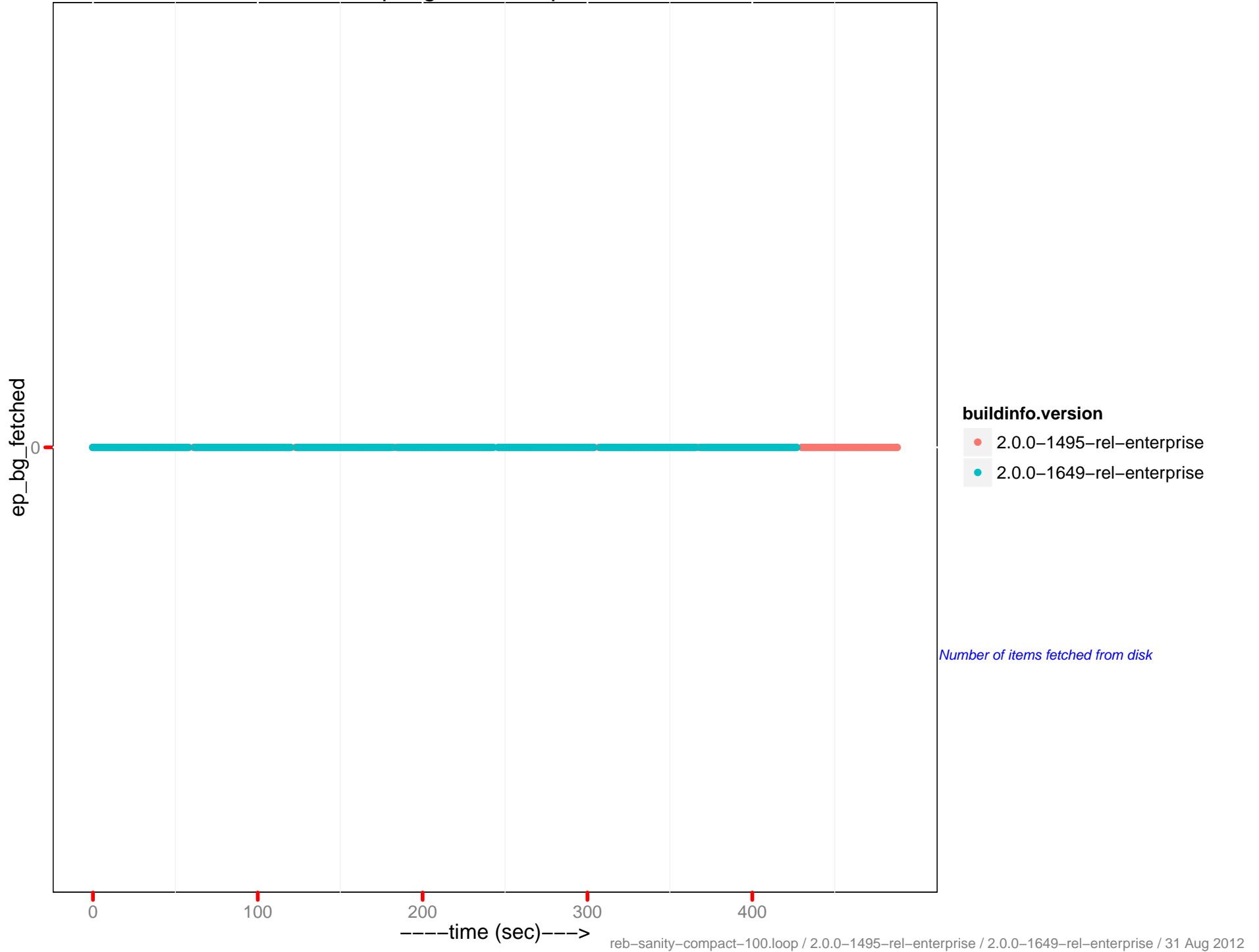
buildinfo.version
● 2.0.0-1495-rel-enterprise
● 2.0.0-1649-rel-enterprise

