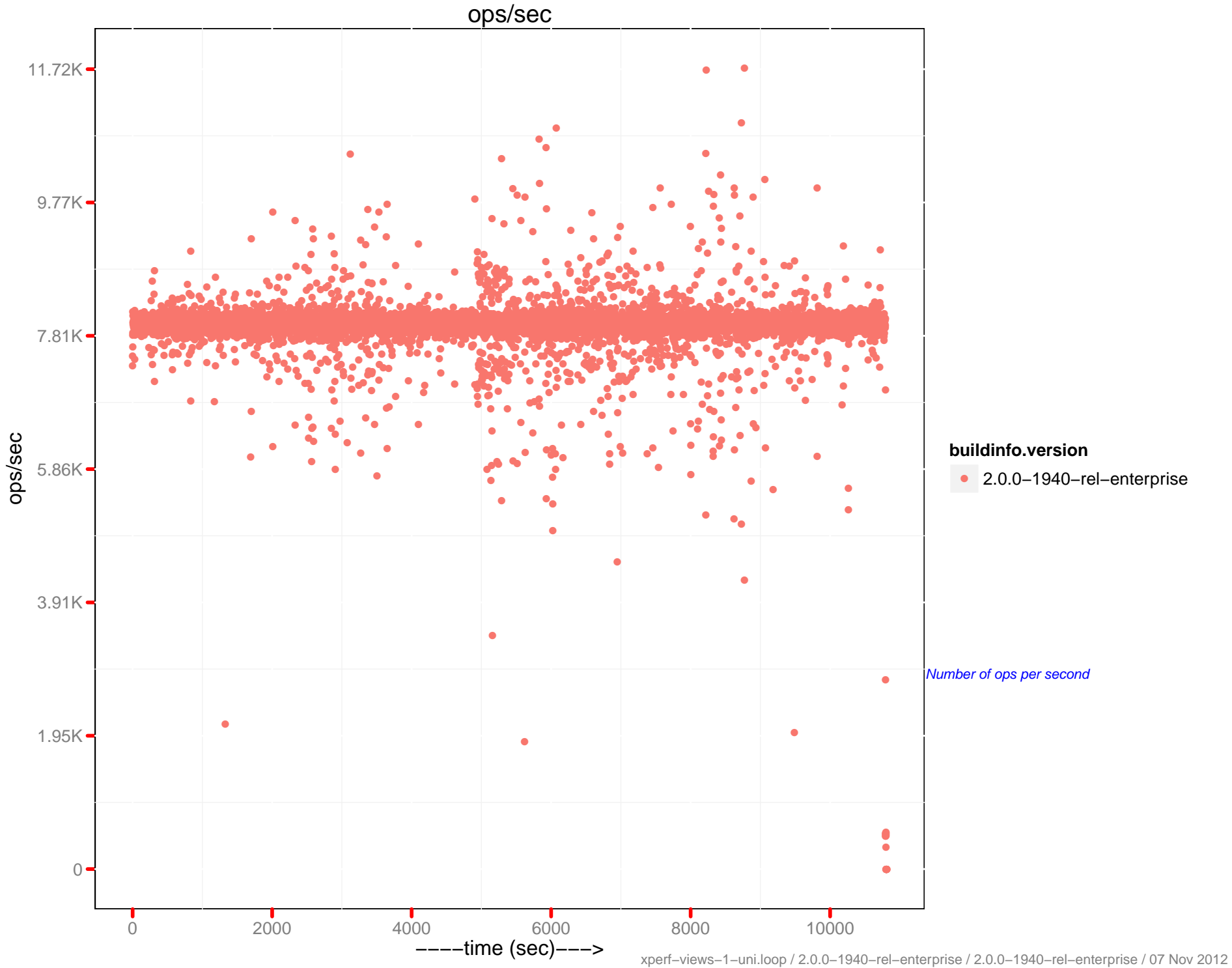
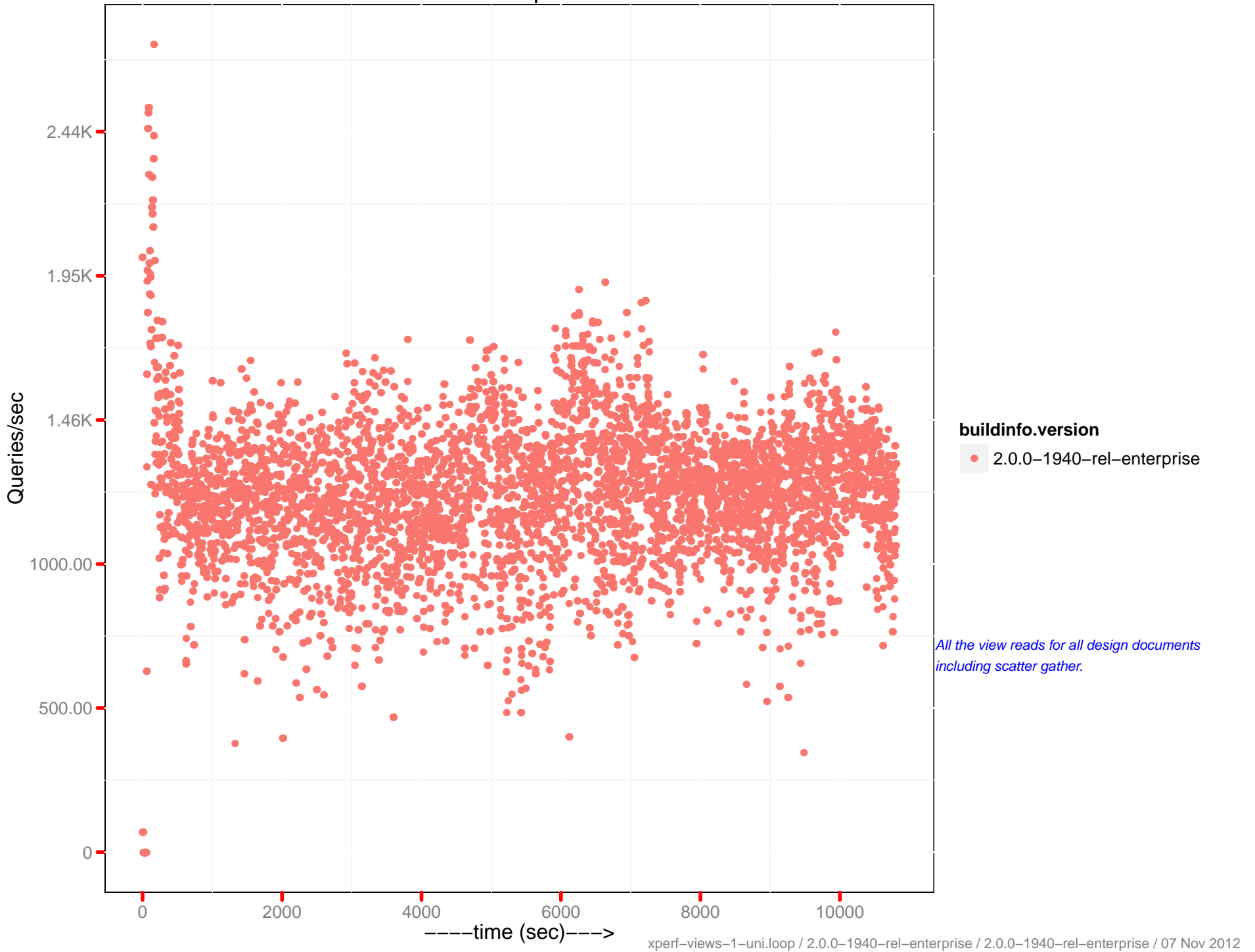


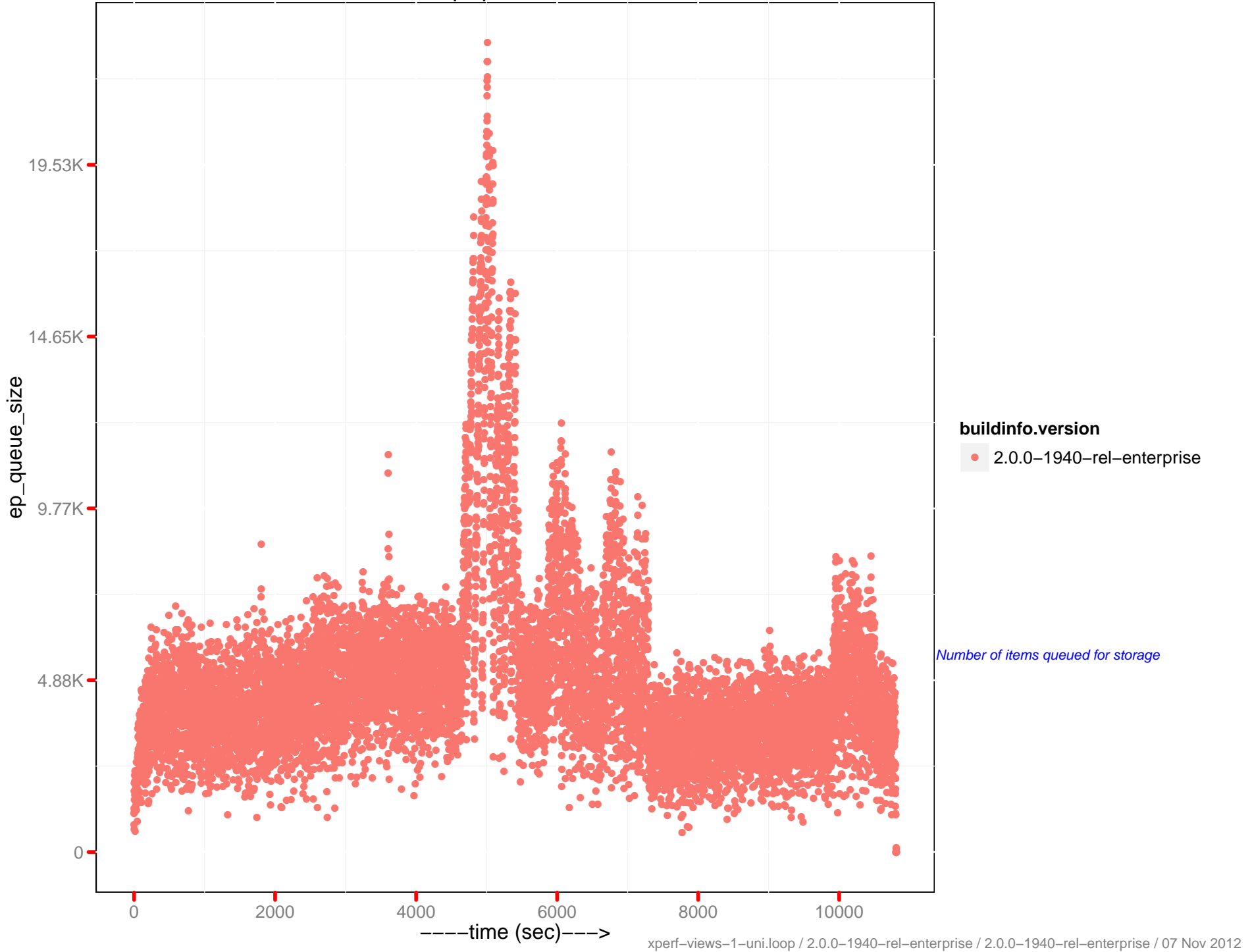
	2.0.0 – 1940	2.0.0 – 1940
<i>Runtime (in hr)</i>	3.01	NA
<i>Avg. Drain Rate</i>	848.88	NANA