Number of items queued for storage

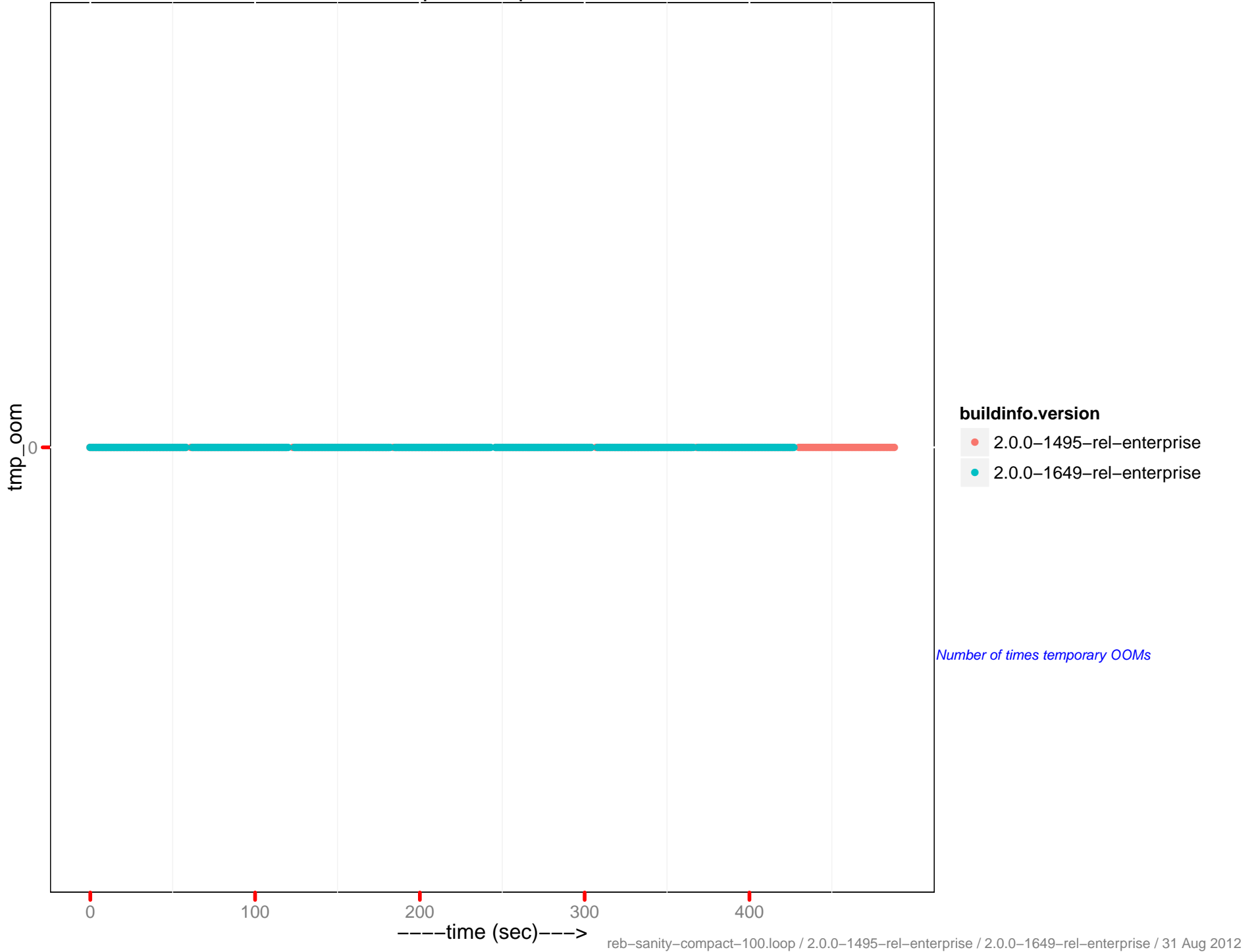
ep_diskqueue_drain



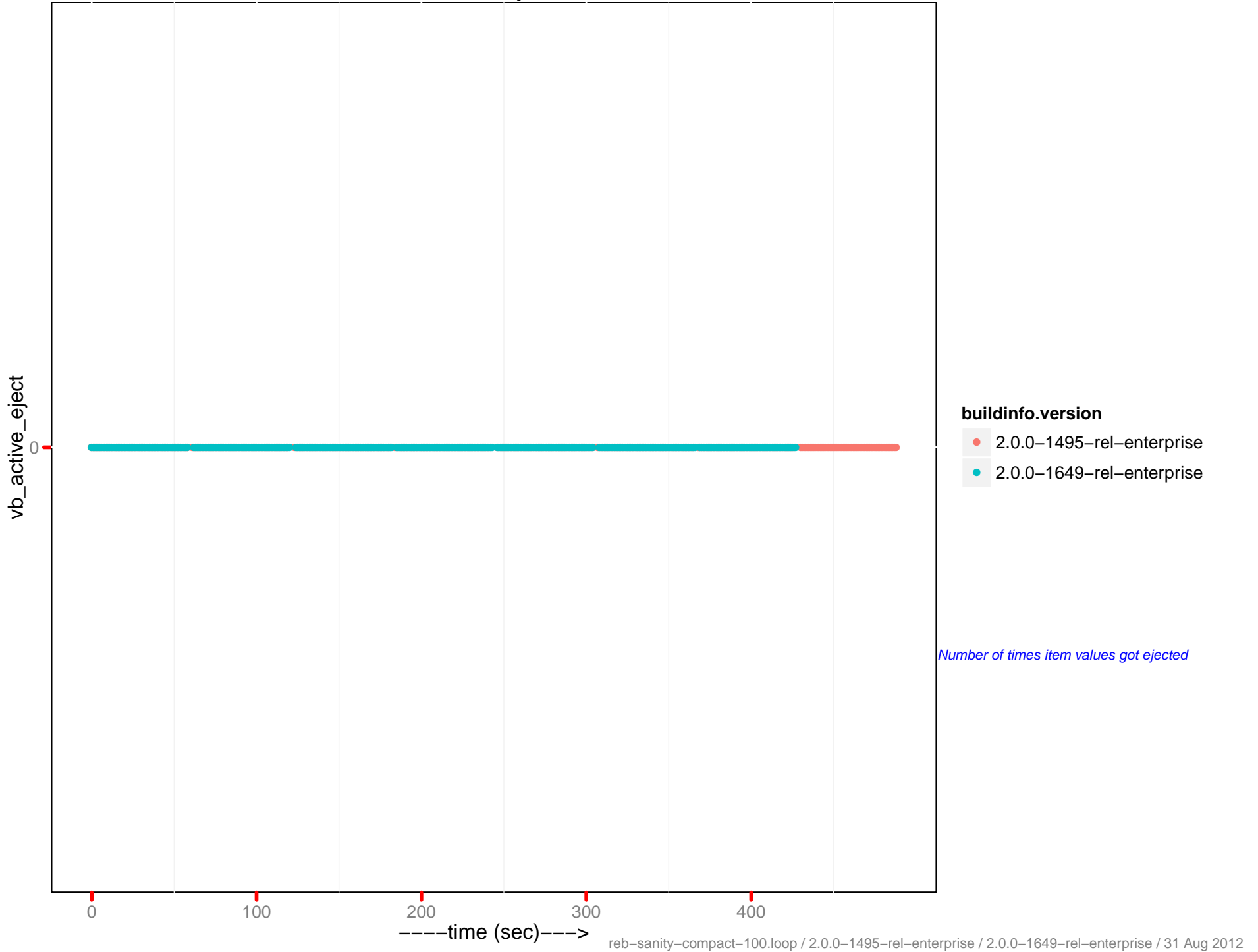
ep_bg_fetched ops/sec



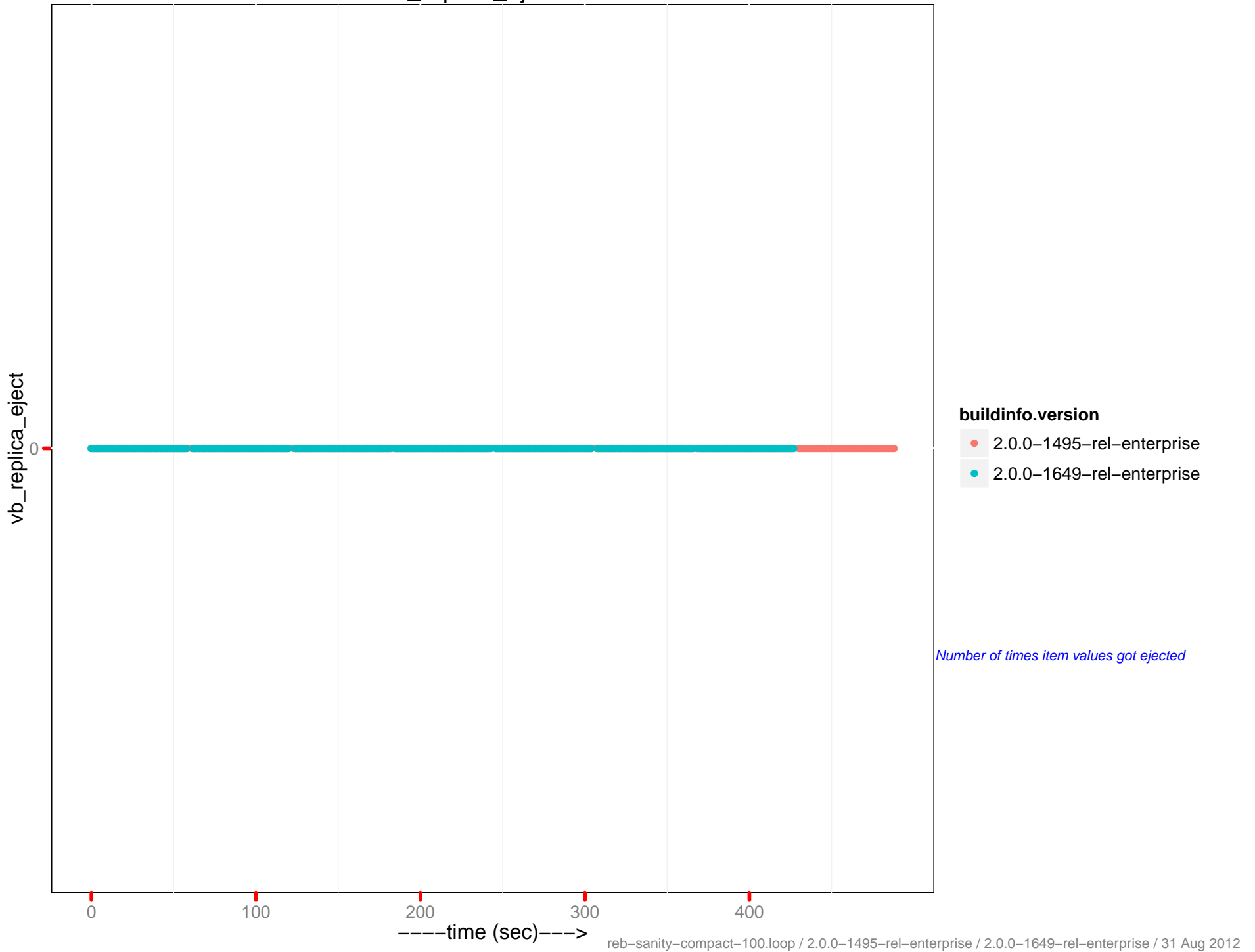
tmp_oom ops/sec



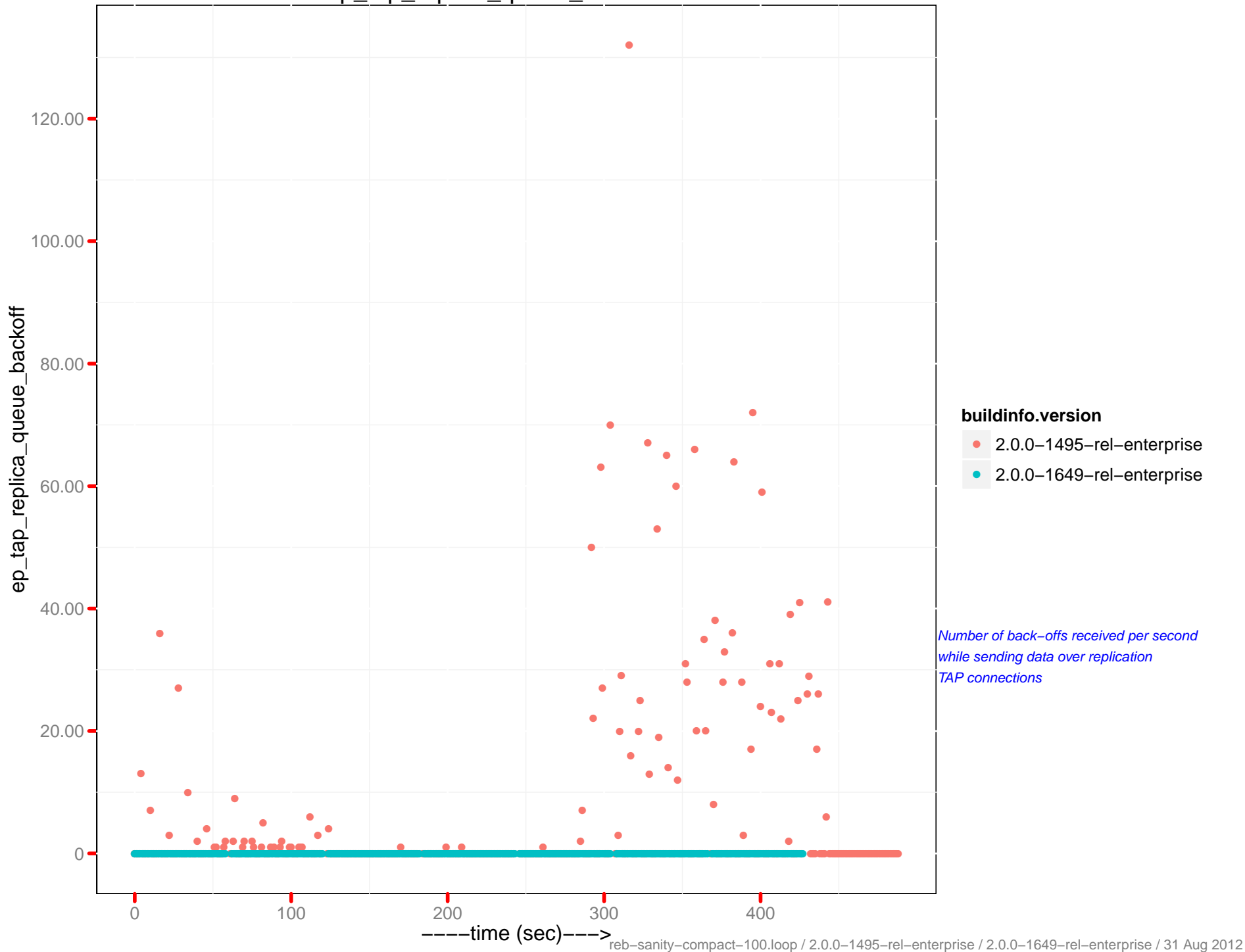
vb_active_eject/sec



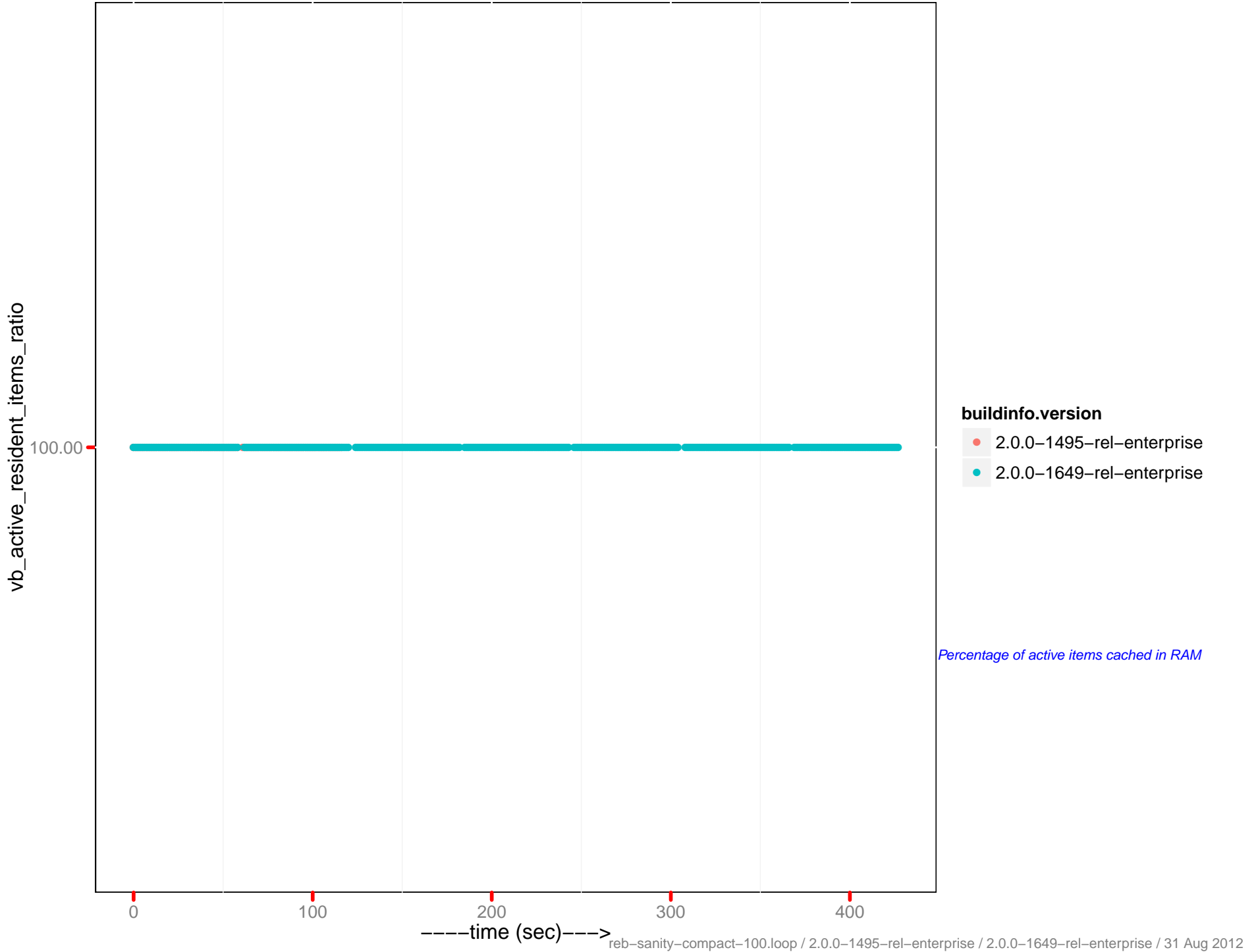
vb_replica_eject/sec



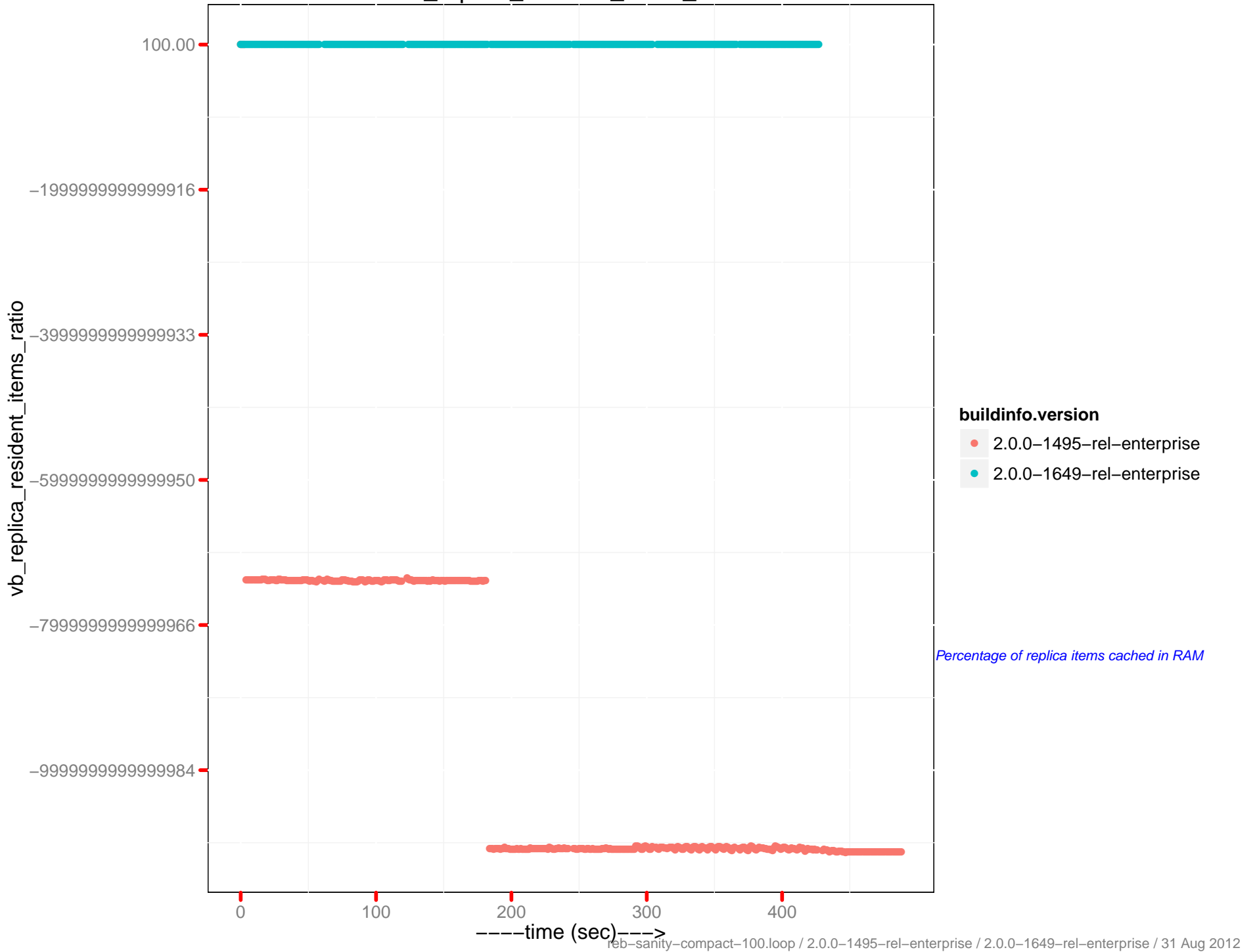
ep_tap_replica_queue_backoff/sec



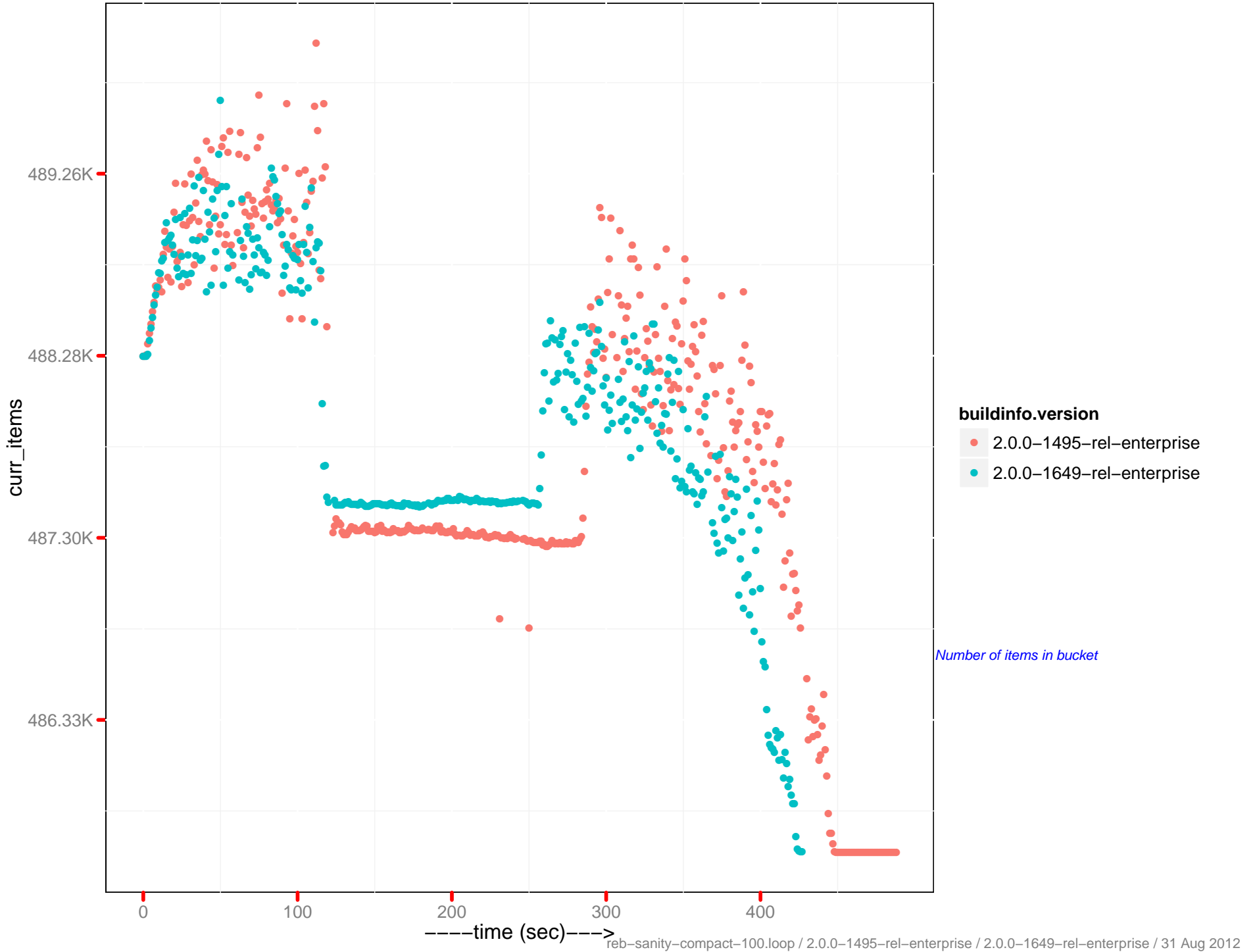
vb_active_resident_items_ratio



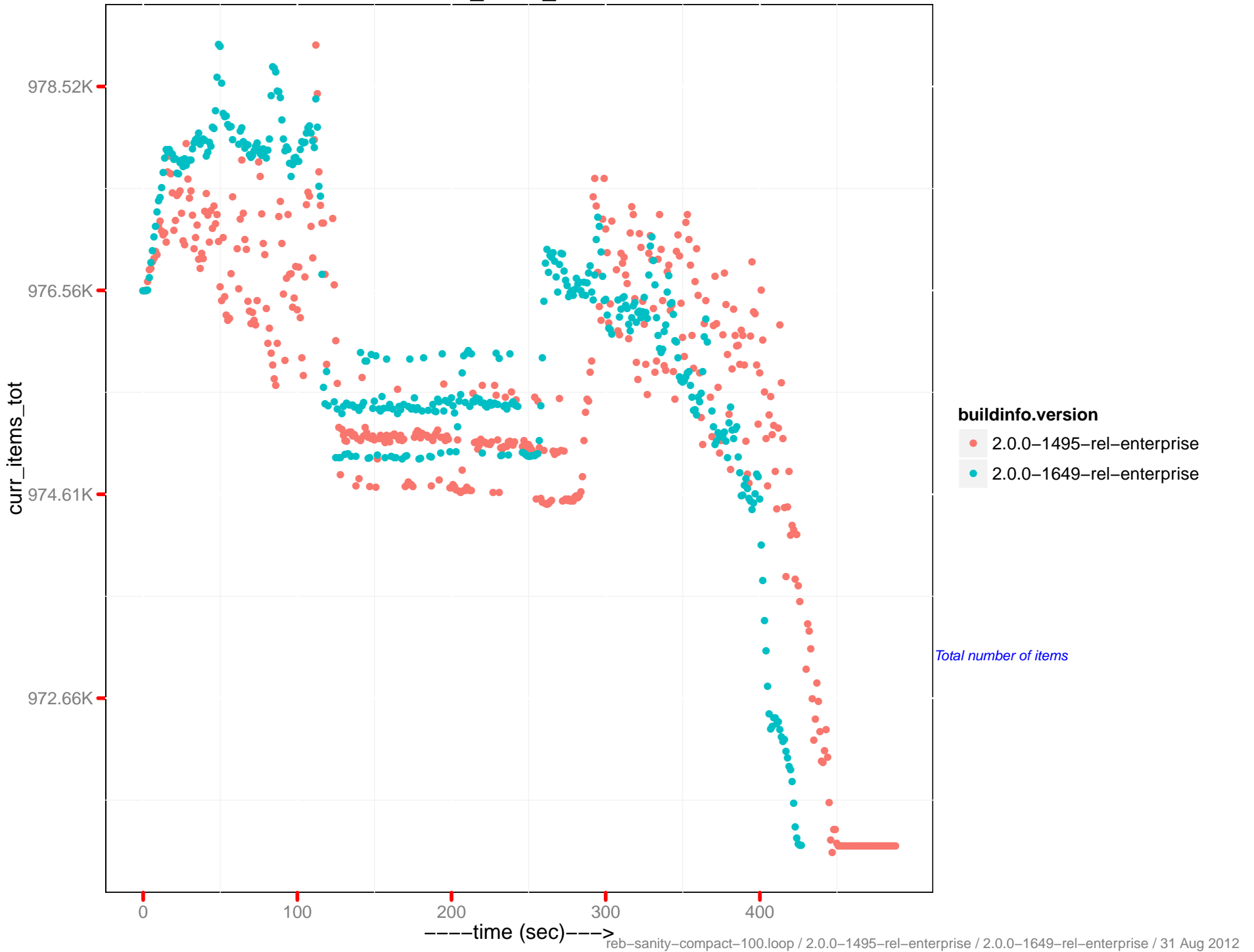
vb_replica_resident_items_ratio



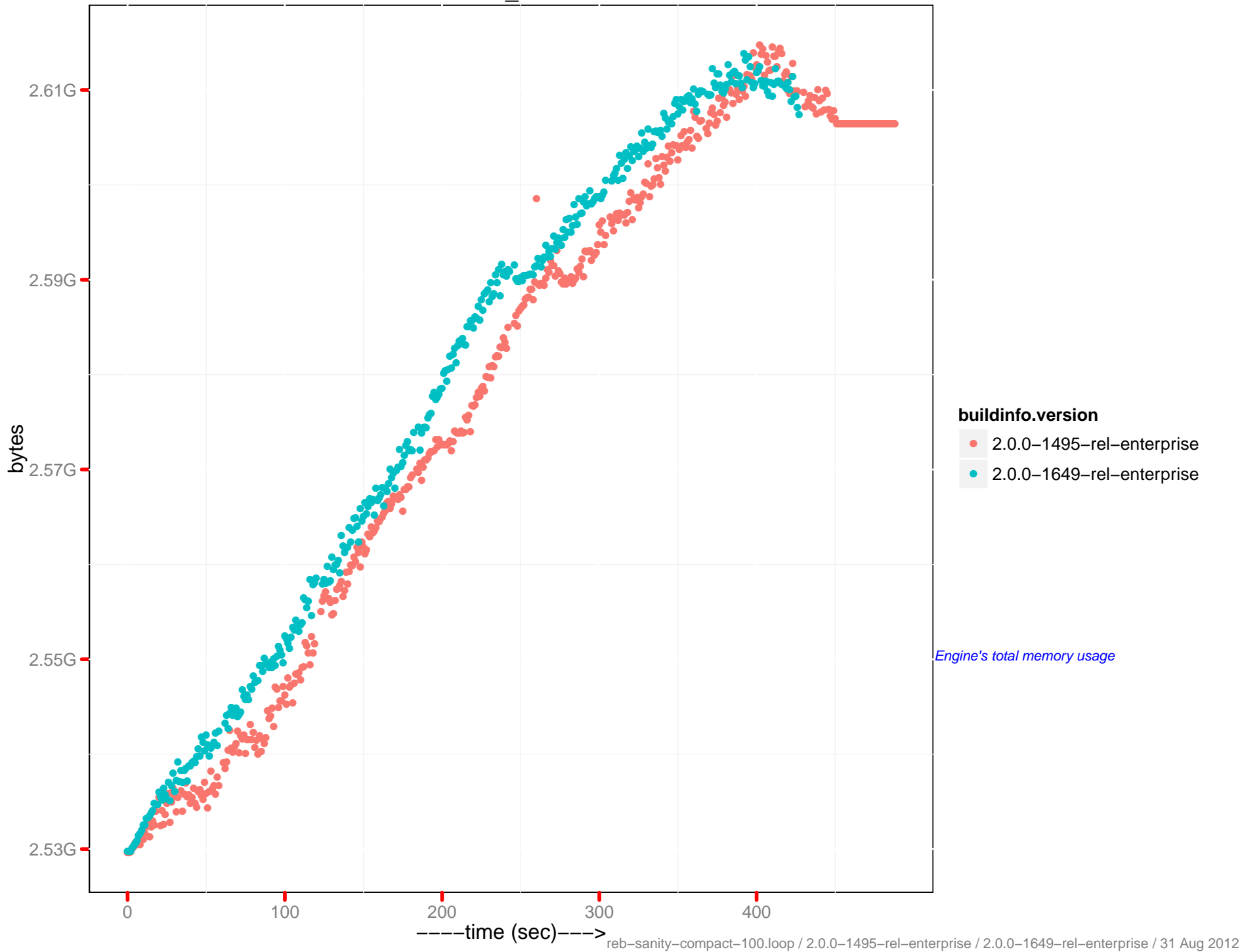
curr_items



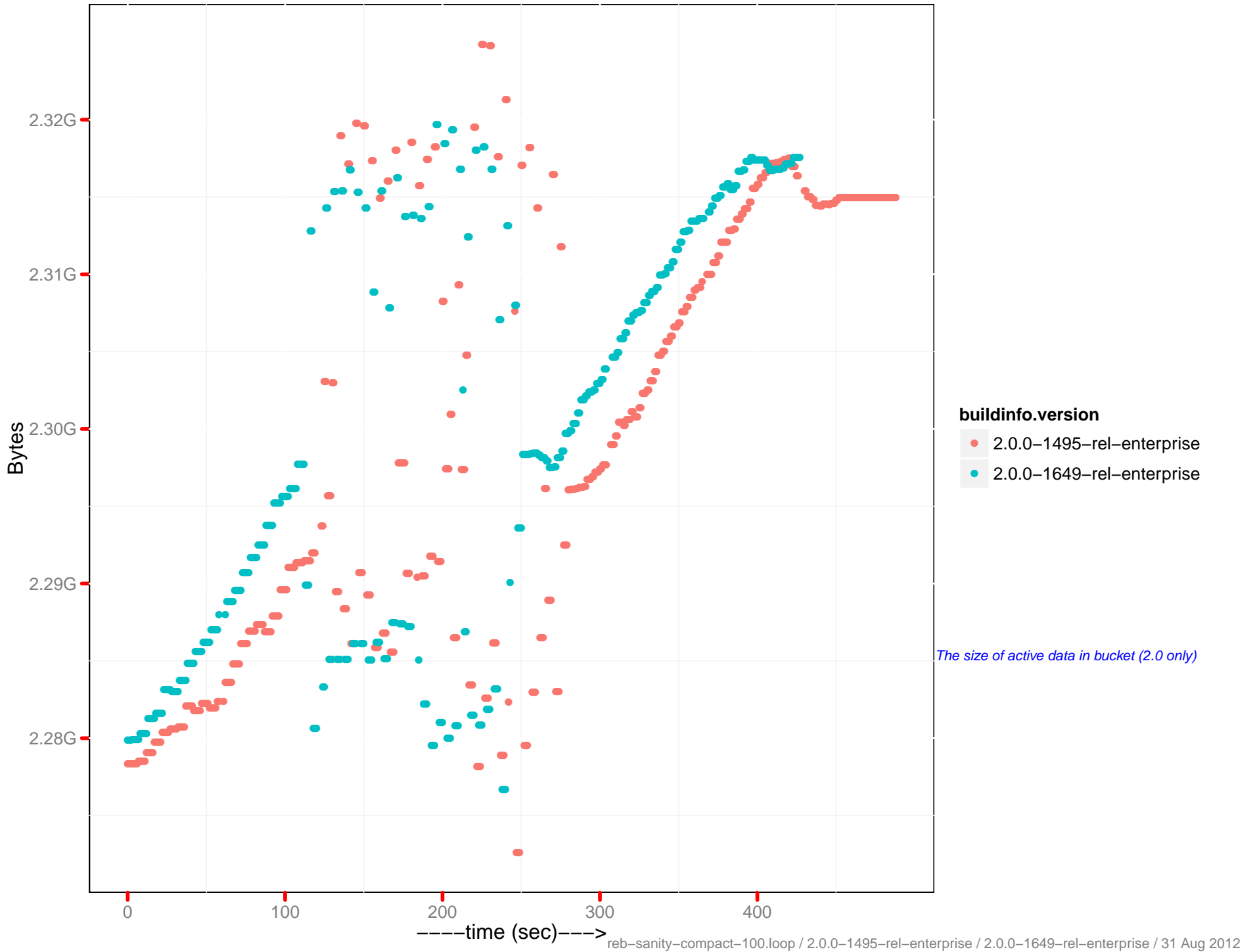
cur_items_total



mem_used



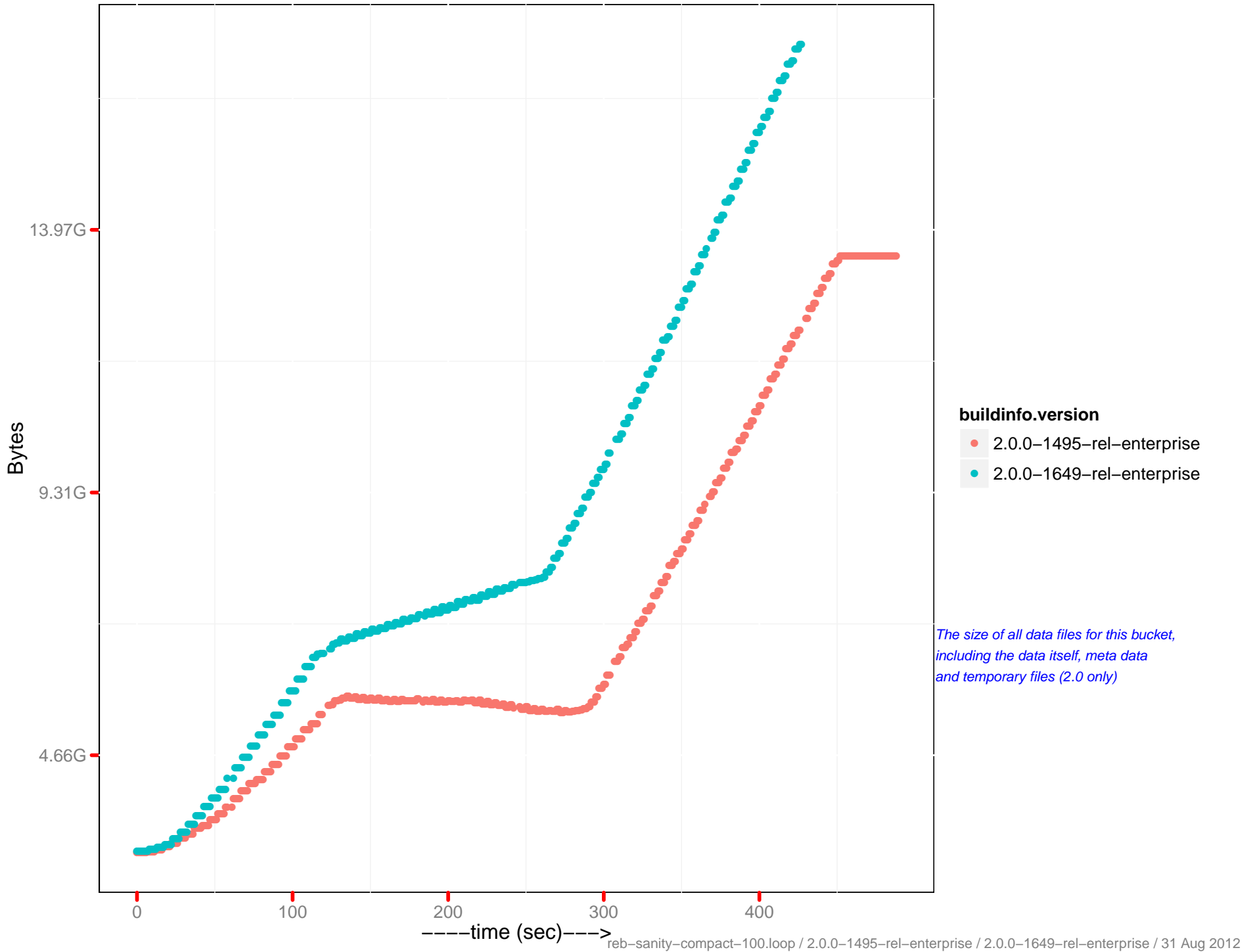
Docs data size



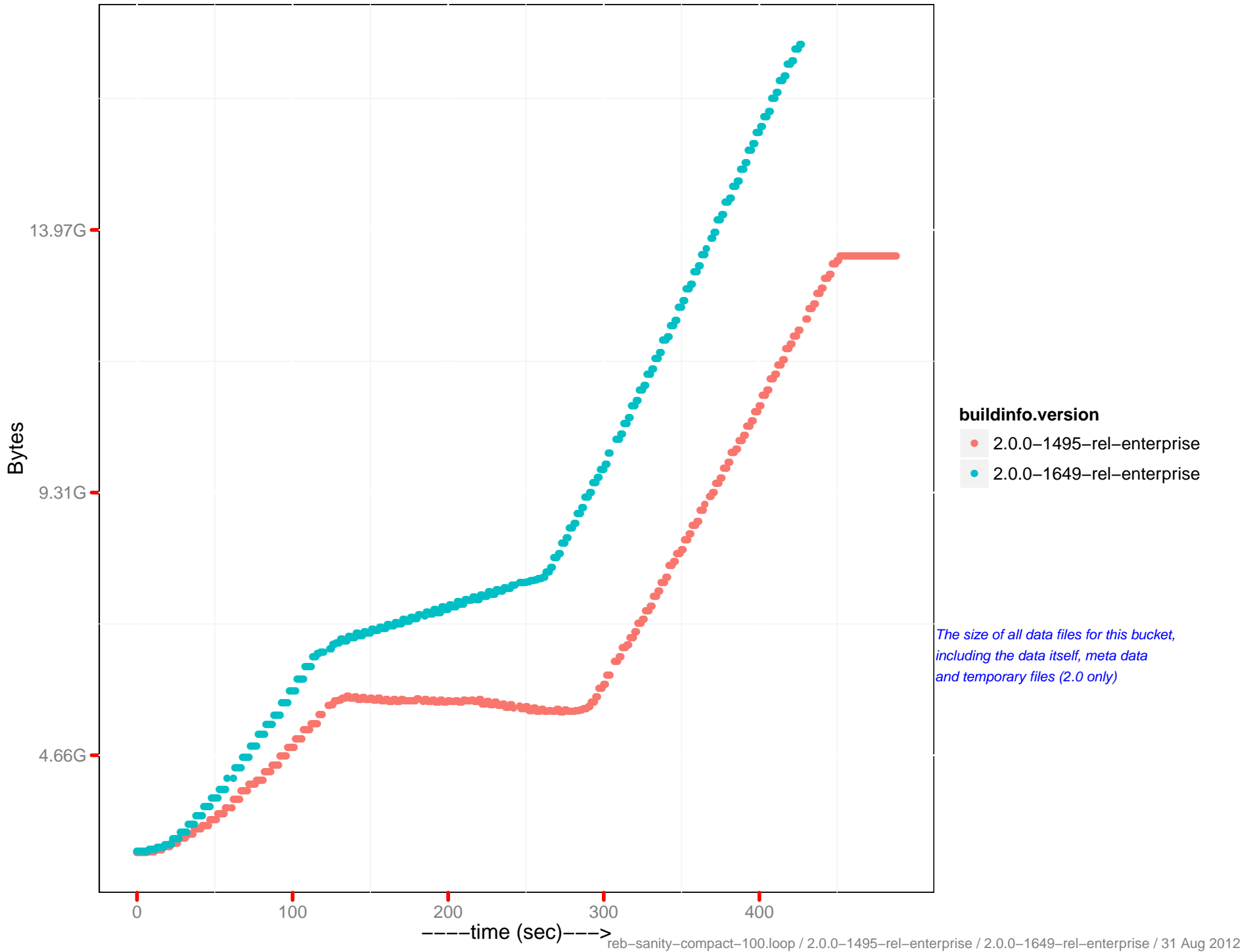
buildinfo.version
● 2.0.0-1495-rel-enterprise
● 2.0.0-1649-rel-enterprise

The size of active data in bucket (2.0 only)

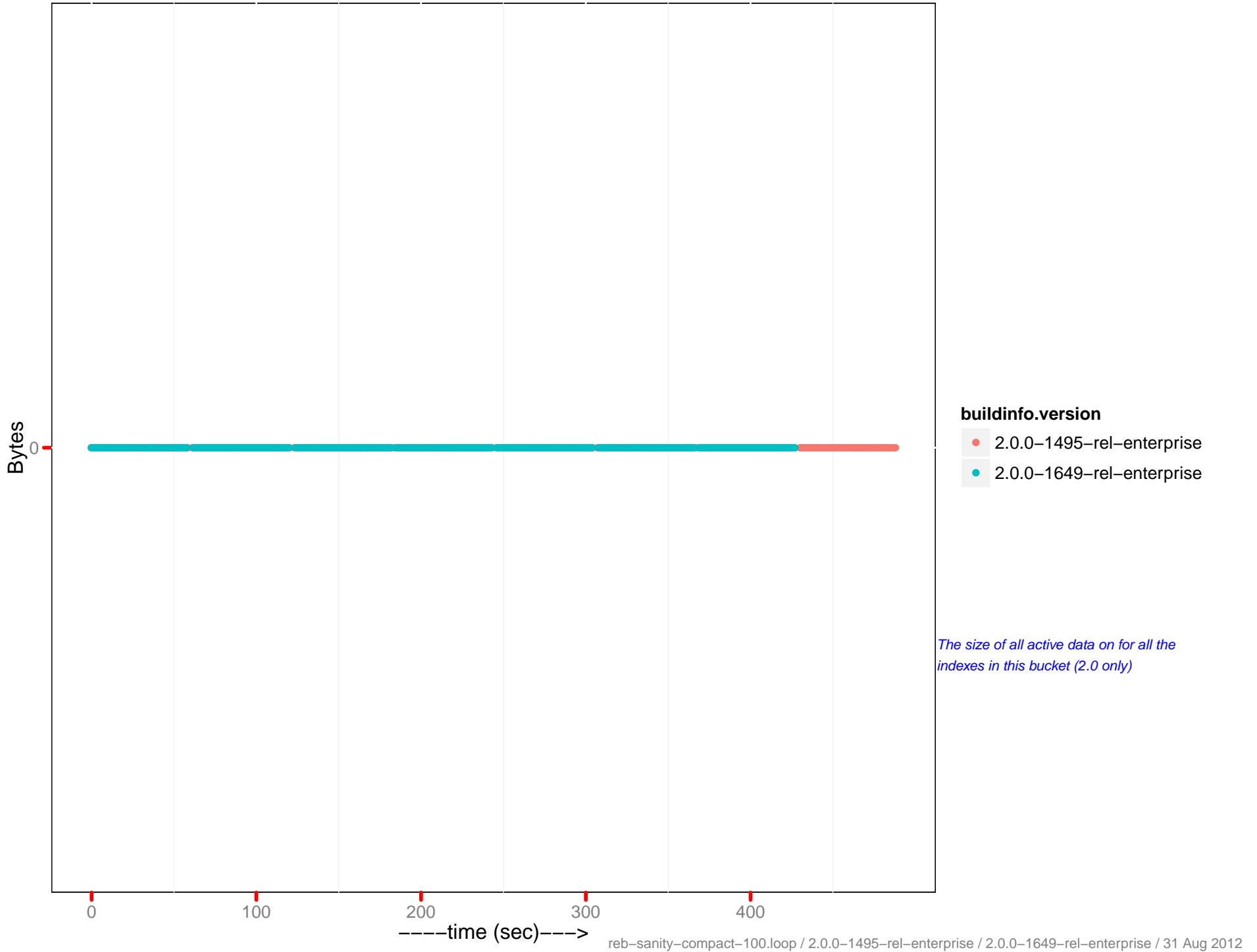
Docs disk size



Docs actual disk size

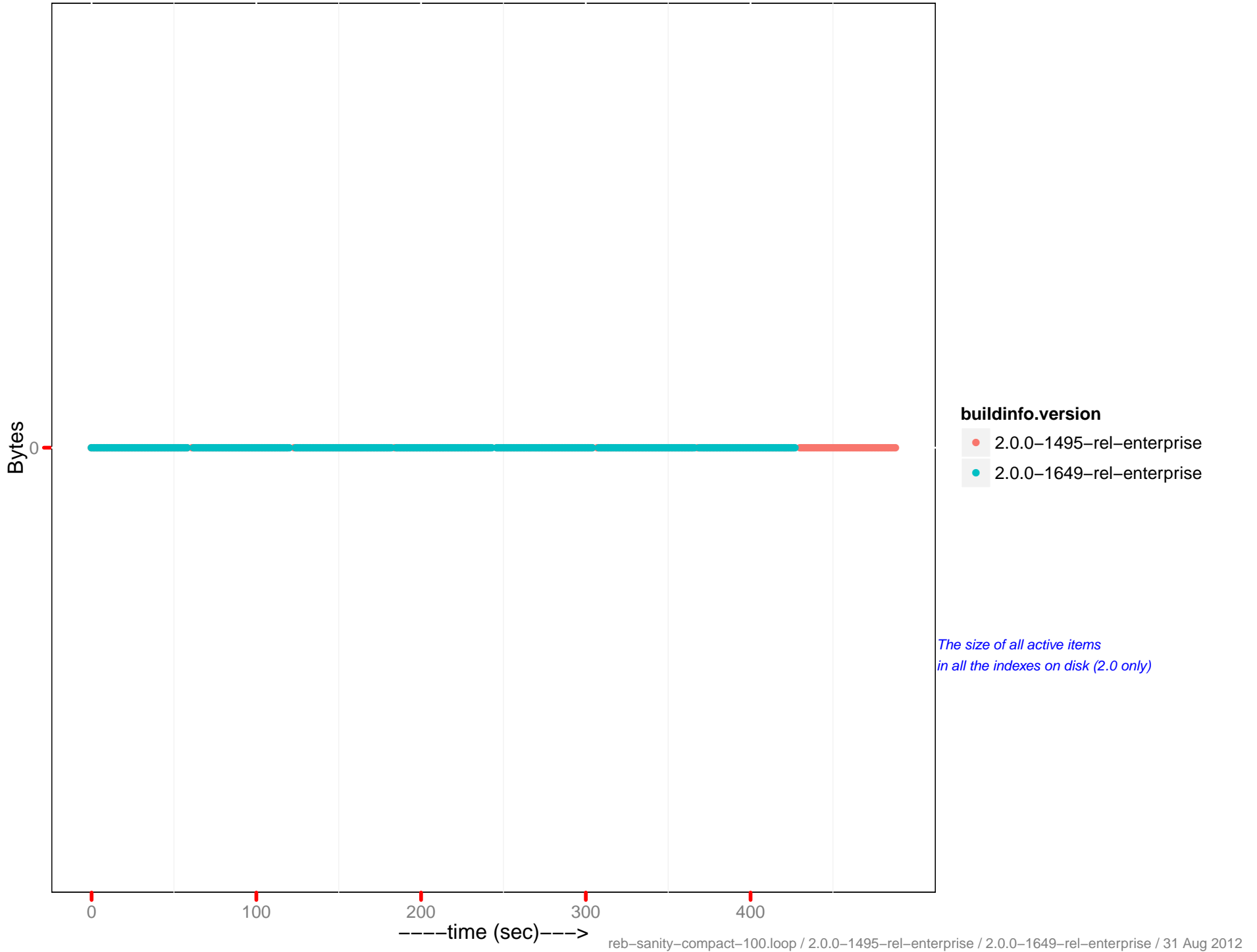


Views data size

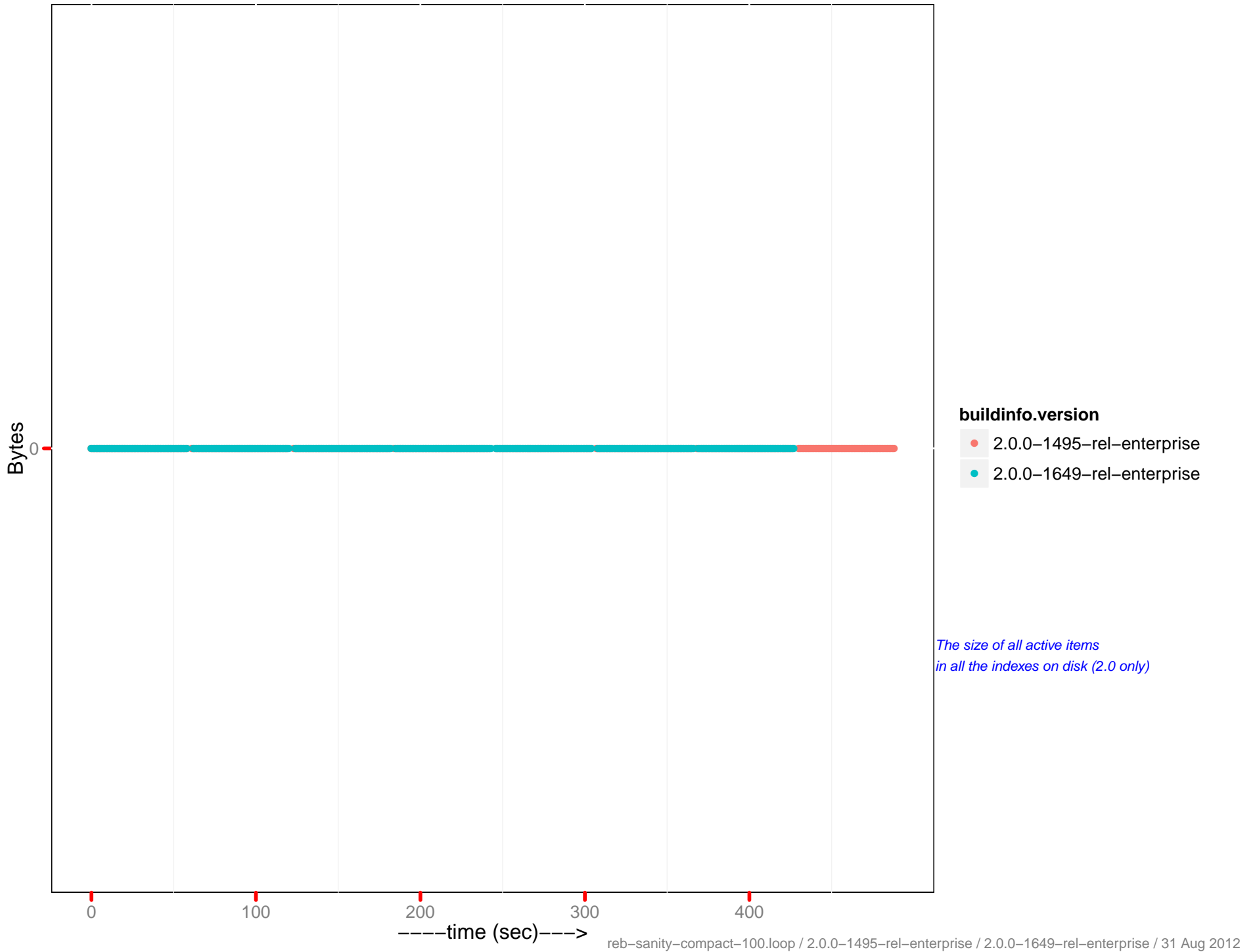


The size of all active data on for all the indexes in this bucket (2.0 only)

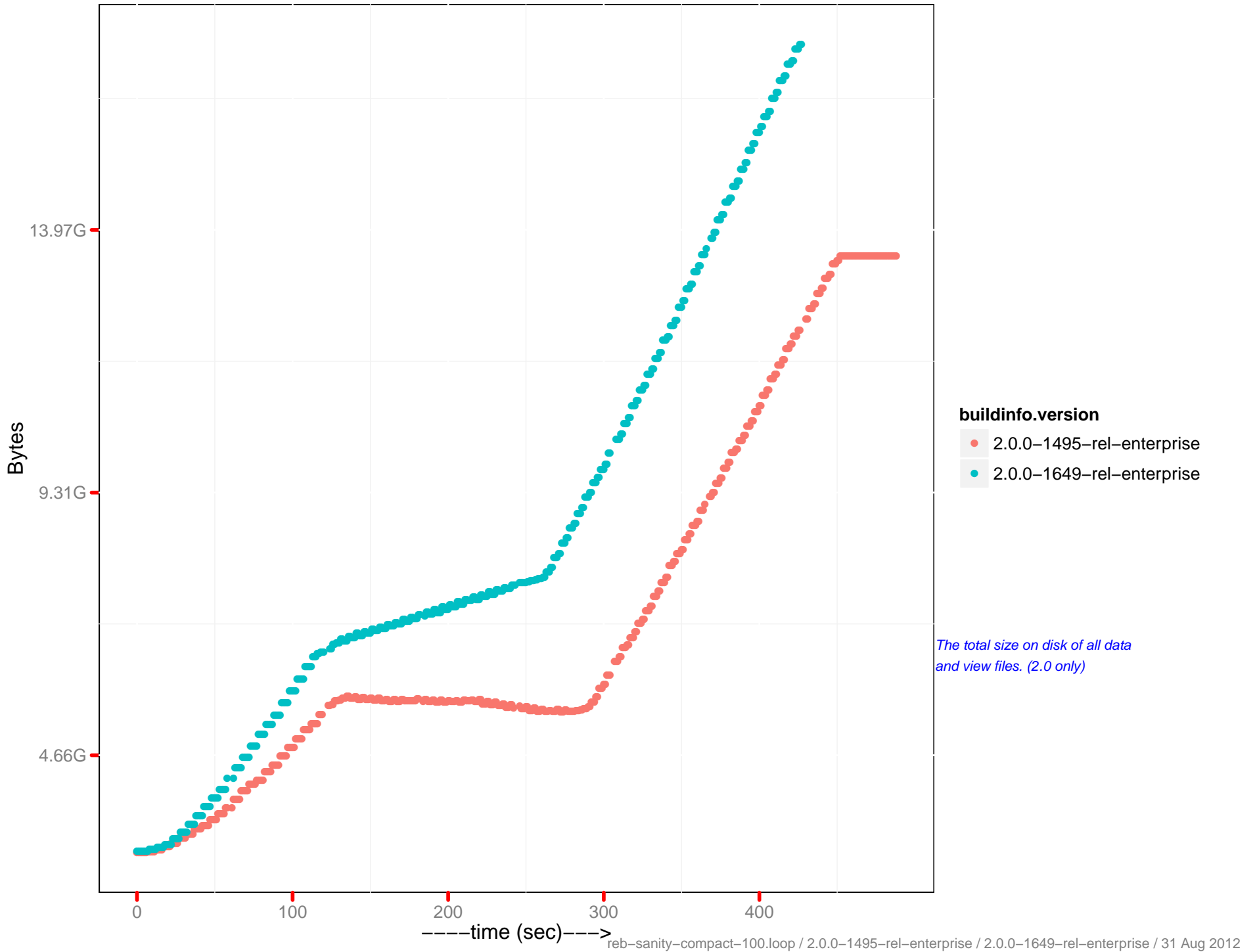
Views disk size



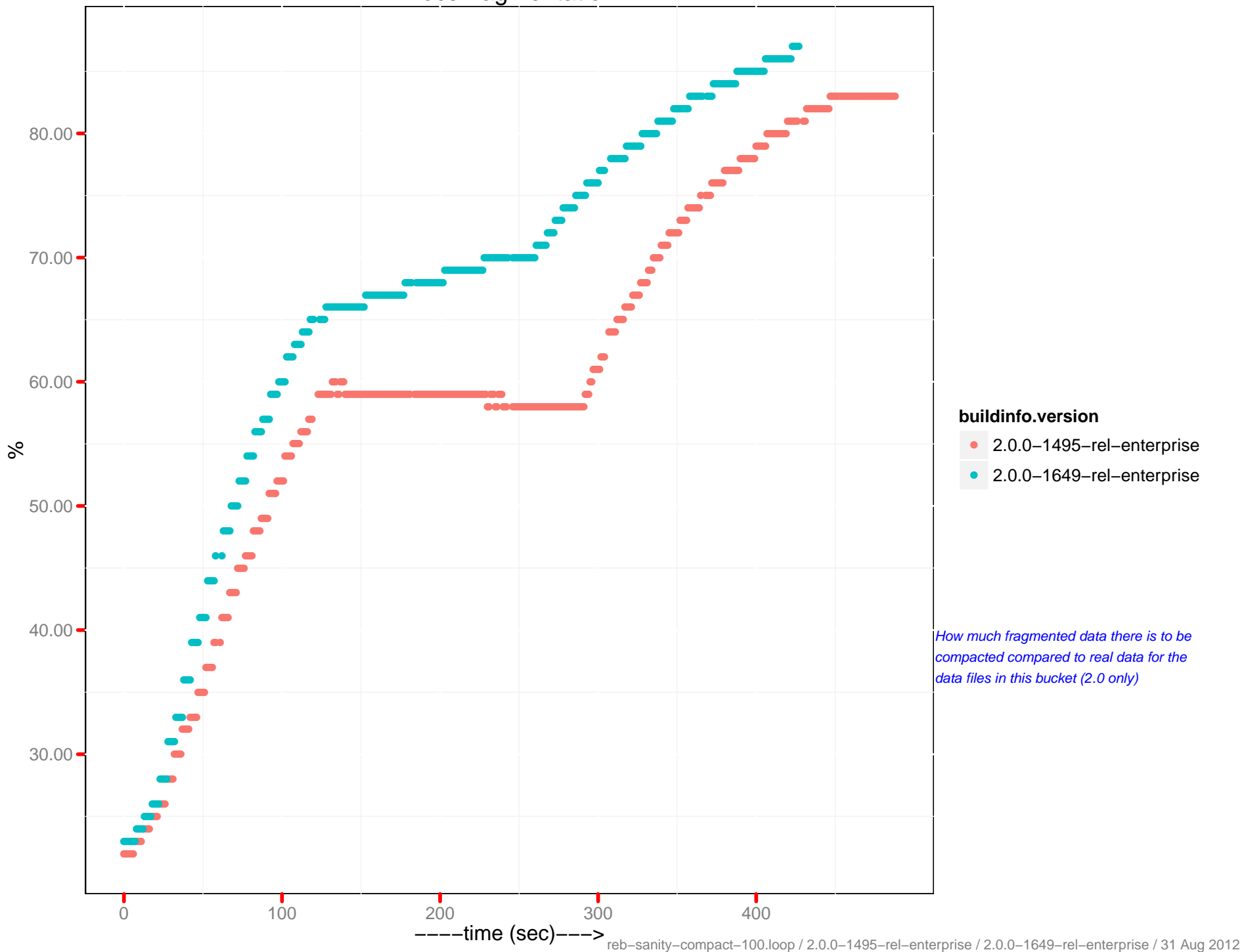
Views actual disk size



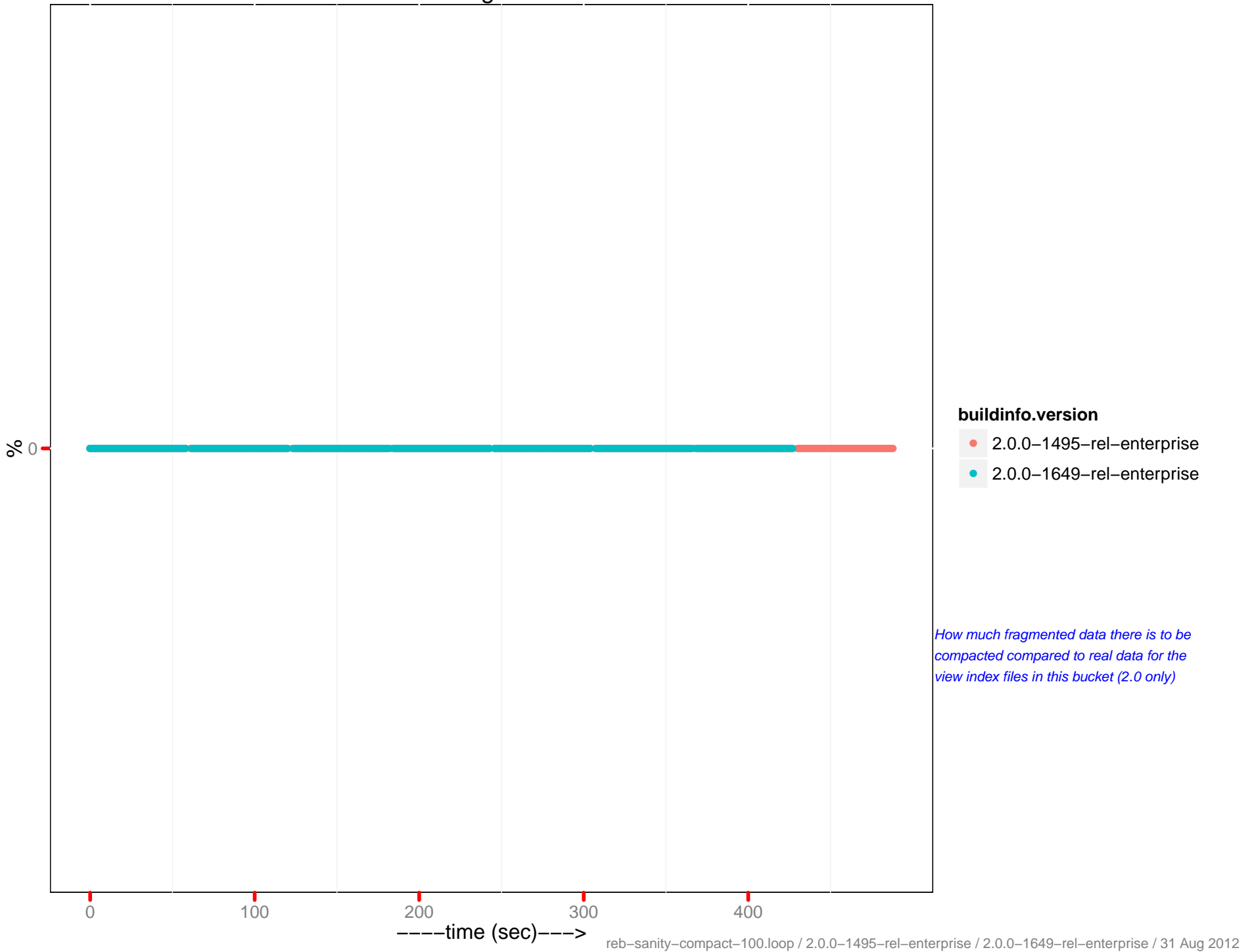
Total disk size



Docs fragmentation



Views fragmentation

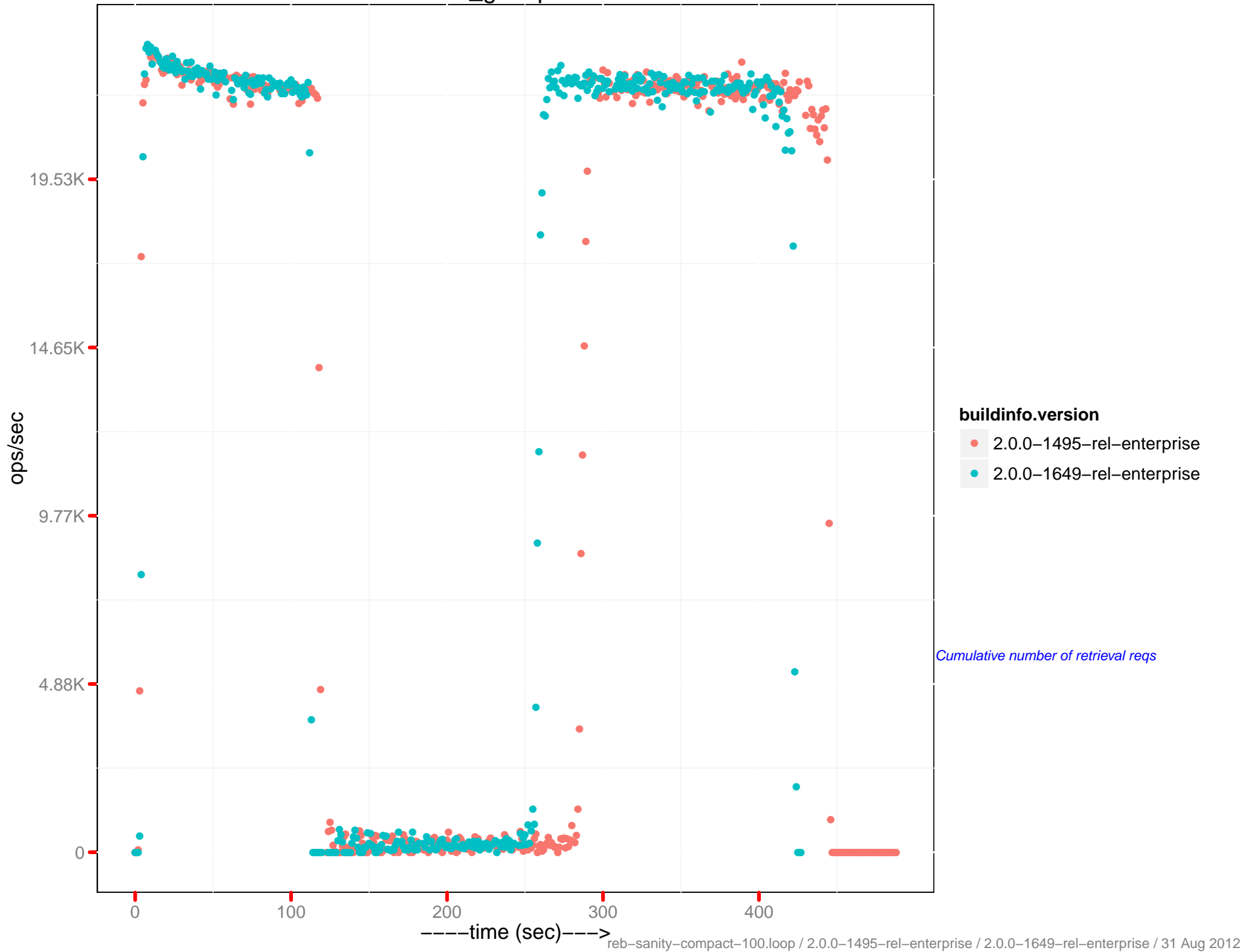


buildinfo.version

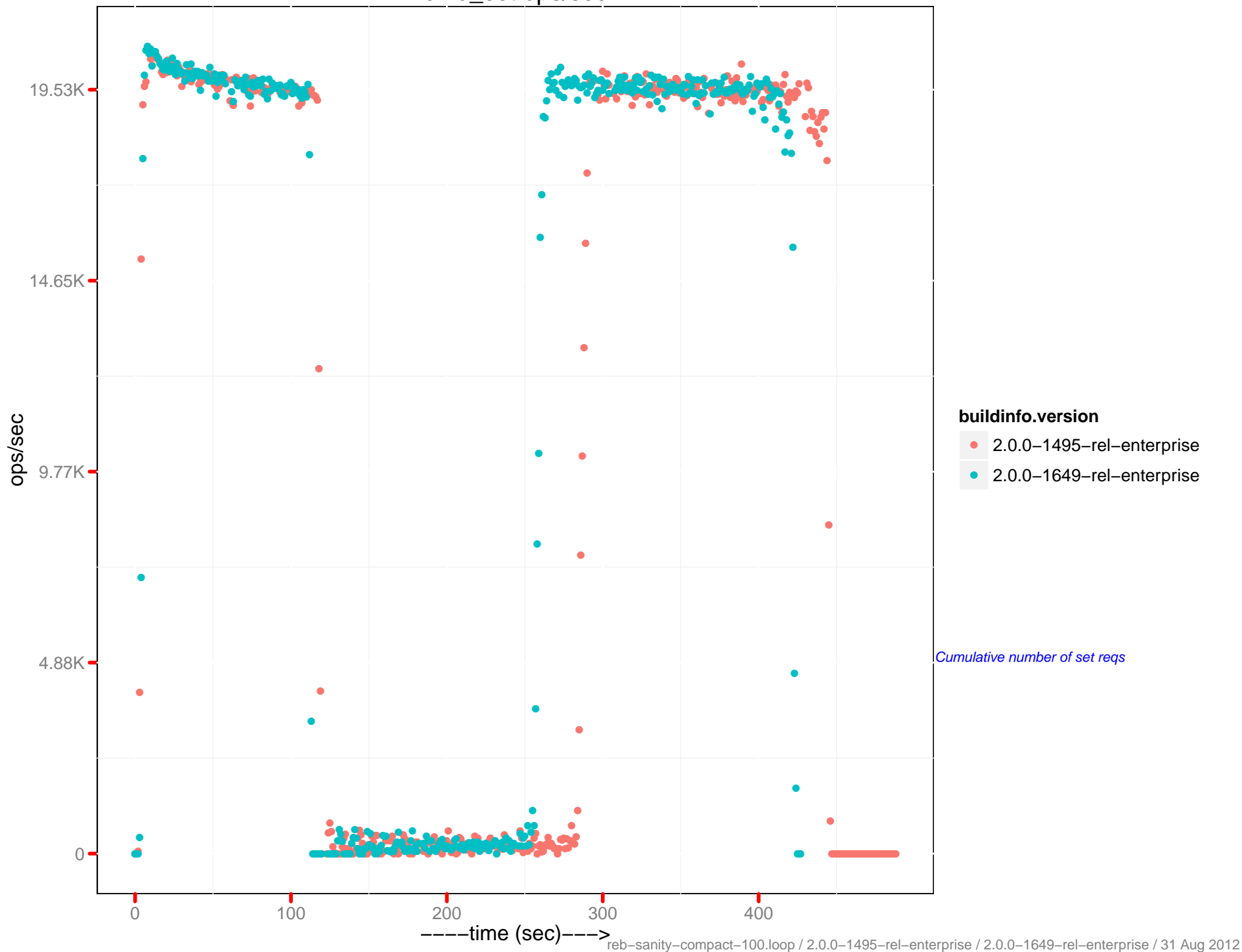
- 2.0.0-1495-rel-enterprise
- 2.0.0-1649-rel-enterprise