<i>Peak Disk (GB)</i>	79.13	NA
<i>Peak Memory (GB)</i>	13.39	NA
<i>Avg. OPS</i>	7.97K	NANA
<i>Avg. mem memcached (GB)</i>	12.64	NA
<i>Avg. mem beam.smp (MB)</i>	671.8	NA
<i>Avg. CPU rate (%)</i>	71.68	NA
<i>Latency-get (90th) (ms)</i>	3.97	NA
<i>Latency-get (95th) (ms)</i>	6.55	NA
<i>Latency-get (99th) (ms)</i>	23.75	NA
<i>Latency-set (90th) (ms)</i>	3.94	NA
<i>Latency-set (95th) (ms)</i>	6.47	NA
<i>Latency-set (99th) (ms)</i>	23.03	NA
<i>Latency-query (80th) (ms)</i>	57.2	NA
<i>Latency-query (90th) (ms)</i>	76.51	NA
<i>Latency-query (95th) (ms)</i>	103.55	NA
<i>Latency-query (99th) (ms)</i>	599.32	NA
<i>Latency-query (99.9th) (ms)</i>	1825.61	NA
<i>Avg. QPS</i>	314.91	NA
<i>Avg. XDC ops/sec</i>	NaN	NA
<i>Avg. XDC docs to replicate</i>	8374.8	NA
<i>Rebalance Time (sec)</i>	0	NA
<i>Testrunner Version</i>	0c43aa6	NA



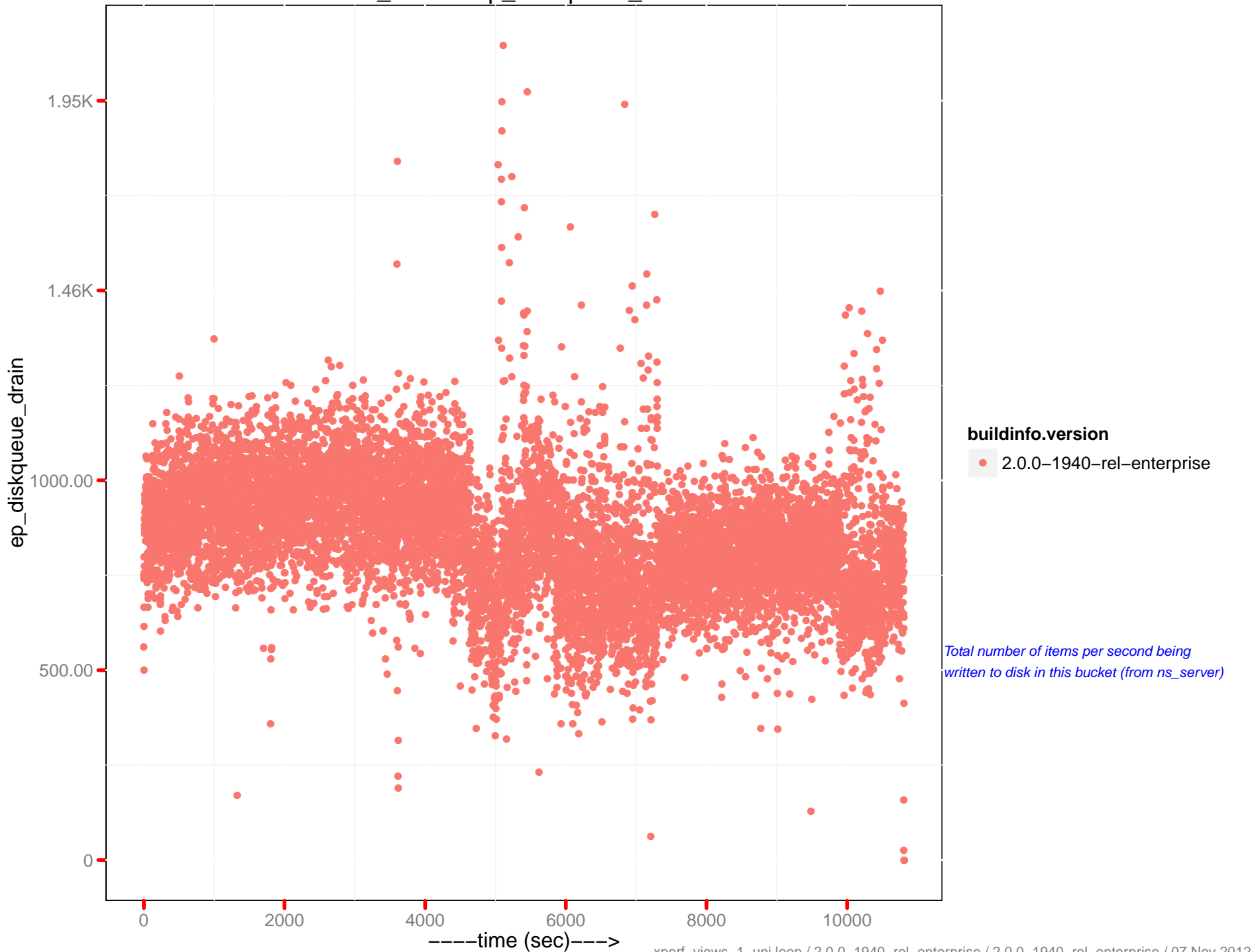