How much fragmented data there is to be compacted compared to real data for the view index files in this bucket (2.0 only)

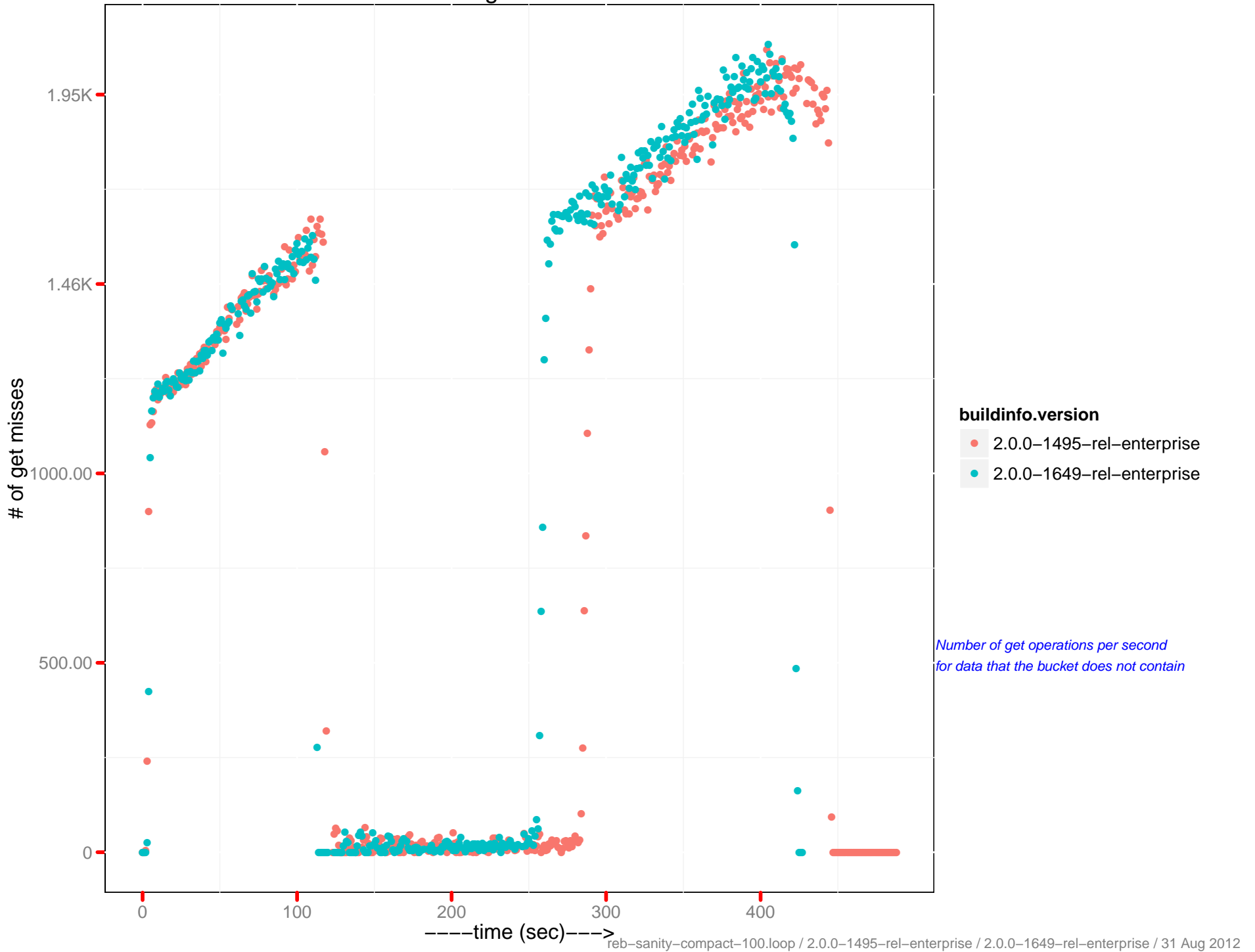
cmd_get ops/sec



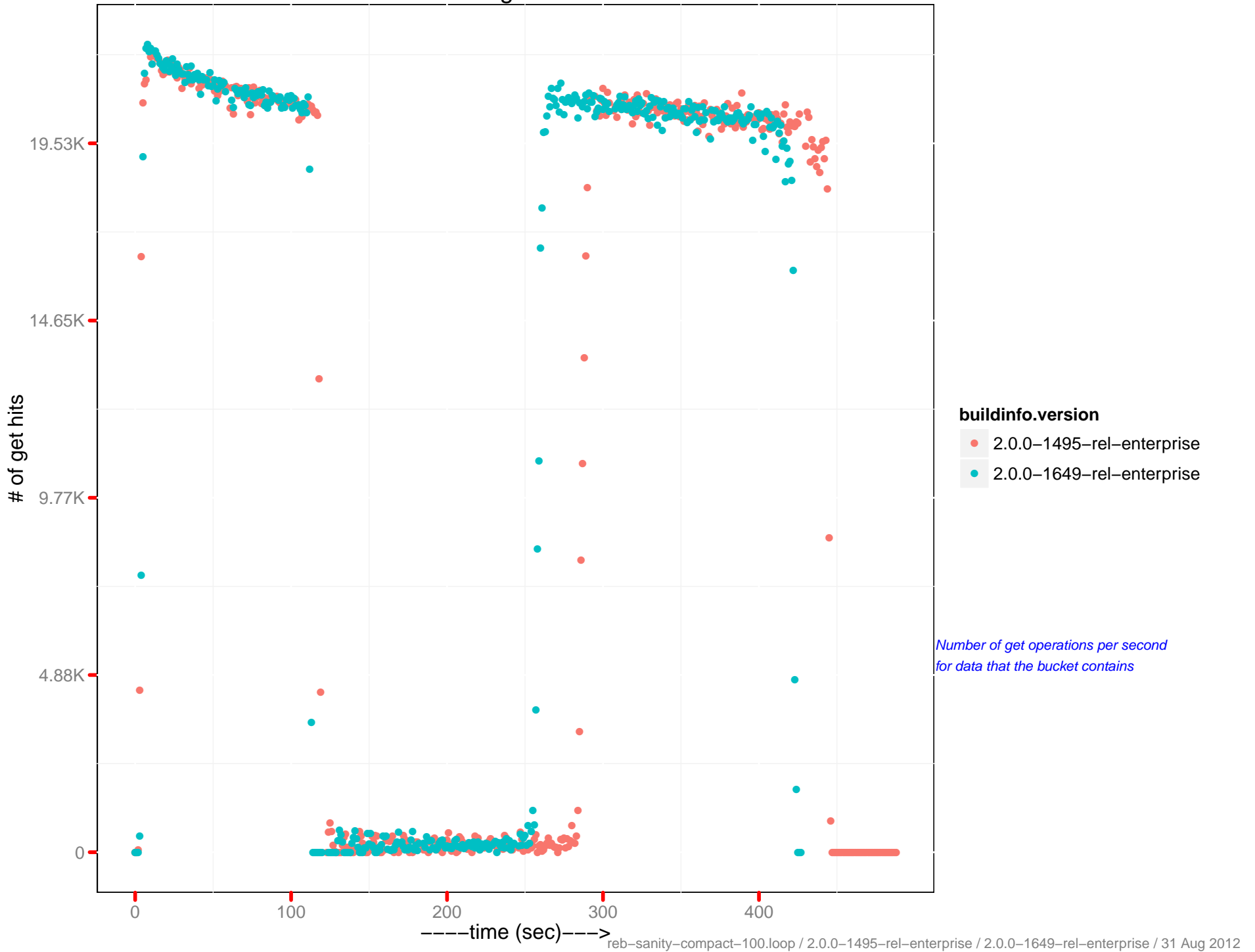
cmd_set ops/sec



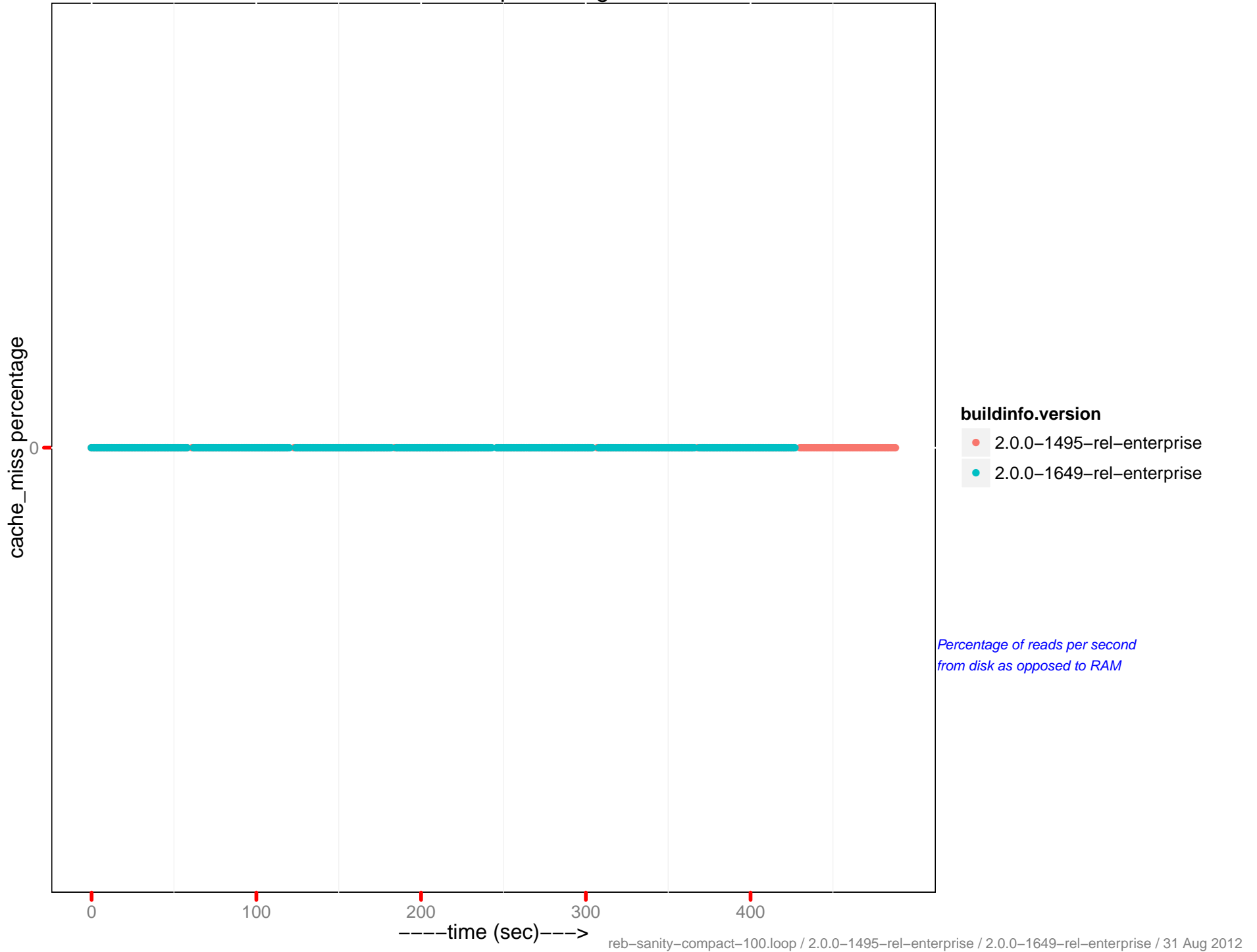
of get misses



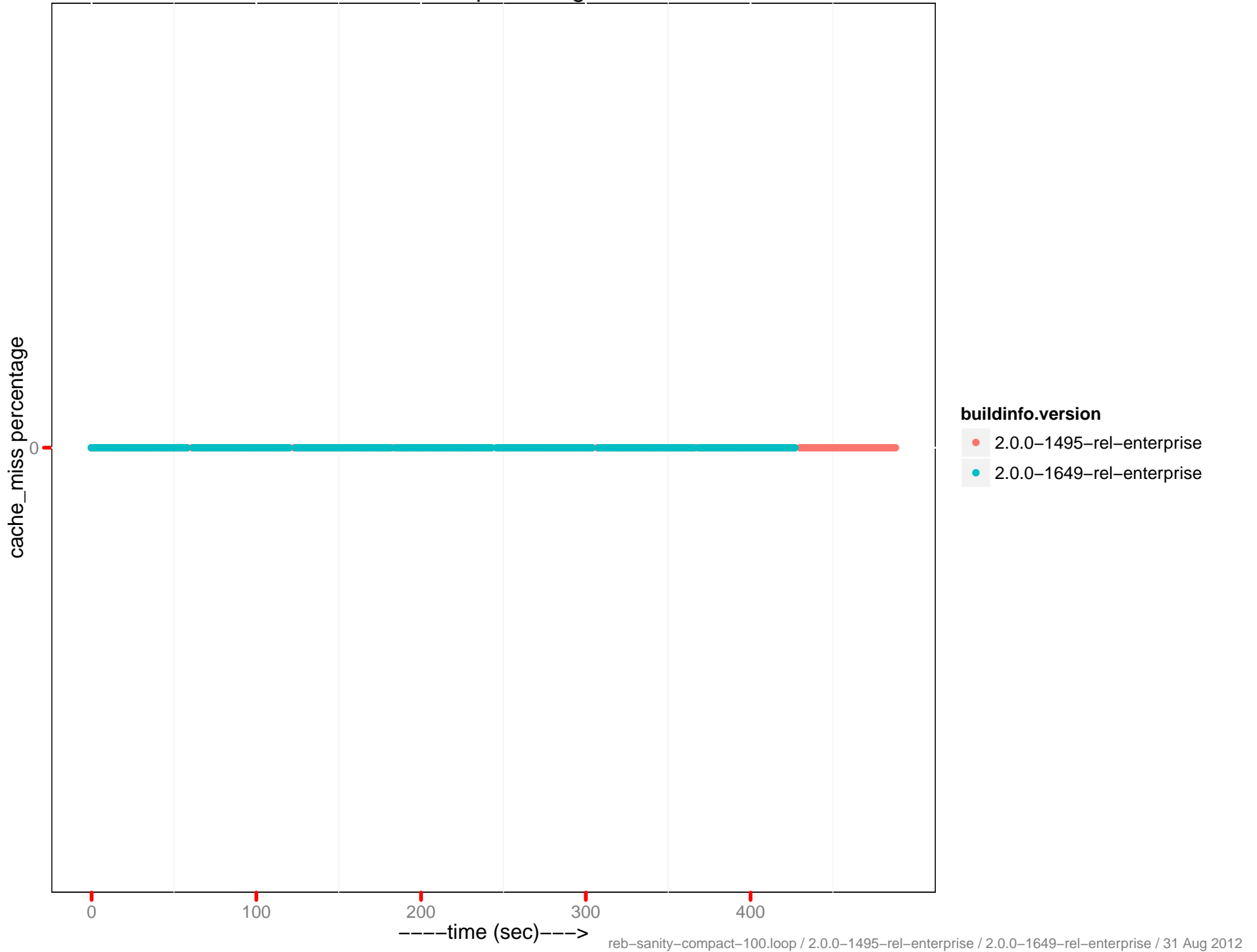
of get hits



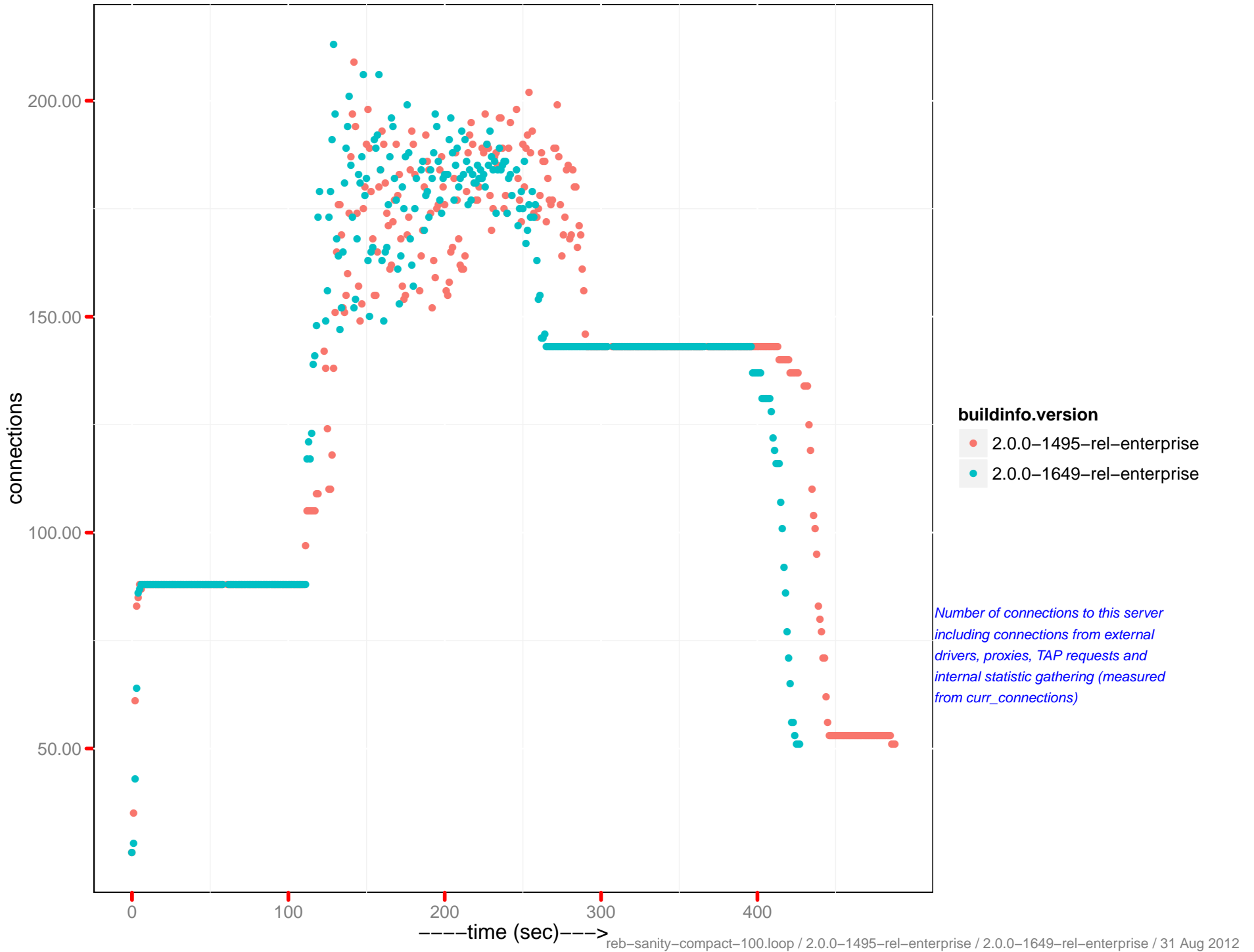
cache_miss percentage



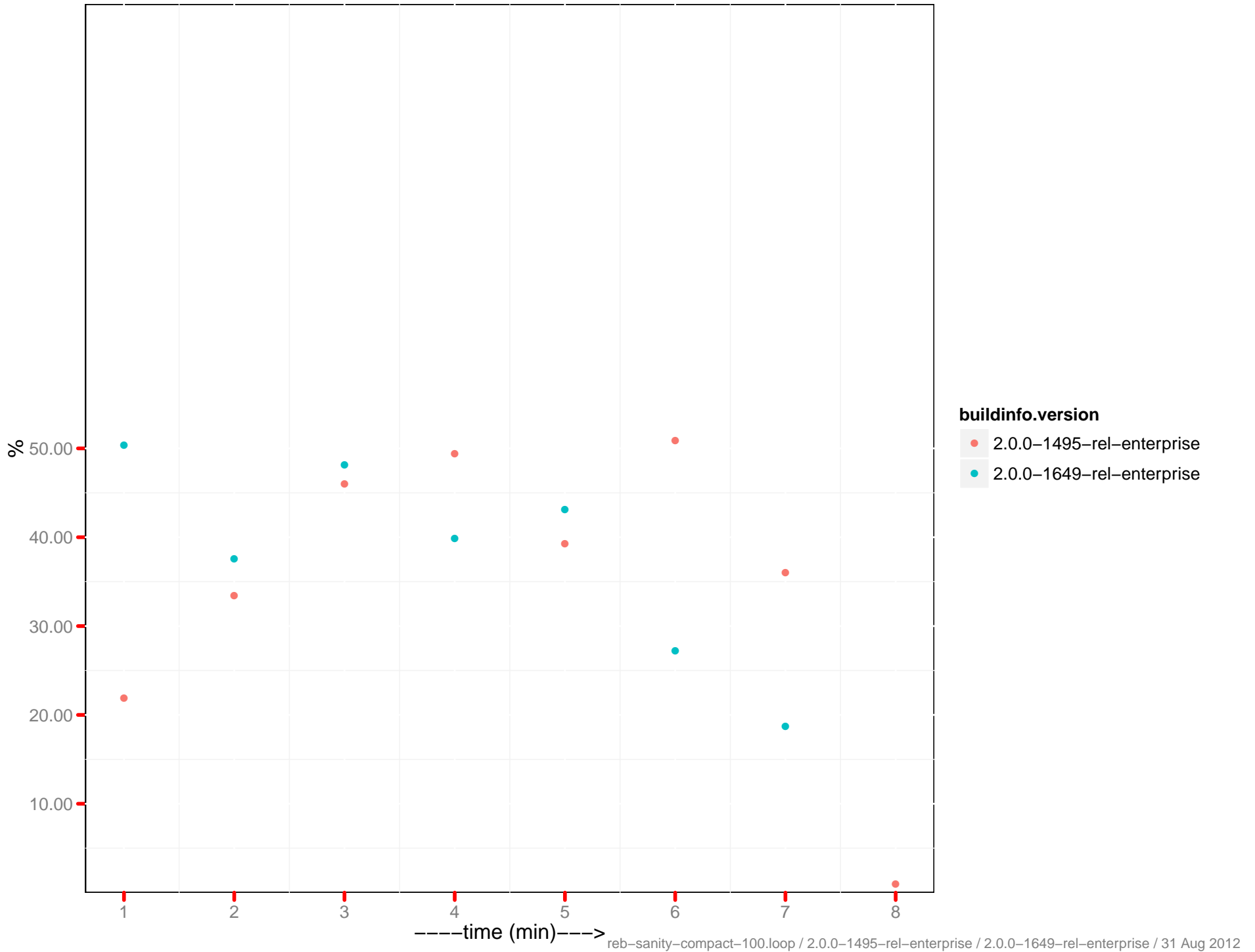
cache_miss percentage 0-5



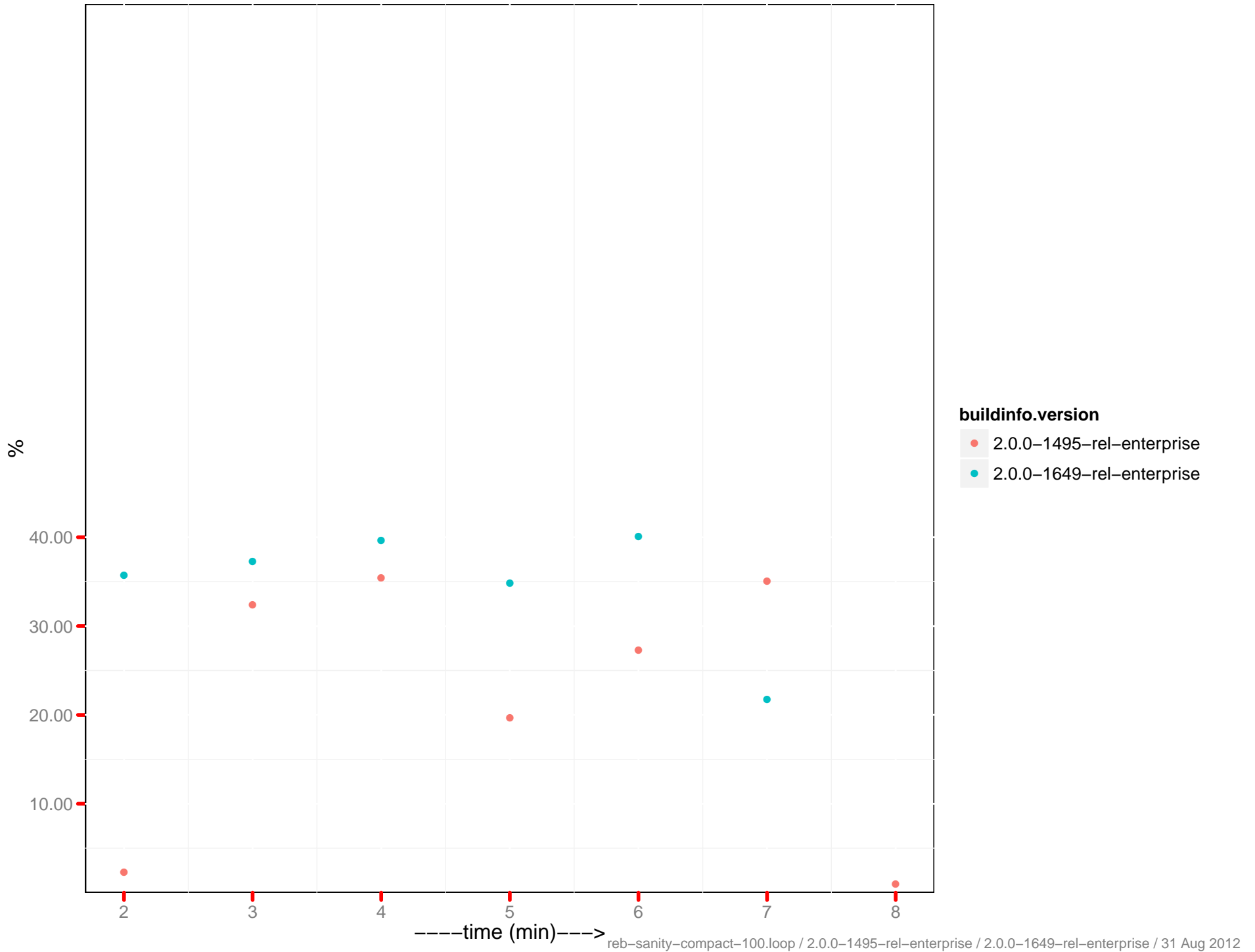
Number of connections



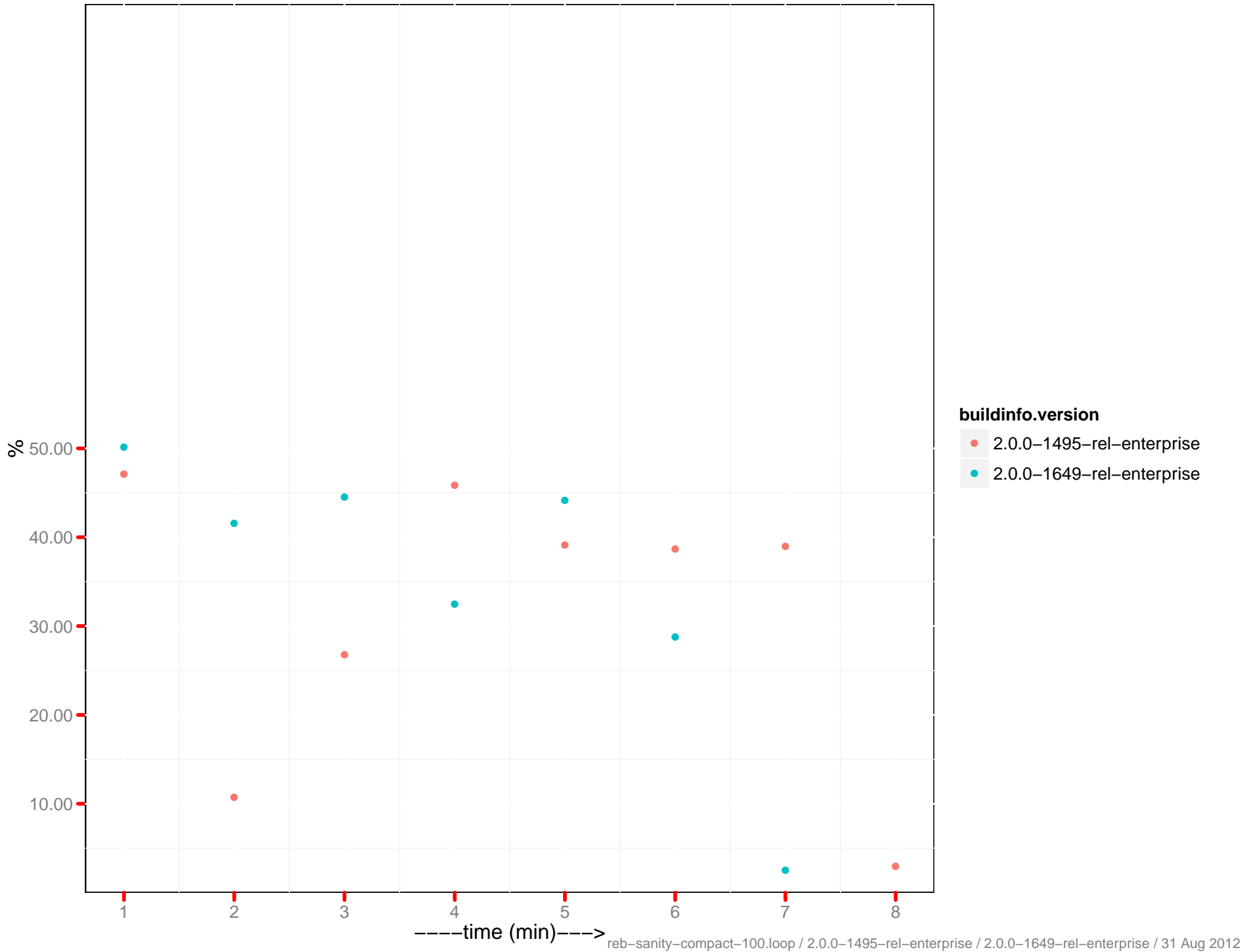
CPU utilization – 10.2.1.65:8091



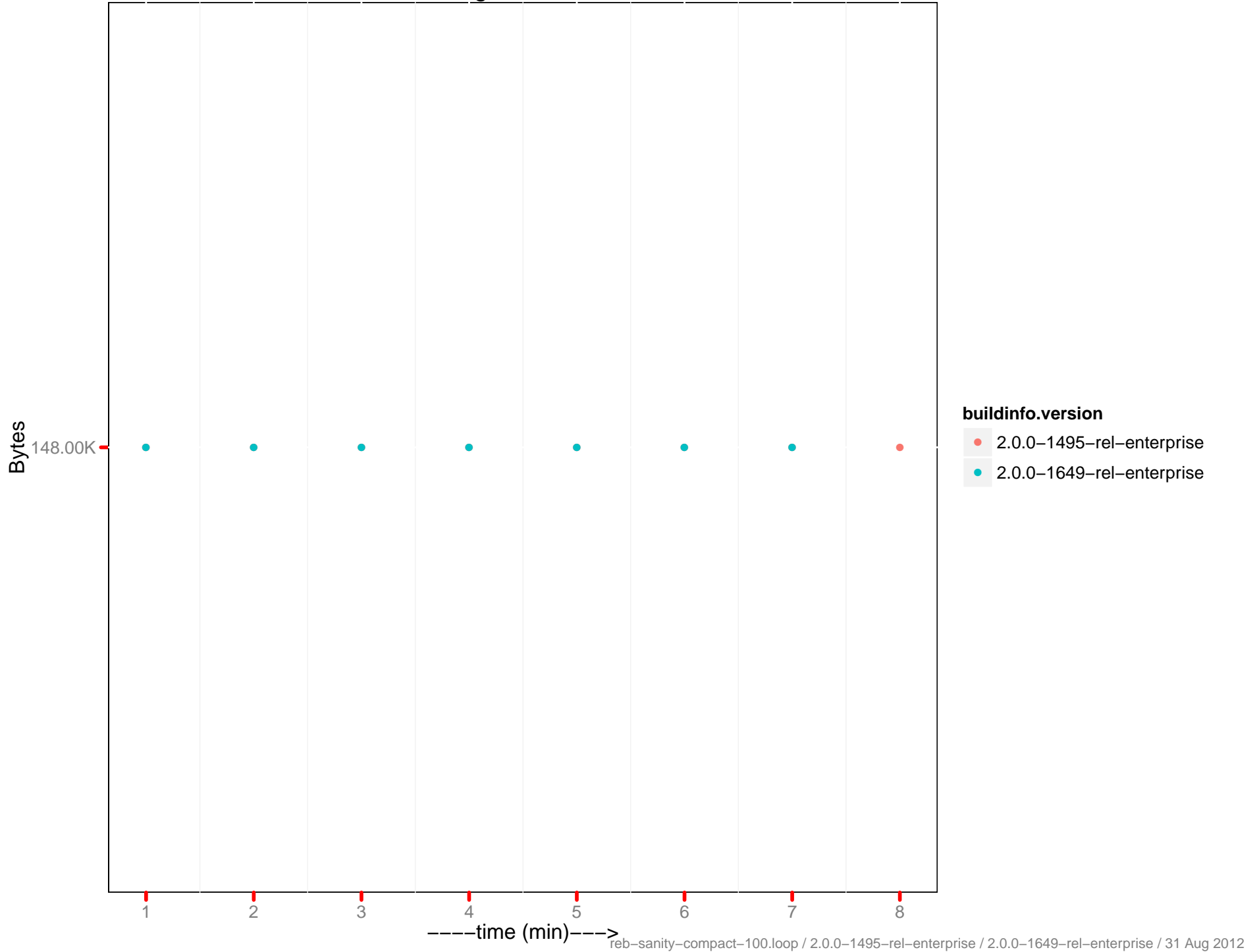
CPU utilization – 10.2.1.66:8091



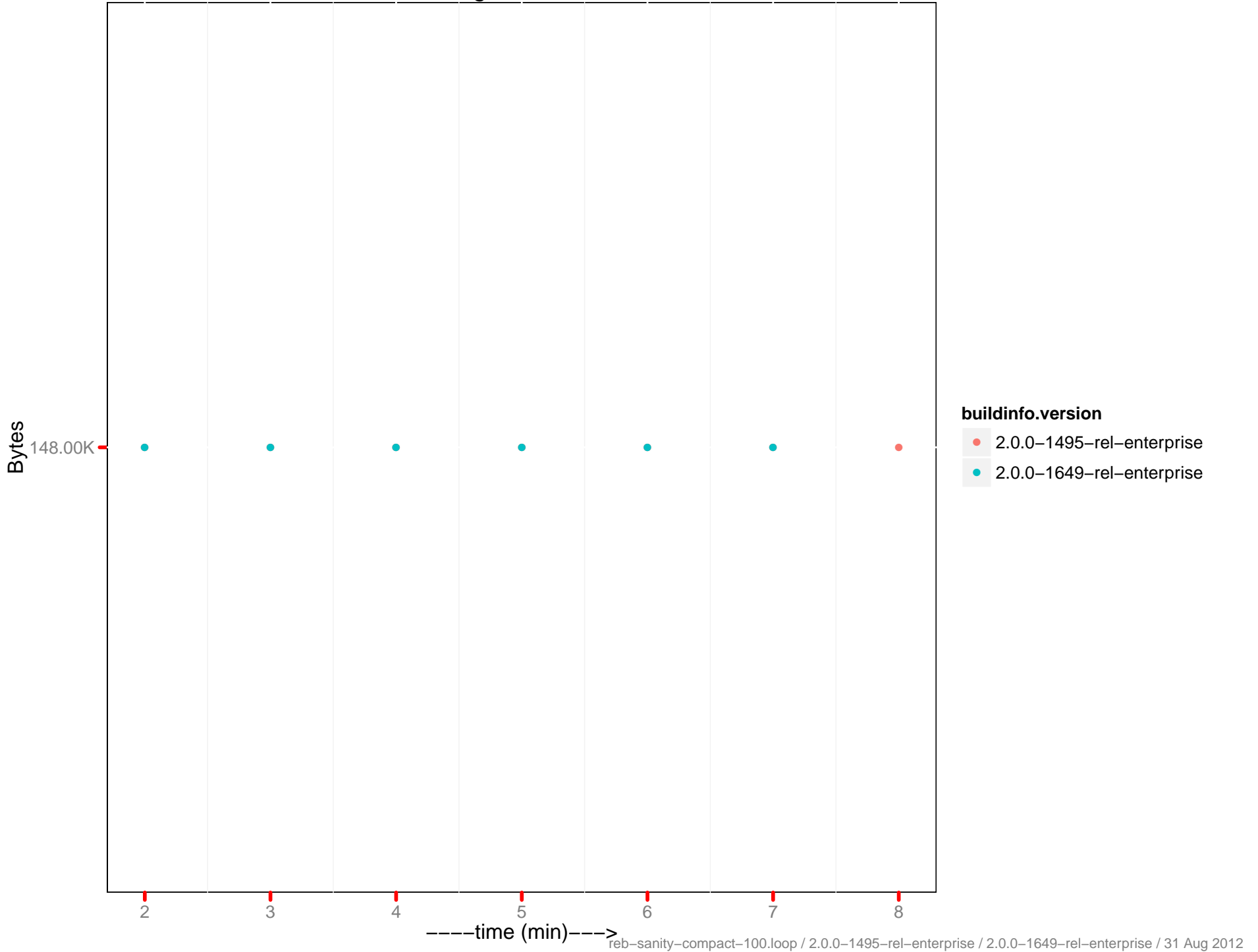
CPU utilization – 10.2.1.67:8091



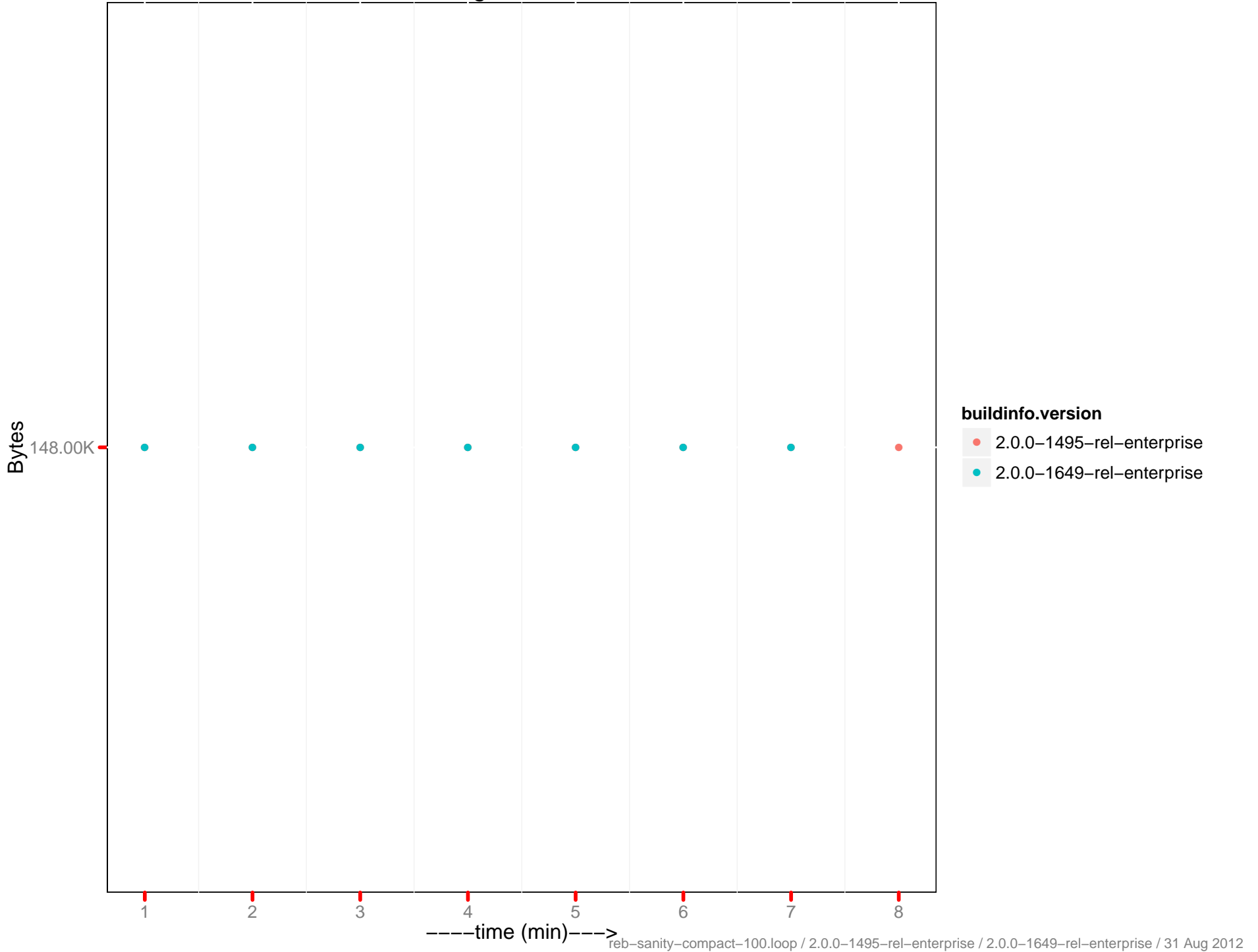
SWAP Usage – 10.2.1.65:8091



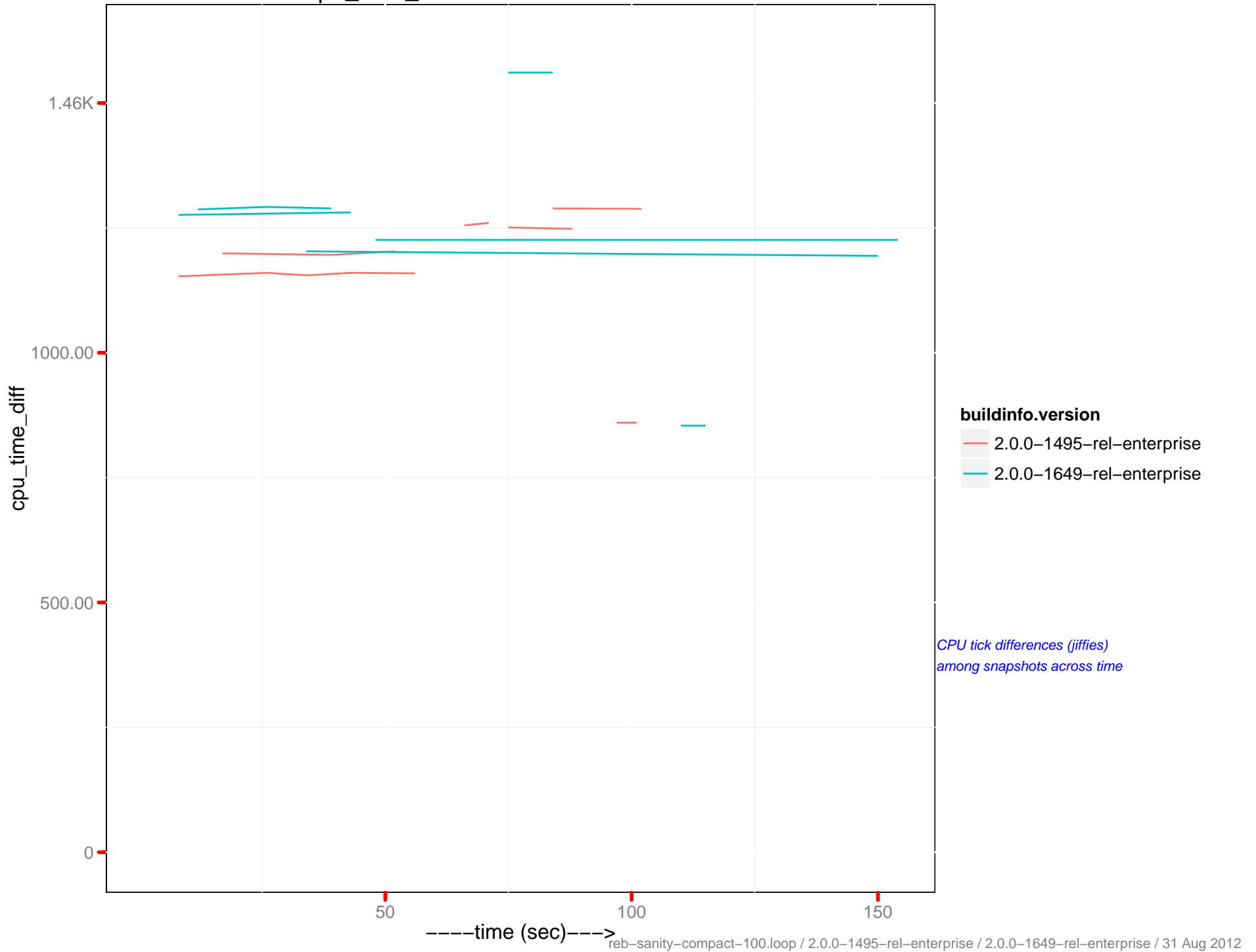
SWAP Usage - 10.2.1.66:8091



SWAP Usage - 10.2.1.67:8091



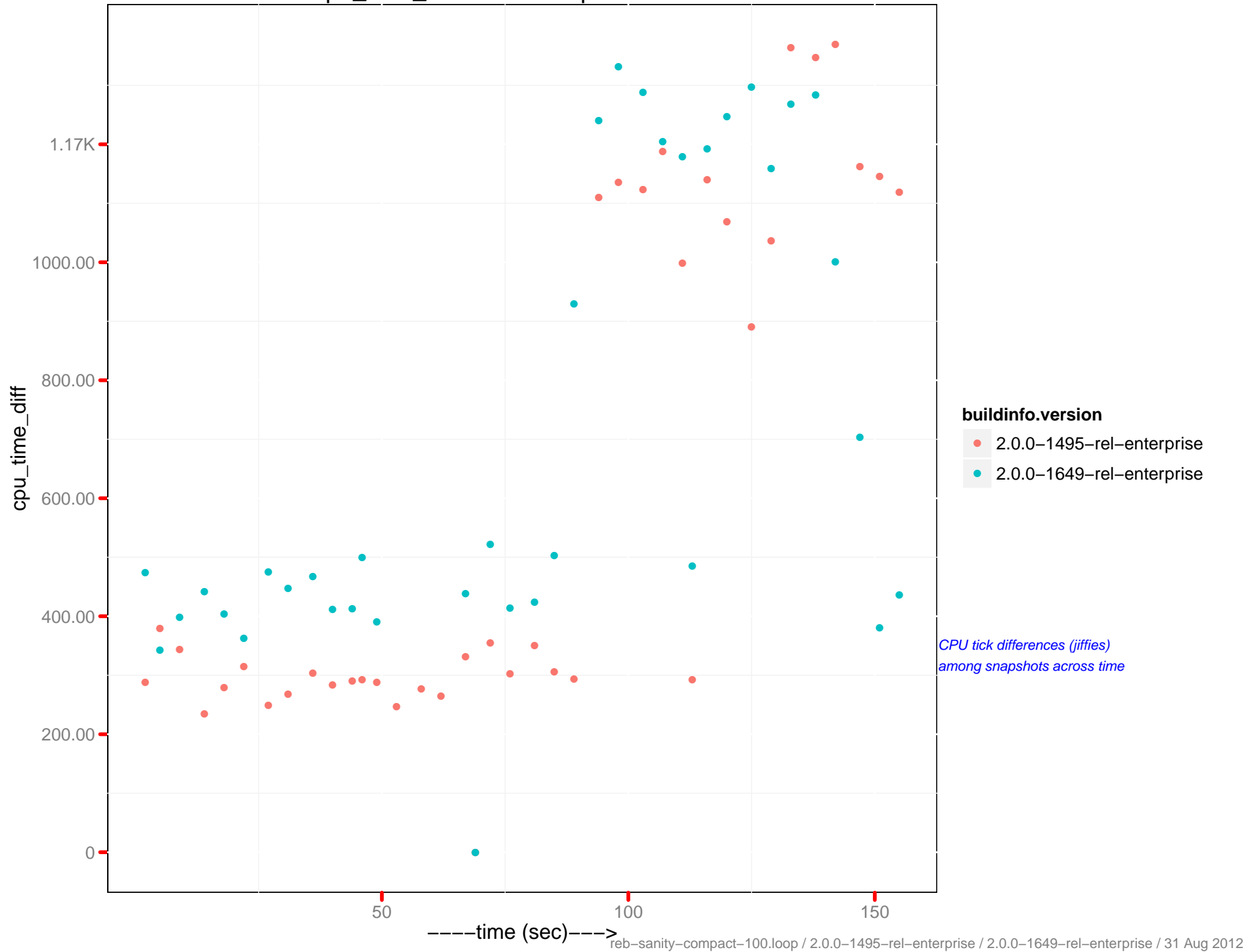
cpu_time_diff: memcached – 10.2.1.65



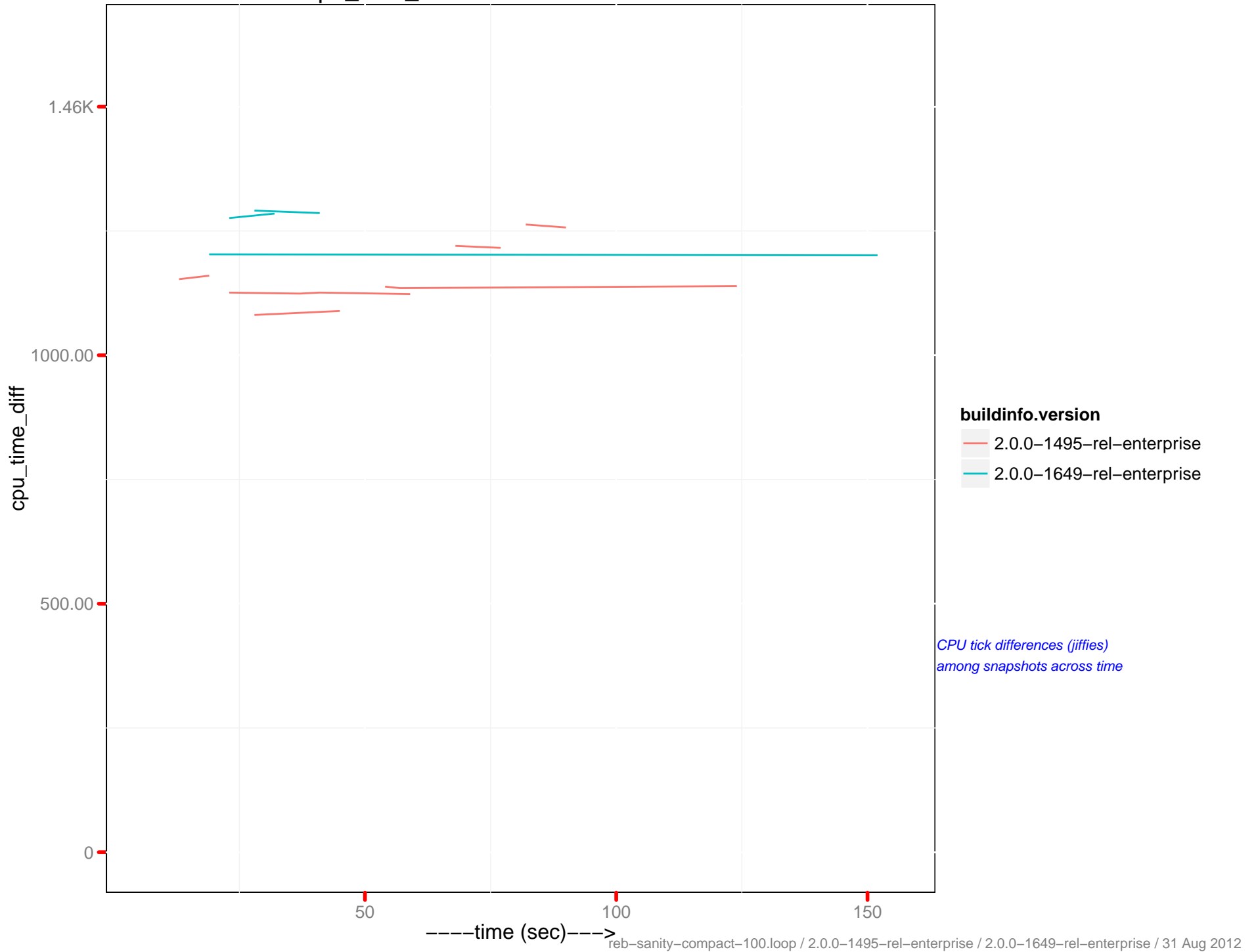