View read per sec.



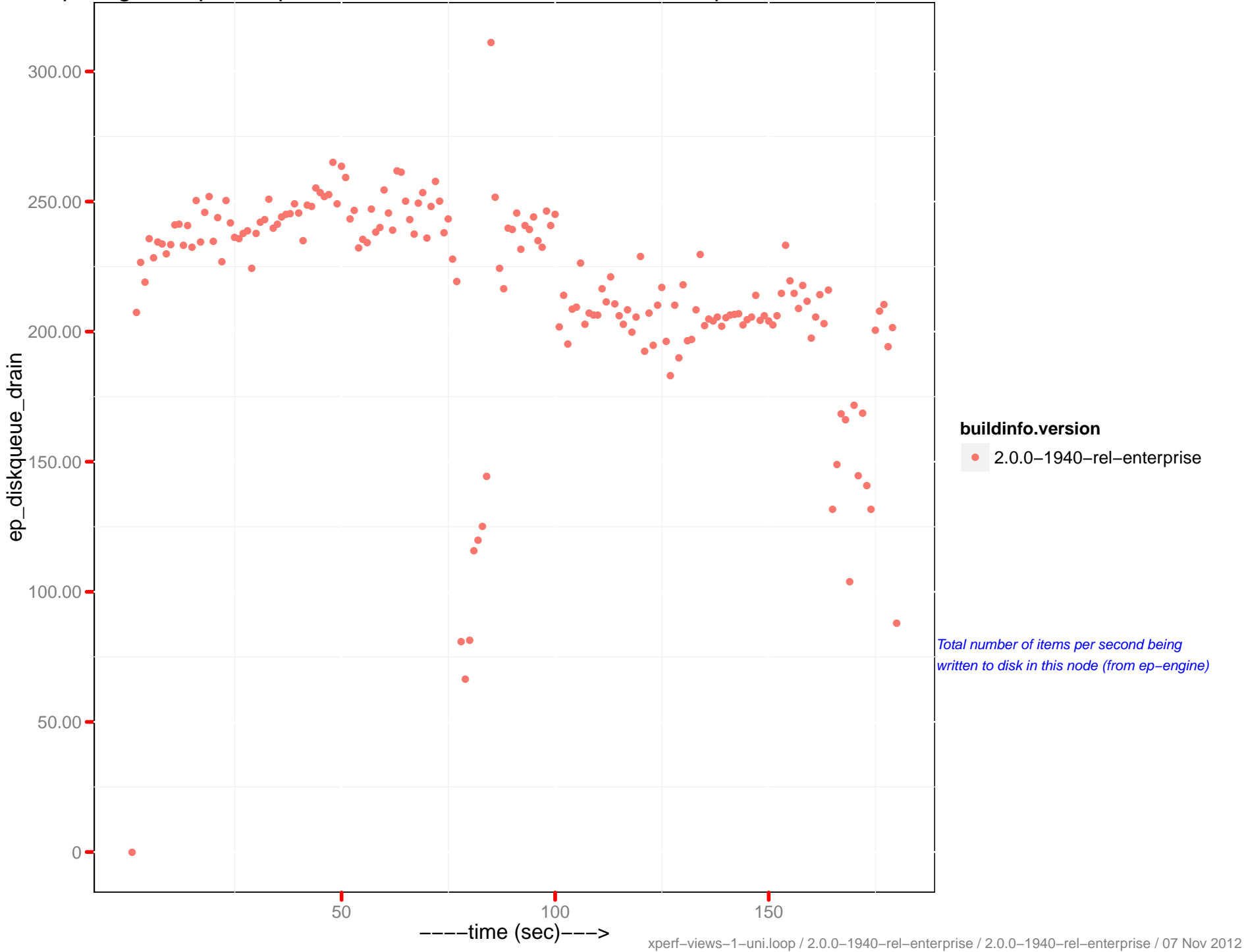
ep queue size



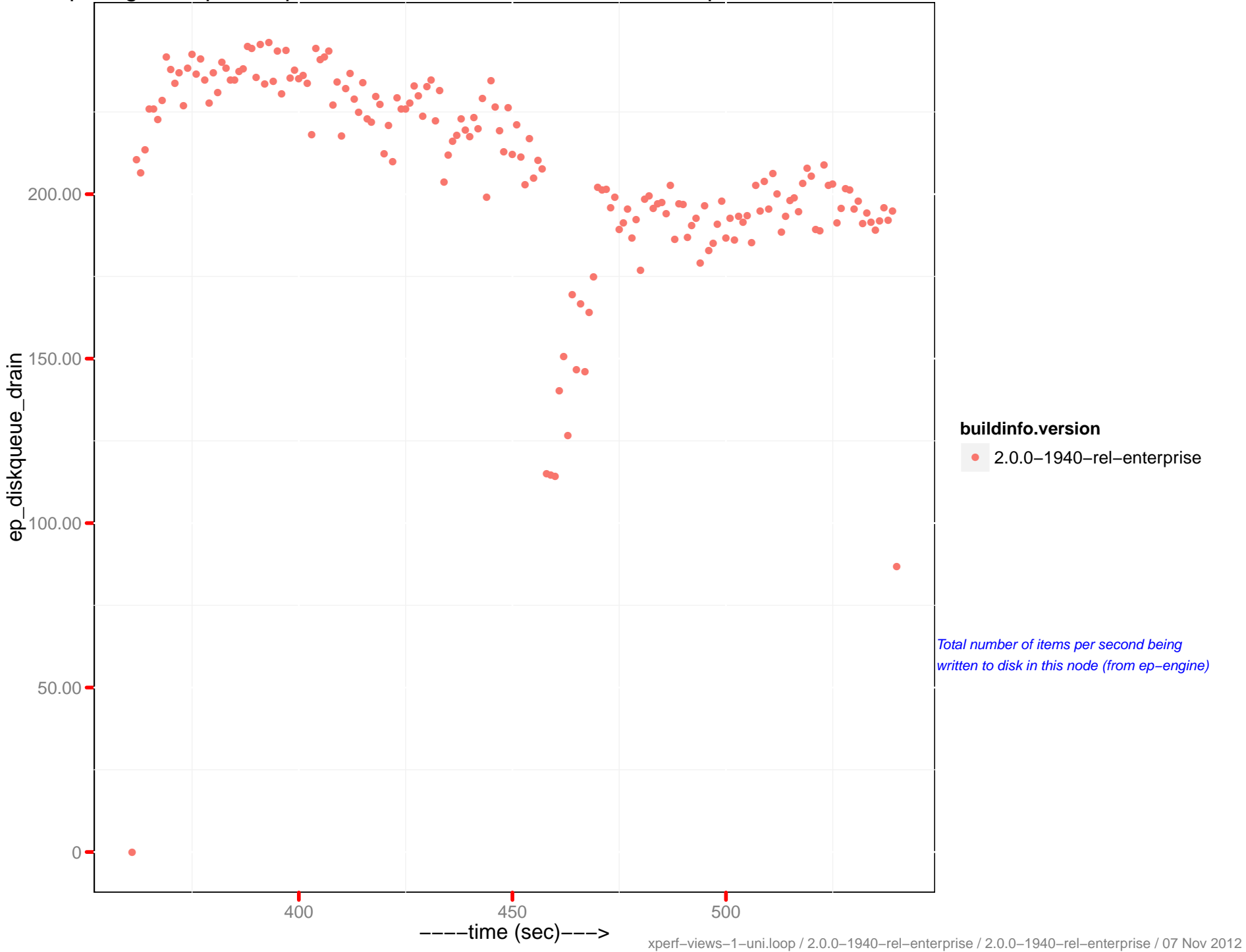
ns_server: ep_diskqueue_drain



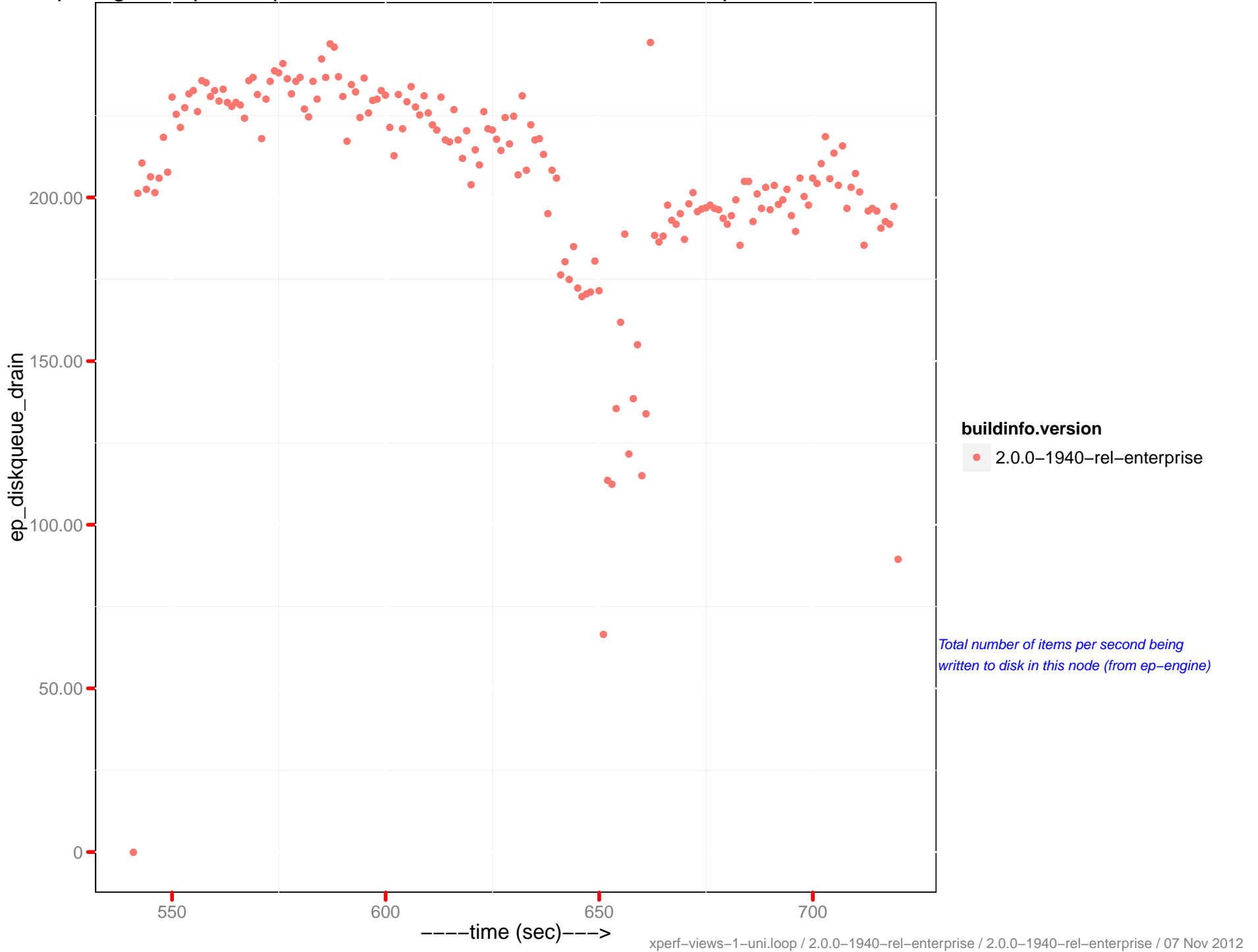