buildinfo.version
2.0.0-1495-rel-enterprise
2.0.0-1649-rel-enterprise

*CPU tick differences (jiffies)
among snapshots across time*

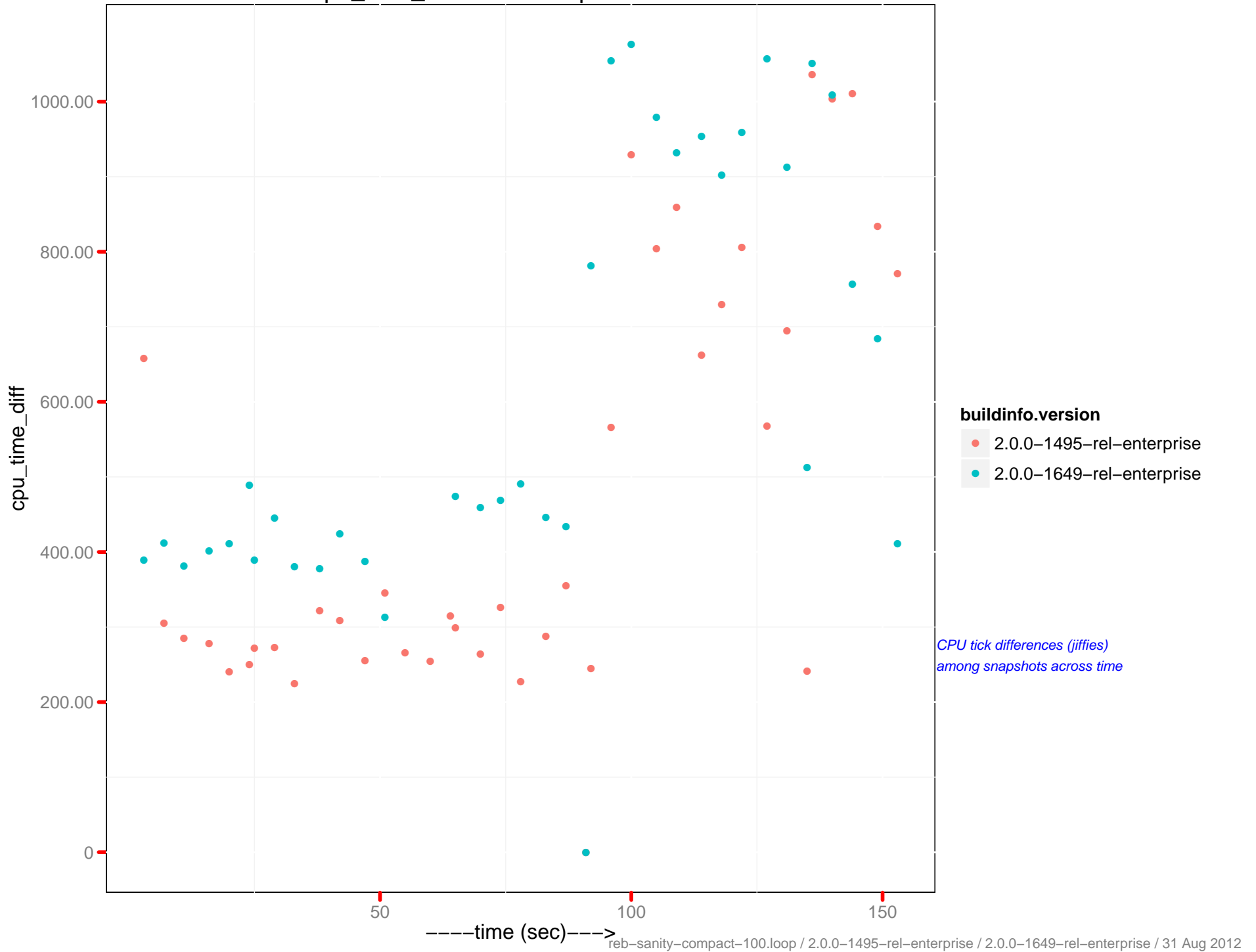
cpu_time_diff : beam.smp - 10.2.1.65



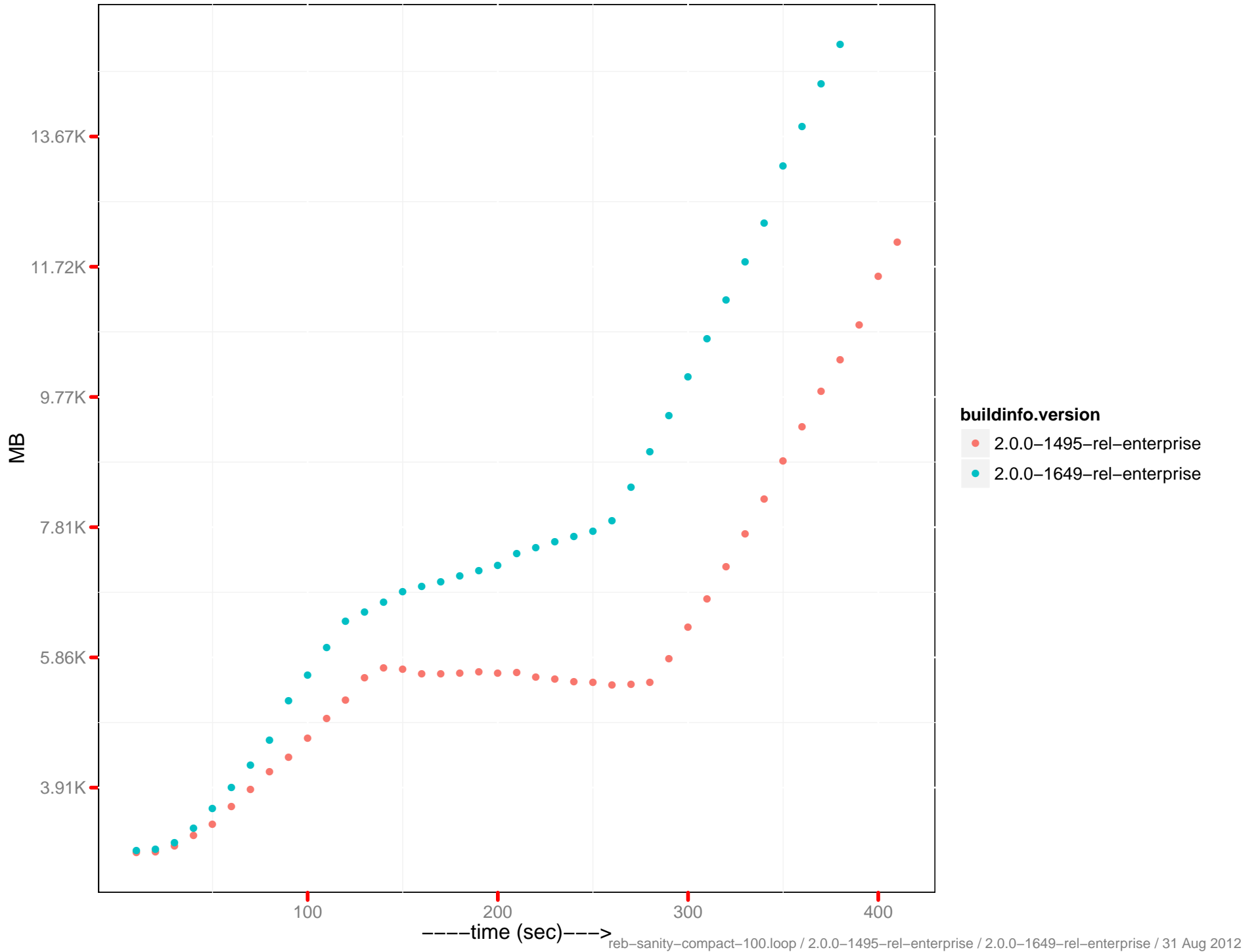
cpu_time_diff: memcached – 10.2.1.67



cpu_time_diff : beam.smp - 10.2.1.67



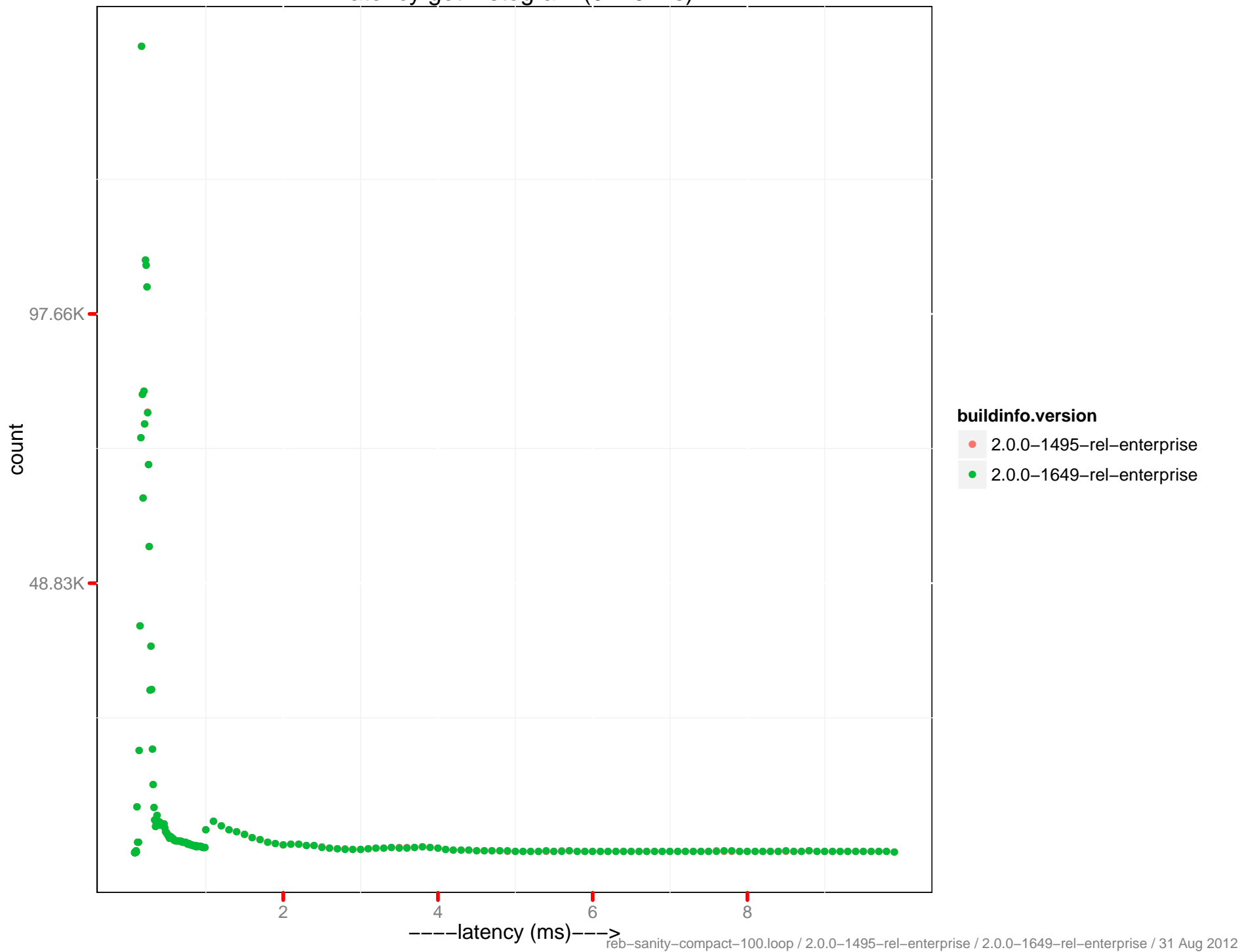
Data disk size



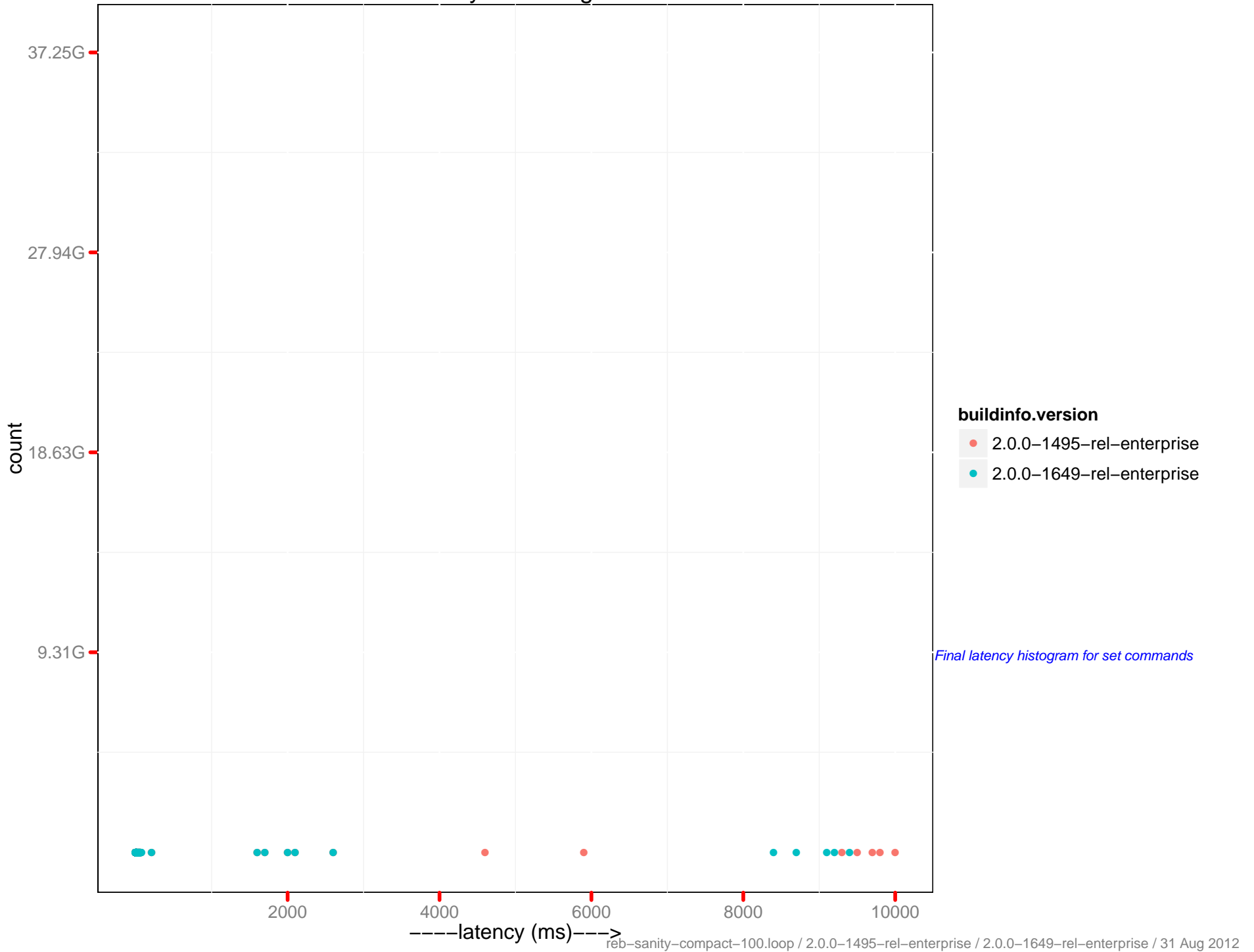
Latency get histogram



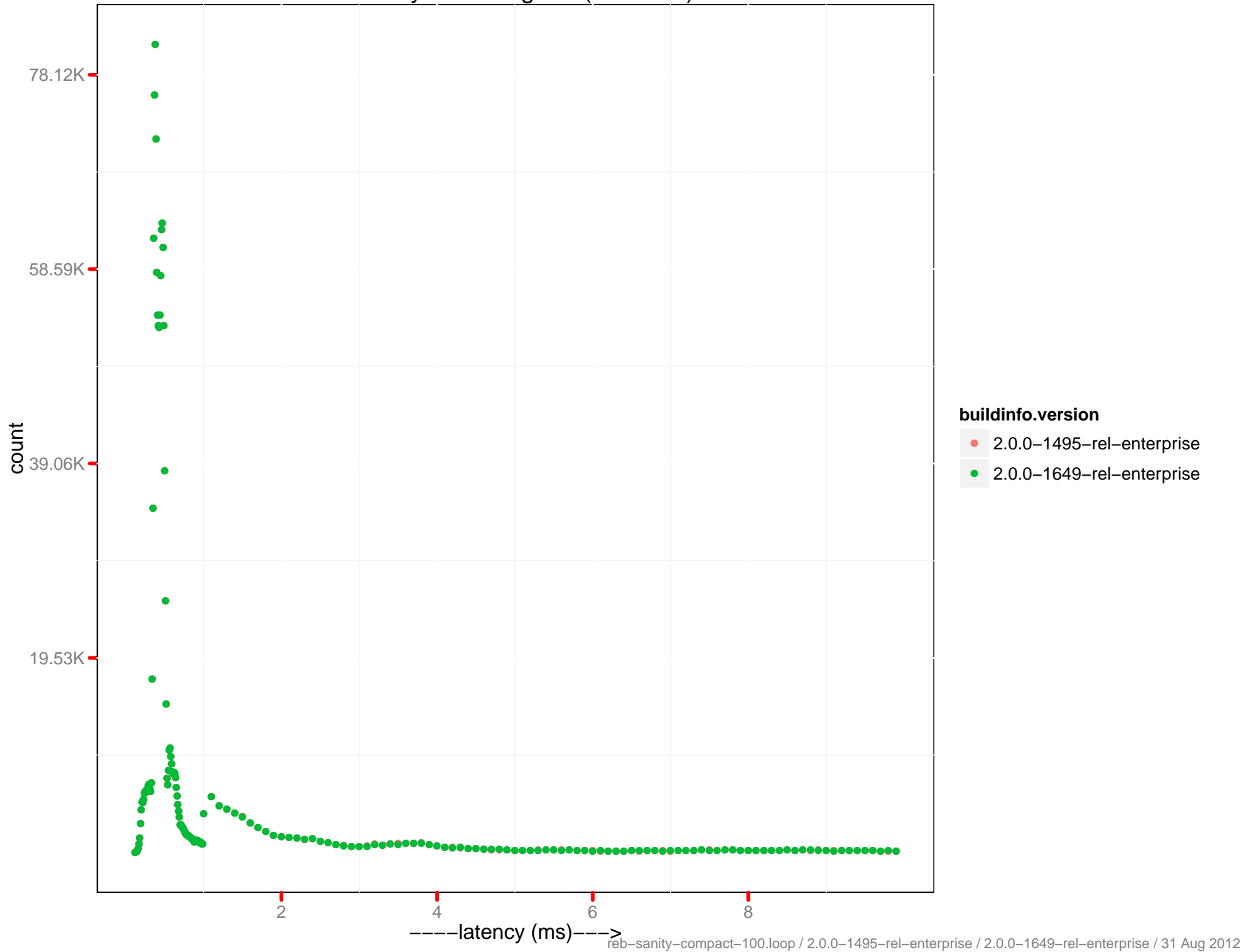
Latency get histogram (0-10 ms)



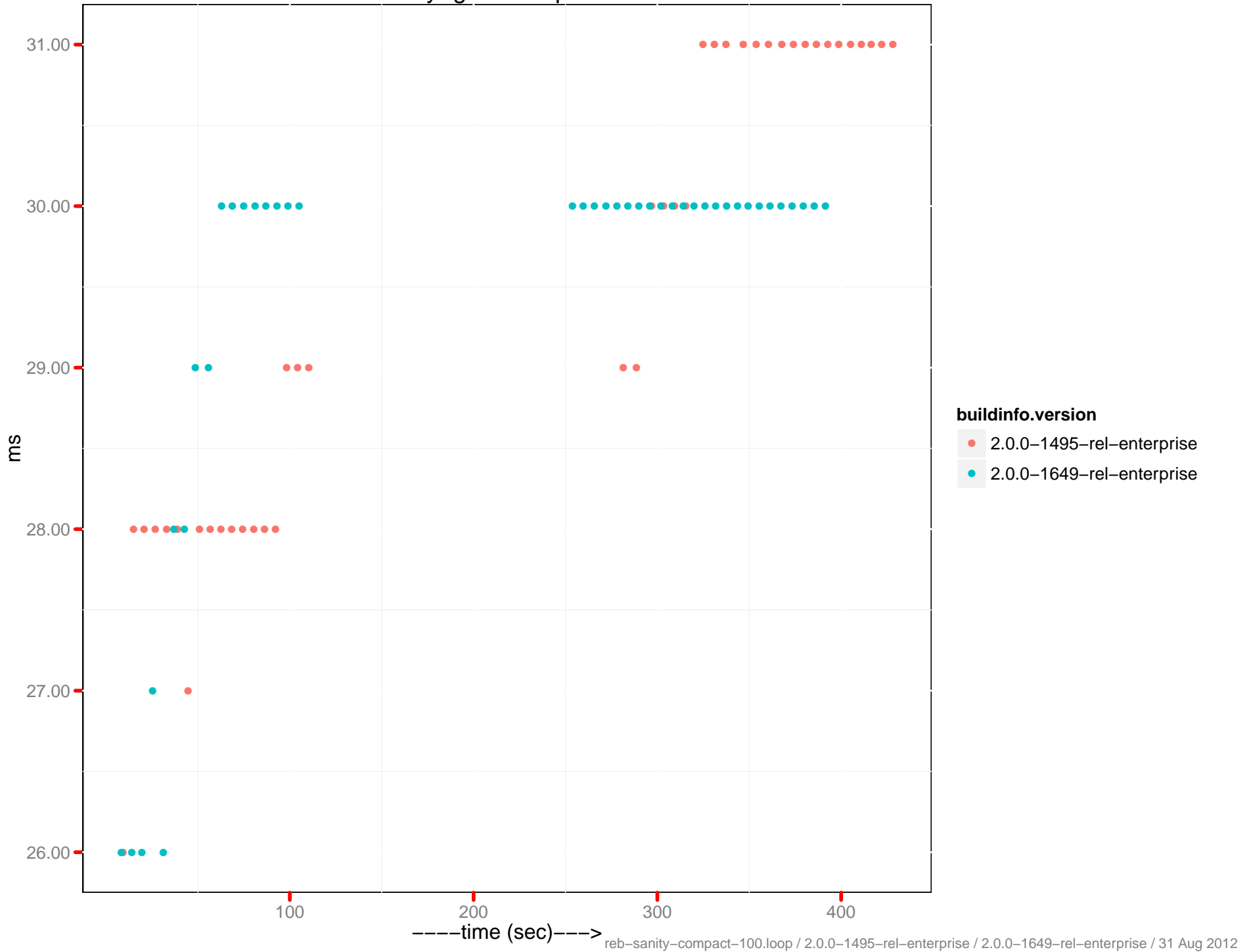
Latency set histogram



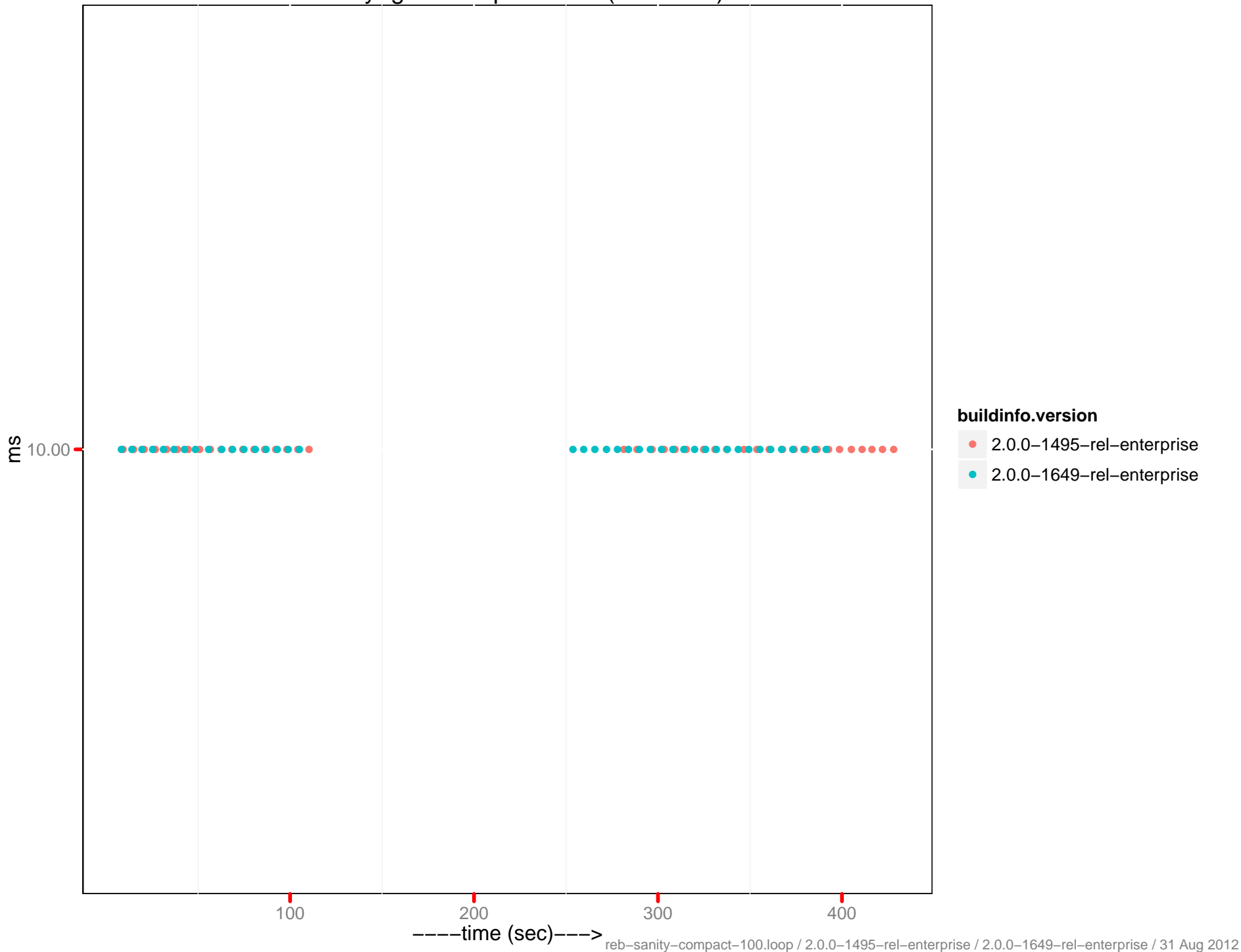
Latency set histogram (0–10 ms)



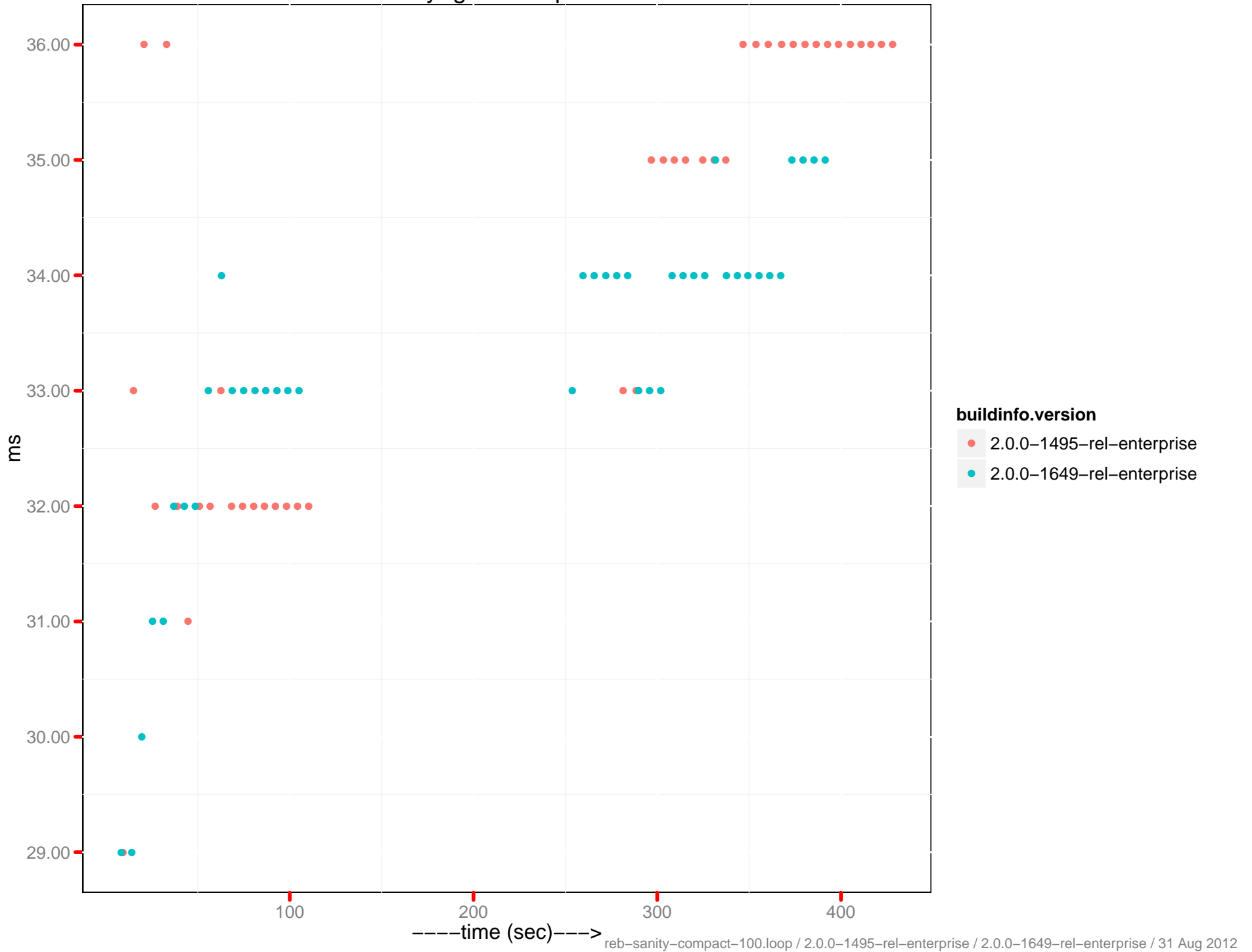
Latency-get 90th percentile



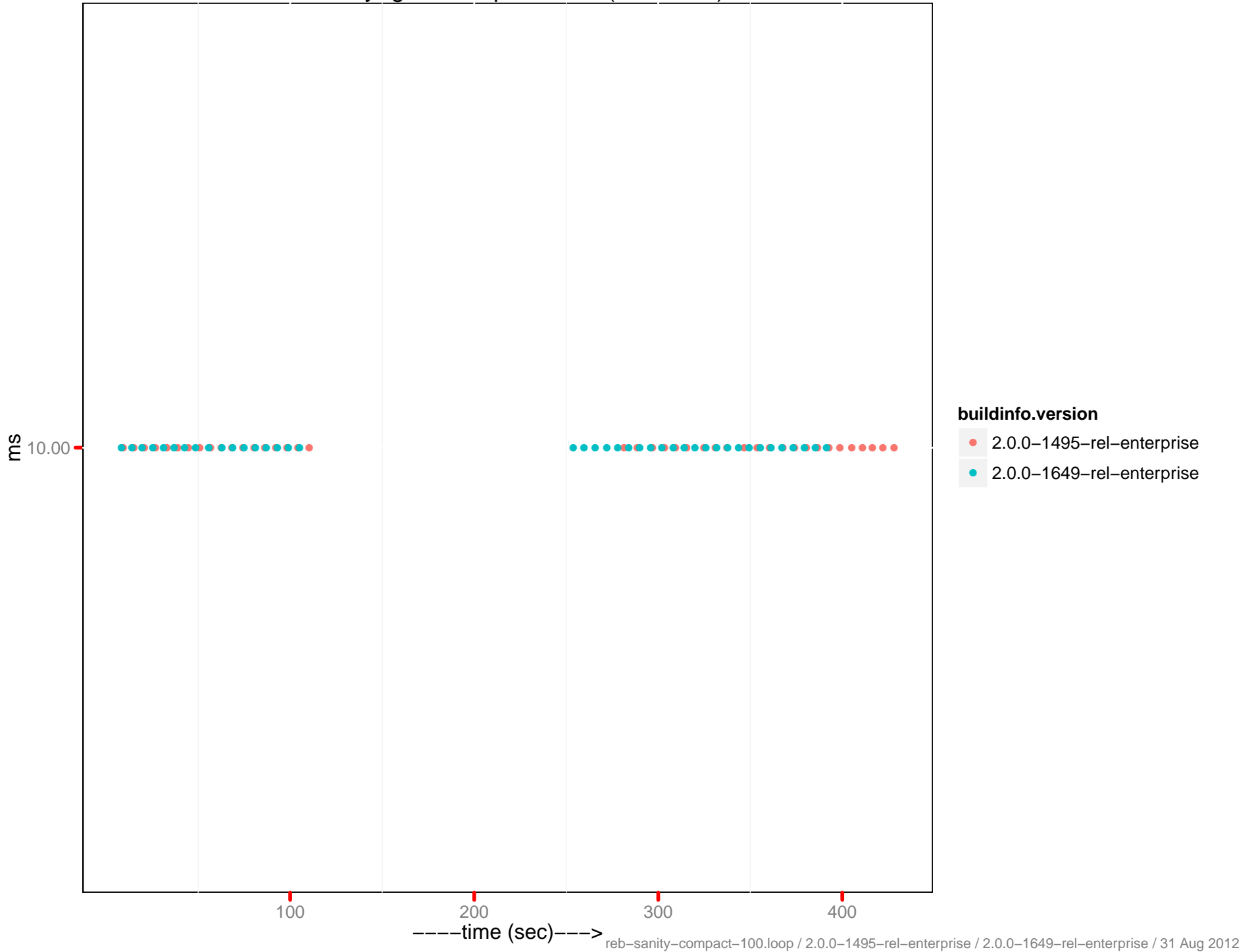
Latency-get 90th percentile (0 - 10ms)



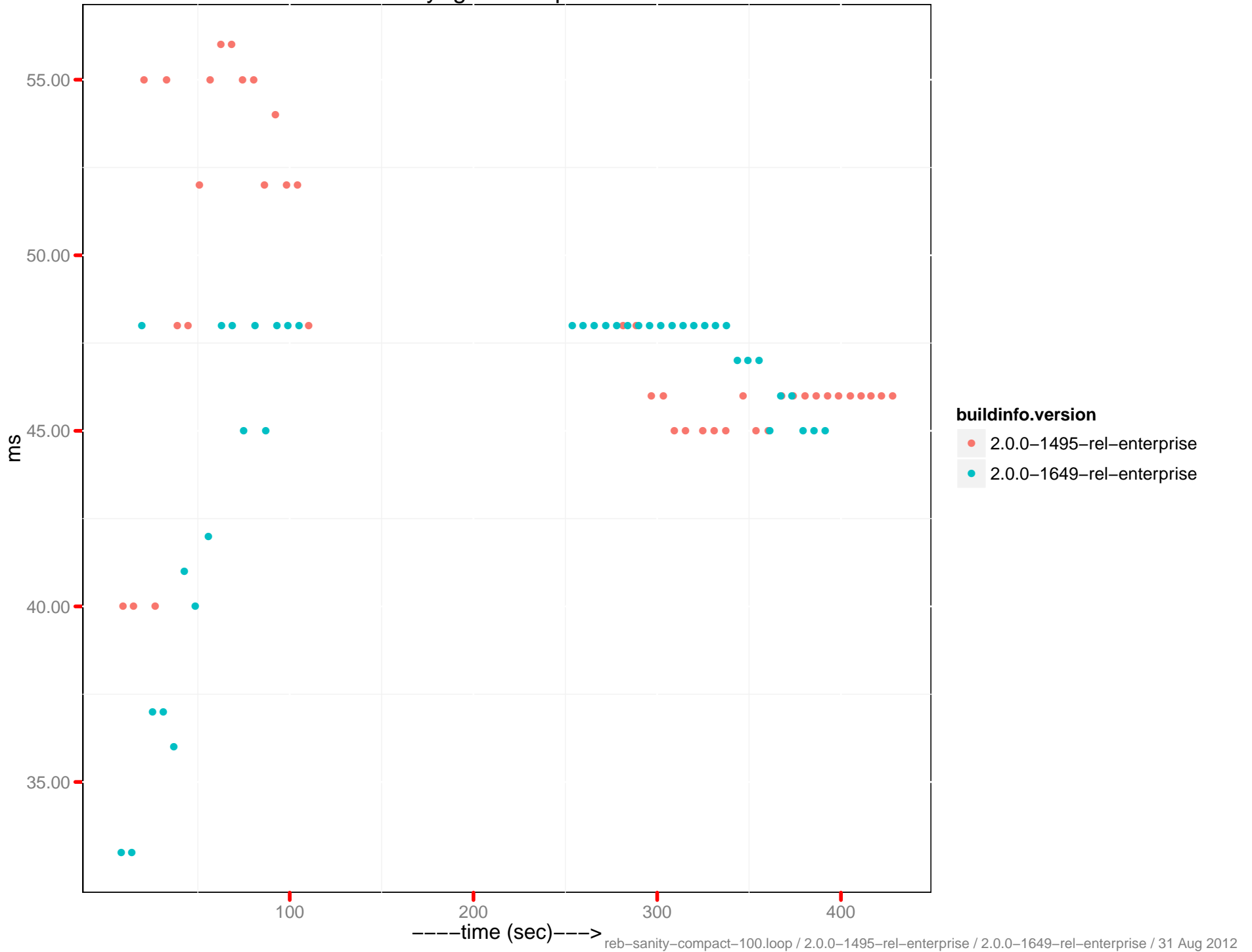
Latency-get 95th percentile



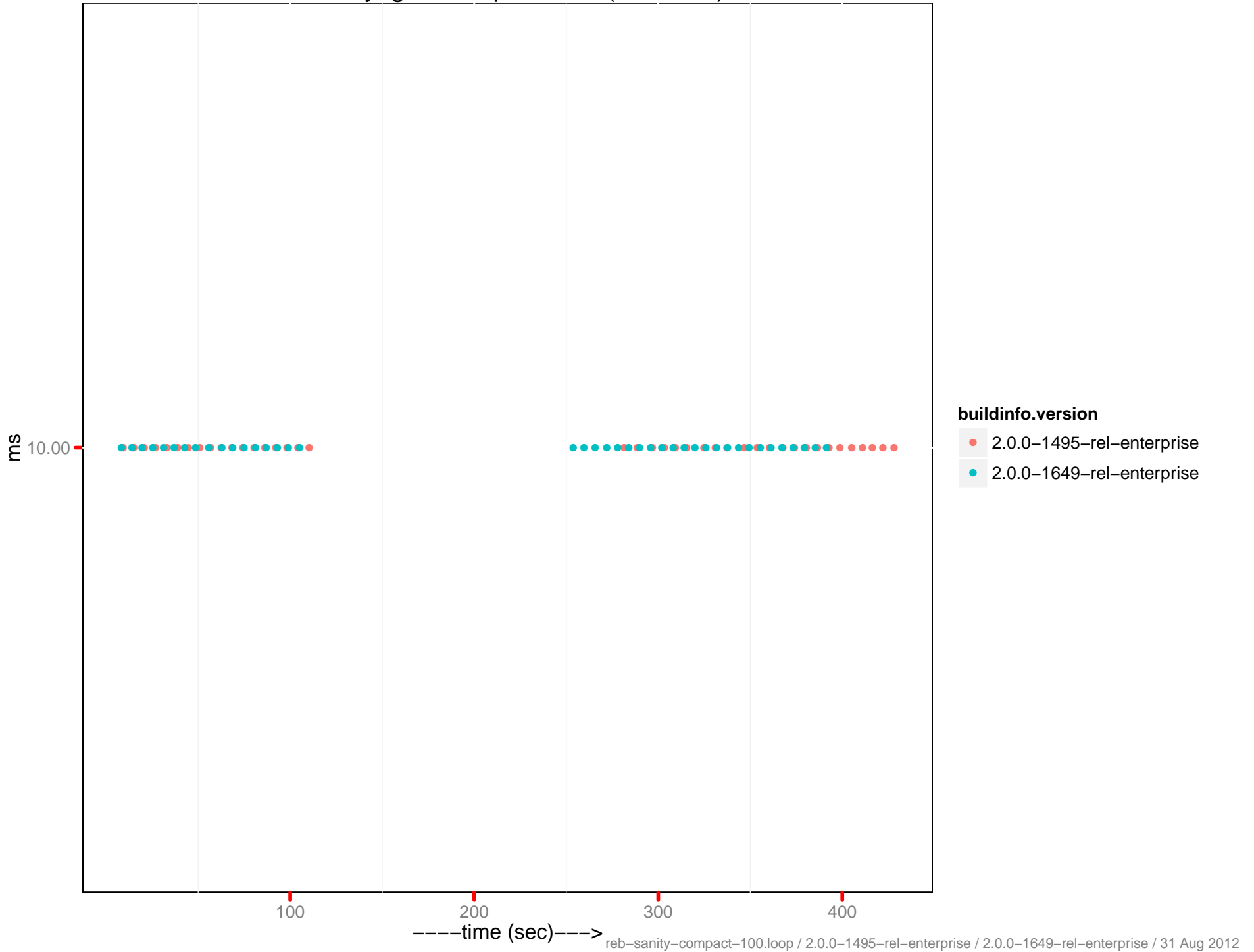
Latency-get 95th percentile (0 - 10ms)