ep-engine : ep_diskqueue_drain - ec2-107-20-6-150.compute-1.amazonaws.com



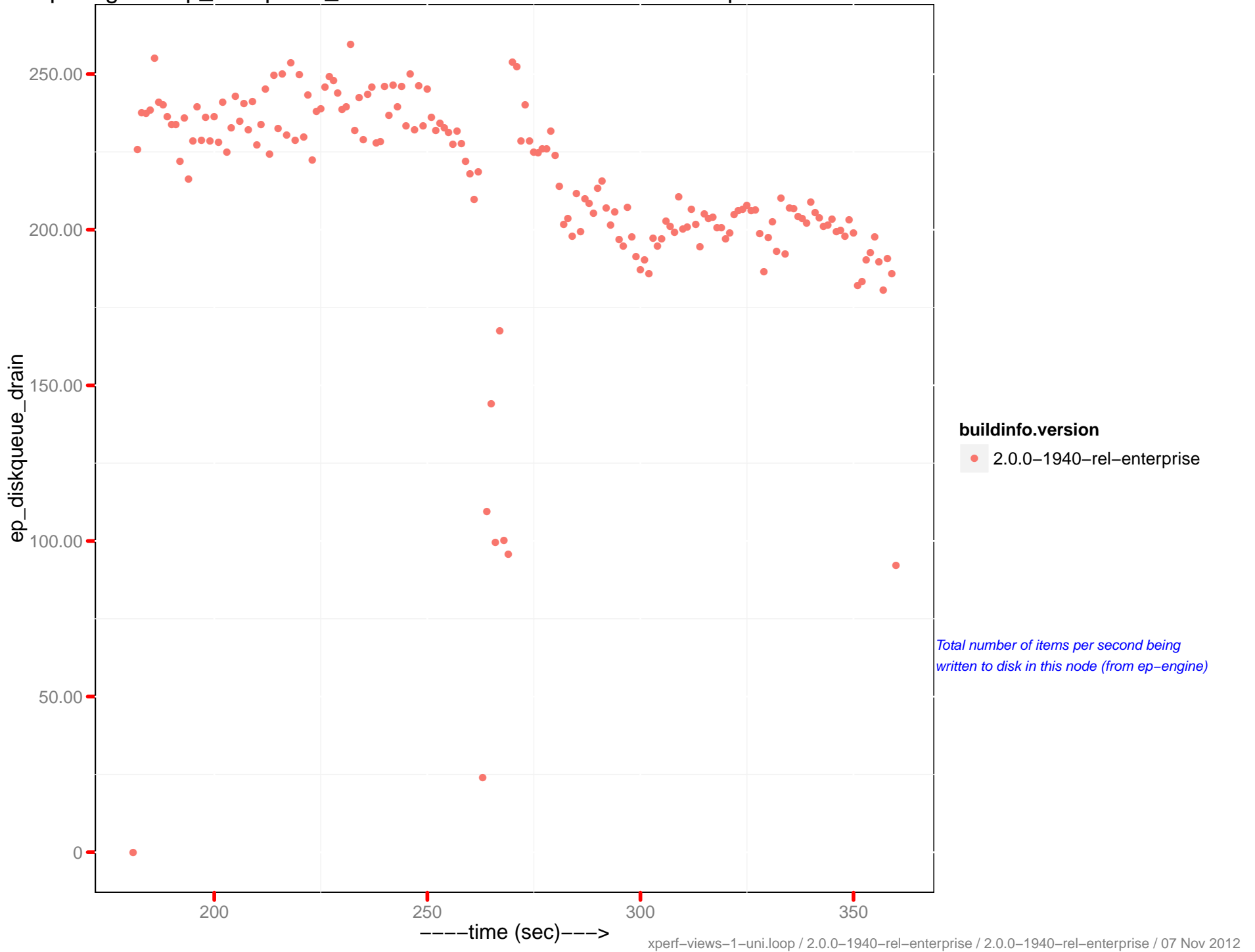
ep-engine : ep_diskqueue_drain - ec2-23-20-61-95.compute-1.amazonaws.com



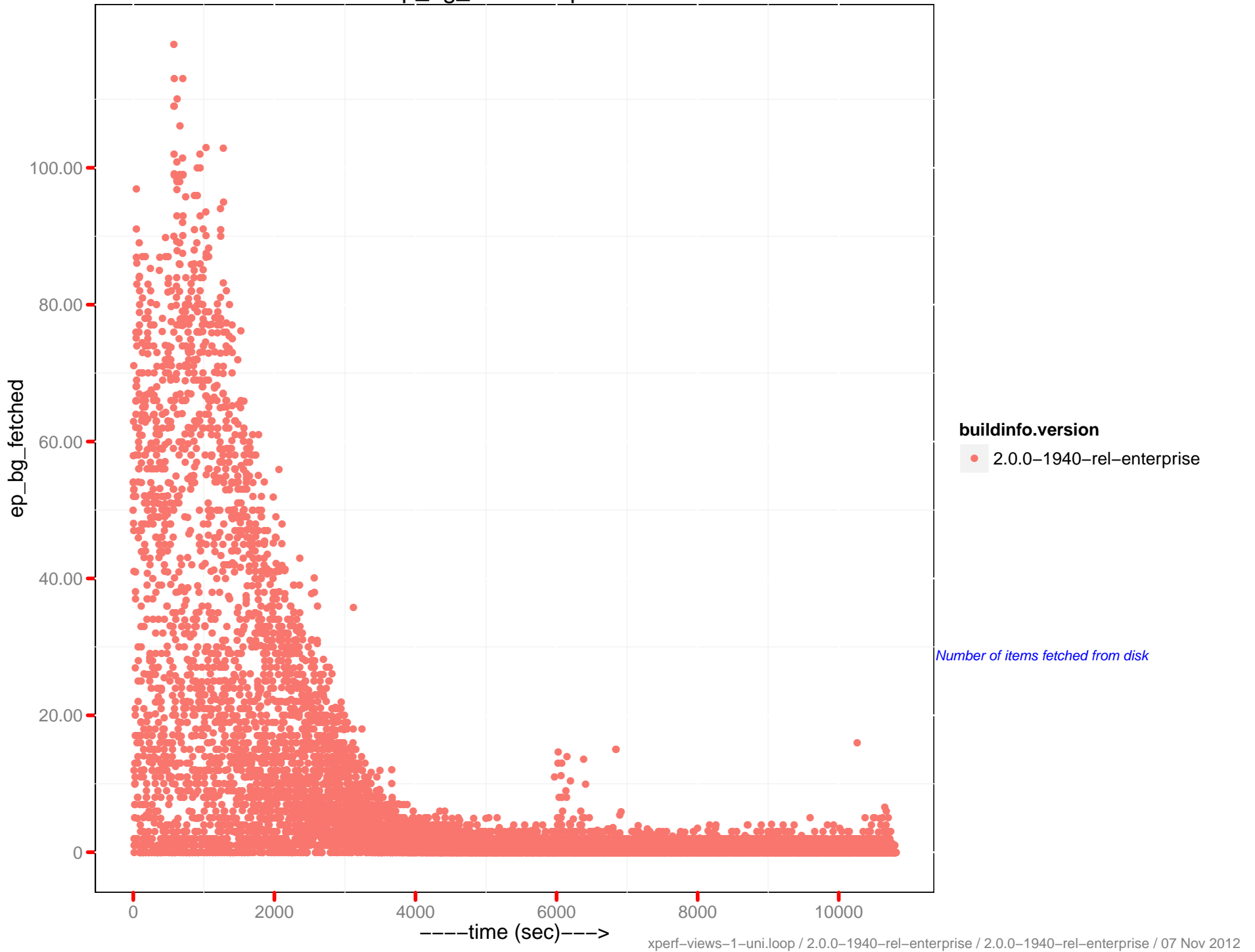