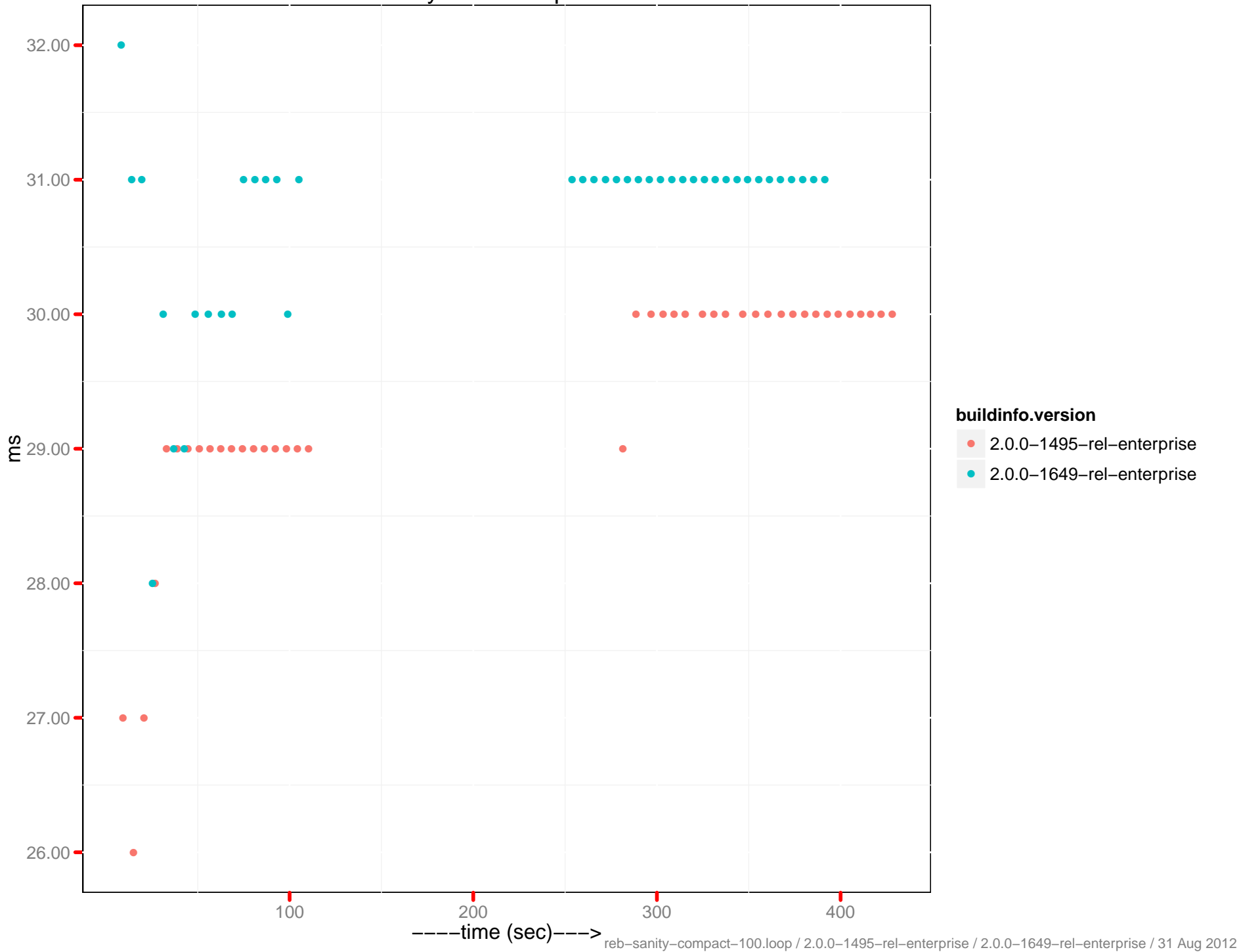
Latency-get 99th percentile



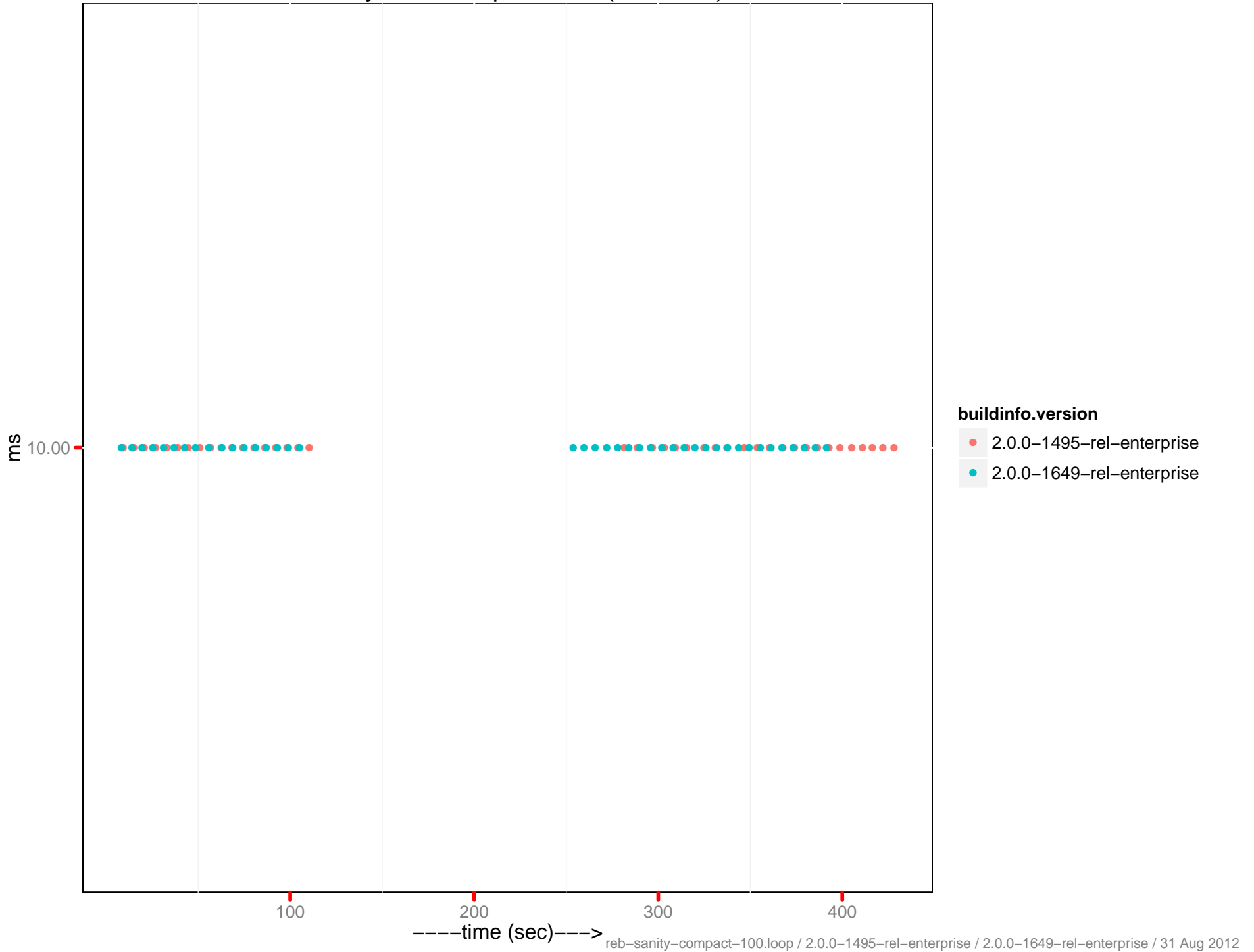
Latency-get 99th percentile (0 - 10ms)



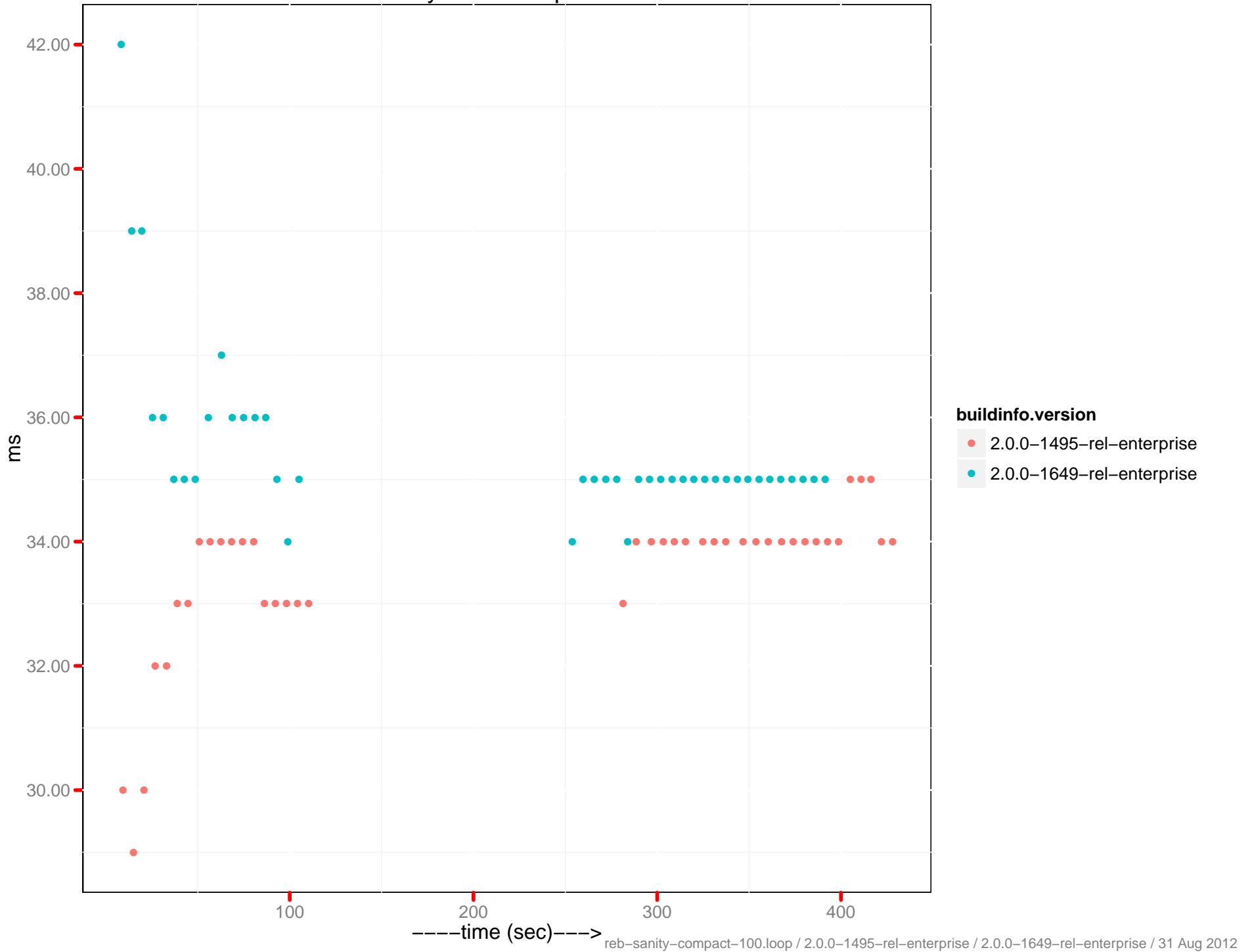
Latency-set 90th percentile



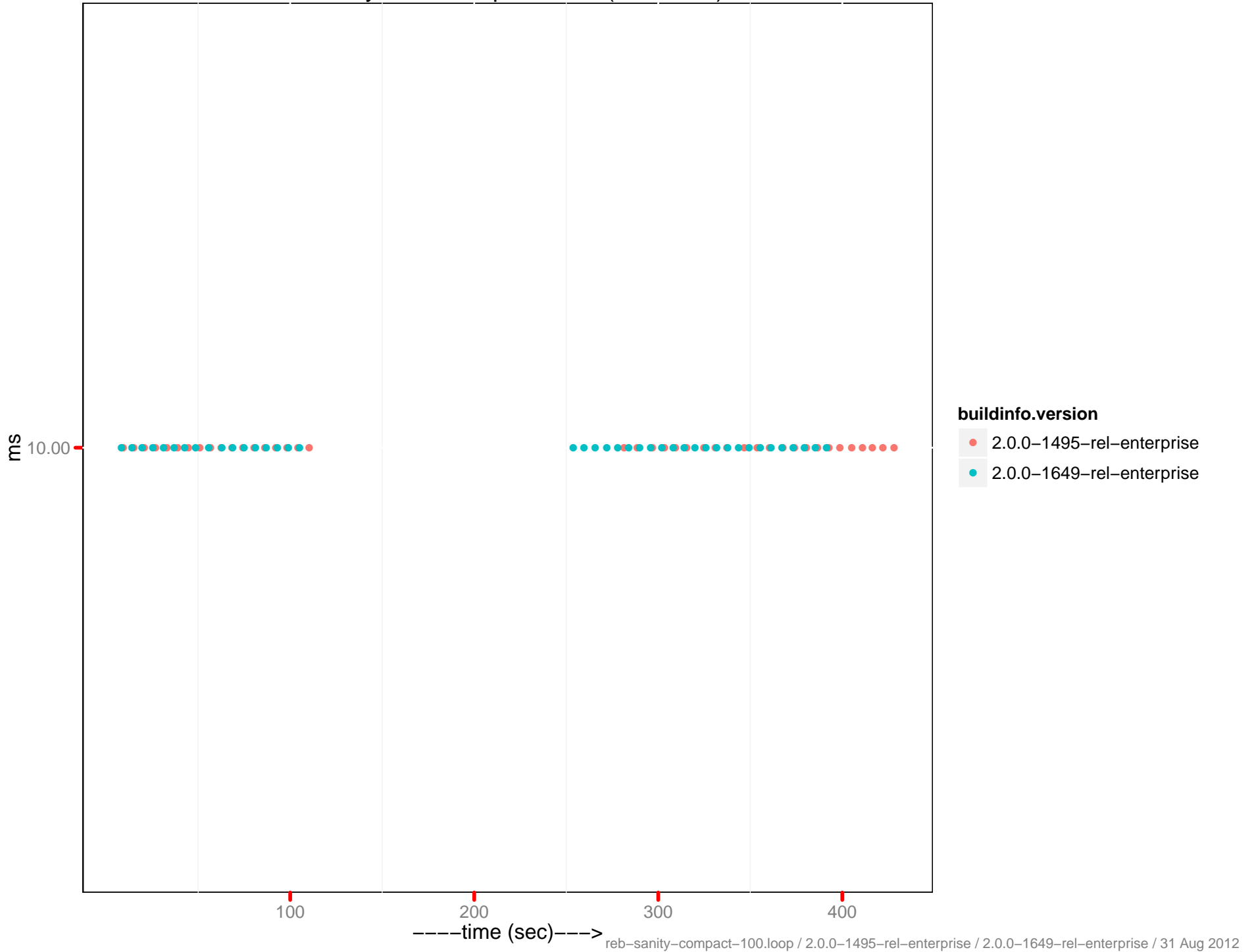
Latency-set 90th percentile (0 – 10ms)



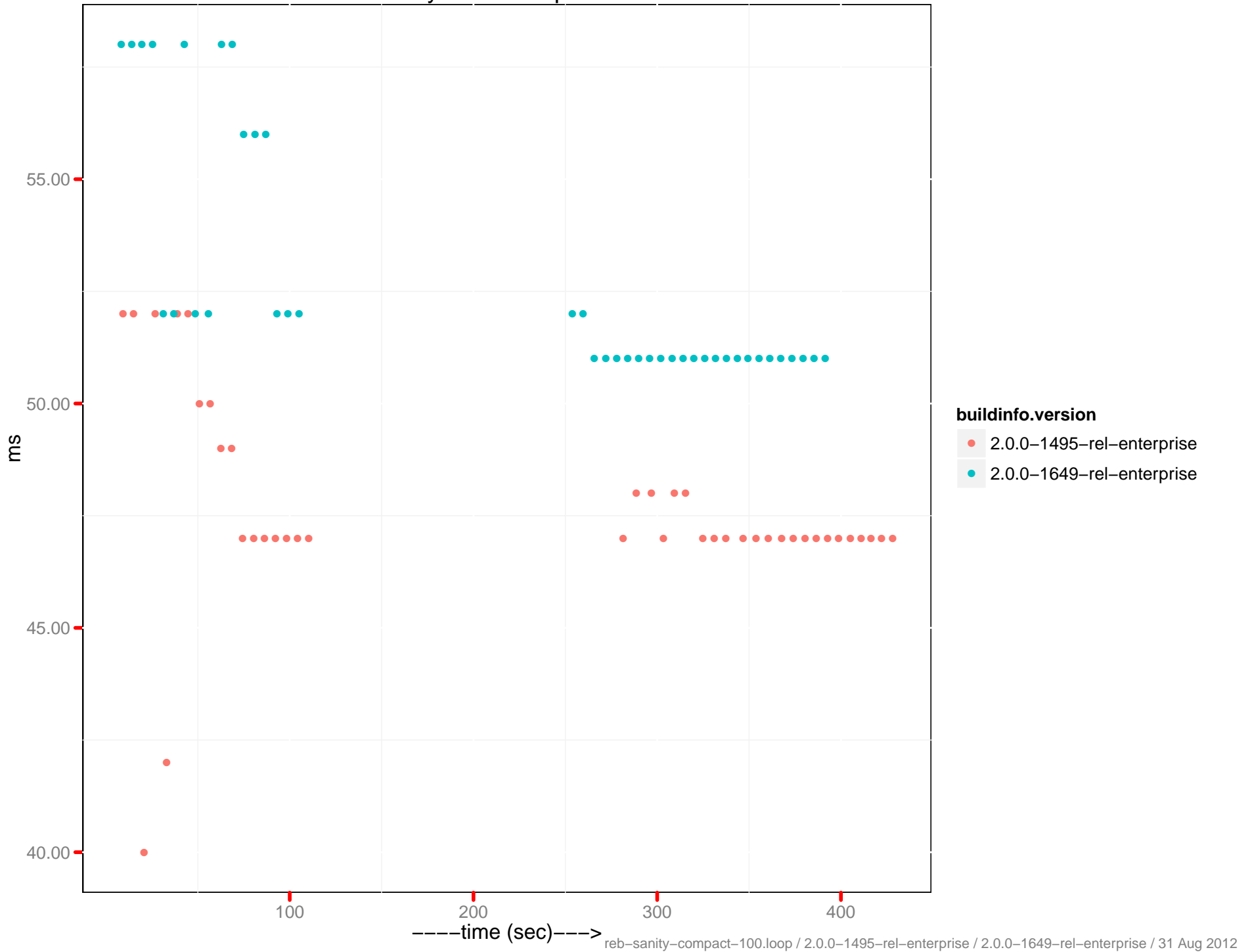
Latency-set 95th percentile



Latency-set 95th percentile (0 – 10ms)

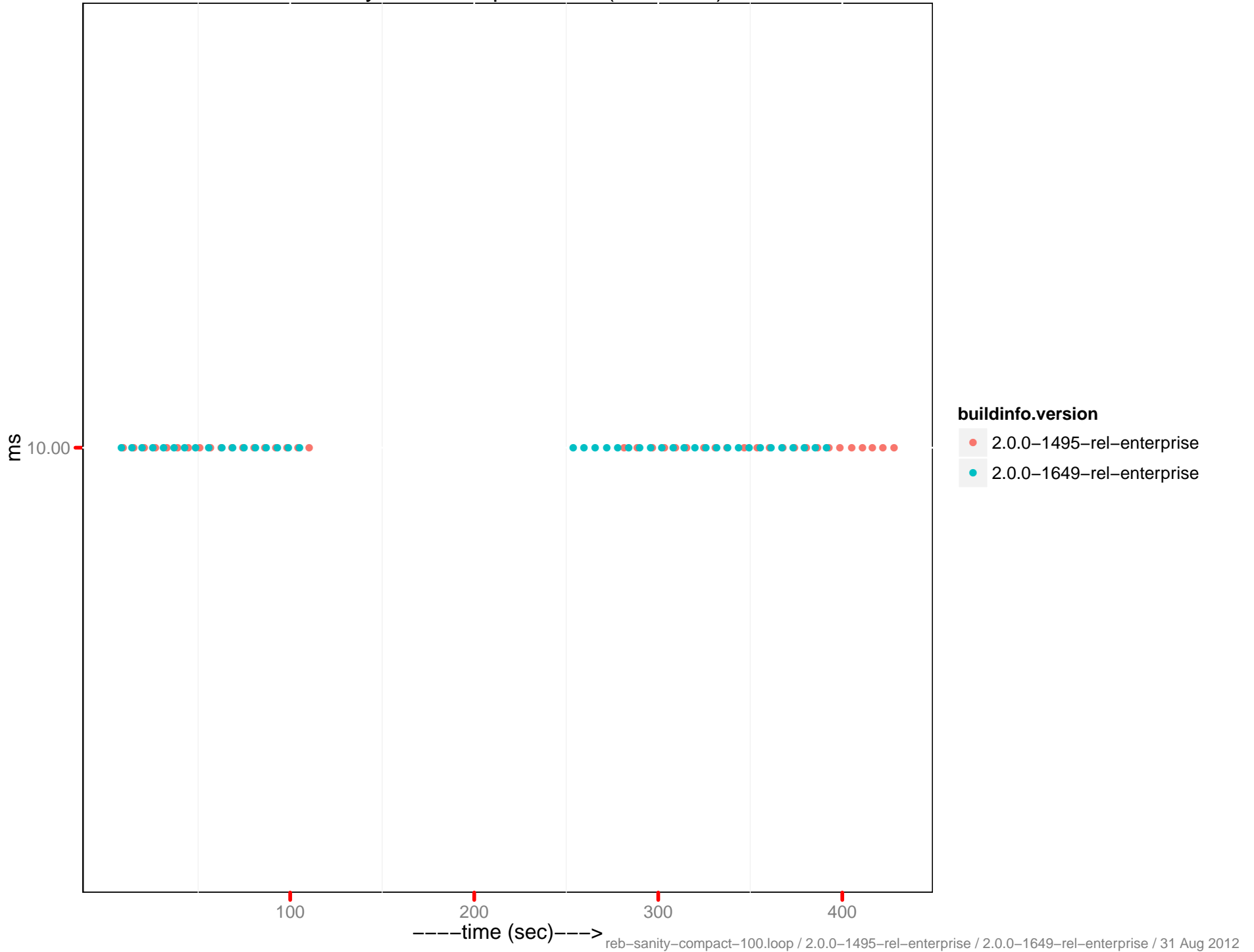


Latency-set 99th percentile

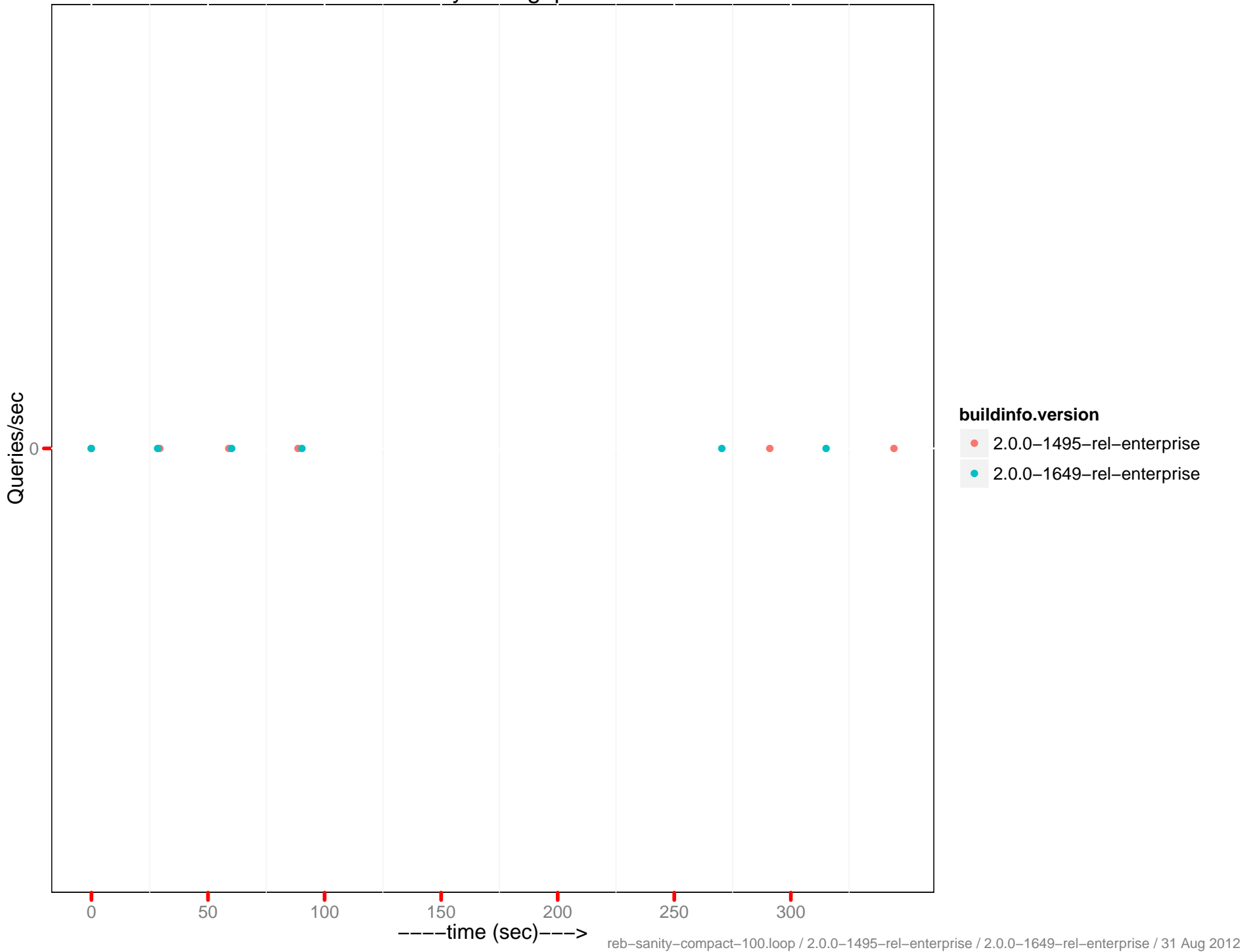


buildinfo.version
● 2.0.0-1495-rel-enterprise
● 2.0.0-1649-rel-enterprise

Latency-set 99th percentile (0 - 10ms)



Query throughput



```
reb-sanity-compact-100.conf
# mixed 0.5M load, 0.1M hot reload, 0.5M access creates, no draining
# rebalance after creating 0.2M items
#
# system memory: 7G per node

performance.eperf.EPerfClient.test_eperf_rebalance

params:

# general
batch=50
kind=nonjson
mem_quota=7000
db_compaction=100

# load phase
hot_init_items=100000
items=500000

# access phase
# Read:Insert:Update:Delete Ratio = 50:4:40:6.
ratio_sets=0.5
ratio_misses=0.05
ratio_creates=0.08
ratio_deletes=0.13
ratio_hot=0.05
ratio_hot_gets=0.099
ratio_hot_sets=0.099
ratio_expirations=0.03
max_creates=500000

# rebalance
rebalance_after=200000
num_nodes_after=2

# control (defaults: pytests/performance/perf_defaults.py)
load_wait_until_drained=1
loop_wait_until_drained=0
mcsoda_heartbeat=3
tear_down=1
tear_down_proxy=1
tear_down_bucket=0
tear_down_cluster=1
tear_down_on_setup=0
```

```
vesta.ini
[global]
username:root
password:couchbase
port:8091
data_path:/data

[servers]
1:10.2.1.65
2:10.2.1.66
3:10.2.1.67
4:10.2.1.68

[clients]
1:10.2.1.60

[membase]
rest_username:Administrator
rest_password:password

[dashboard]
1:dashboard.hq.couchbase.com:80
```