ep-engine : ep_diskqueue_drain - ec2-23-22-246-165.compute-1.amazonaws.com



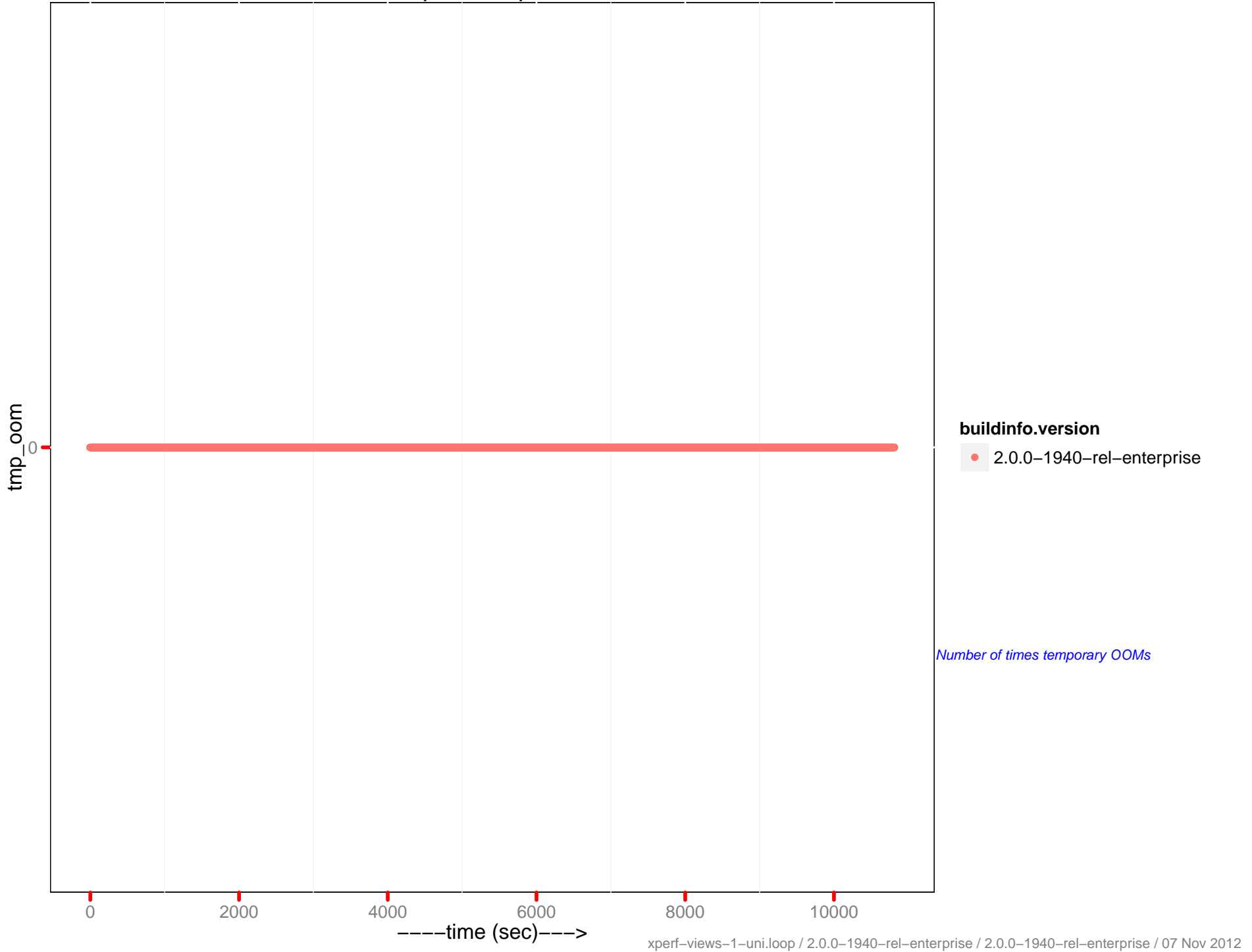
ep-engine : ep_diskqueue_drain - ec2-54-242-190-218.compute-1.amazonaws.com



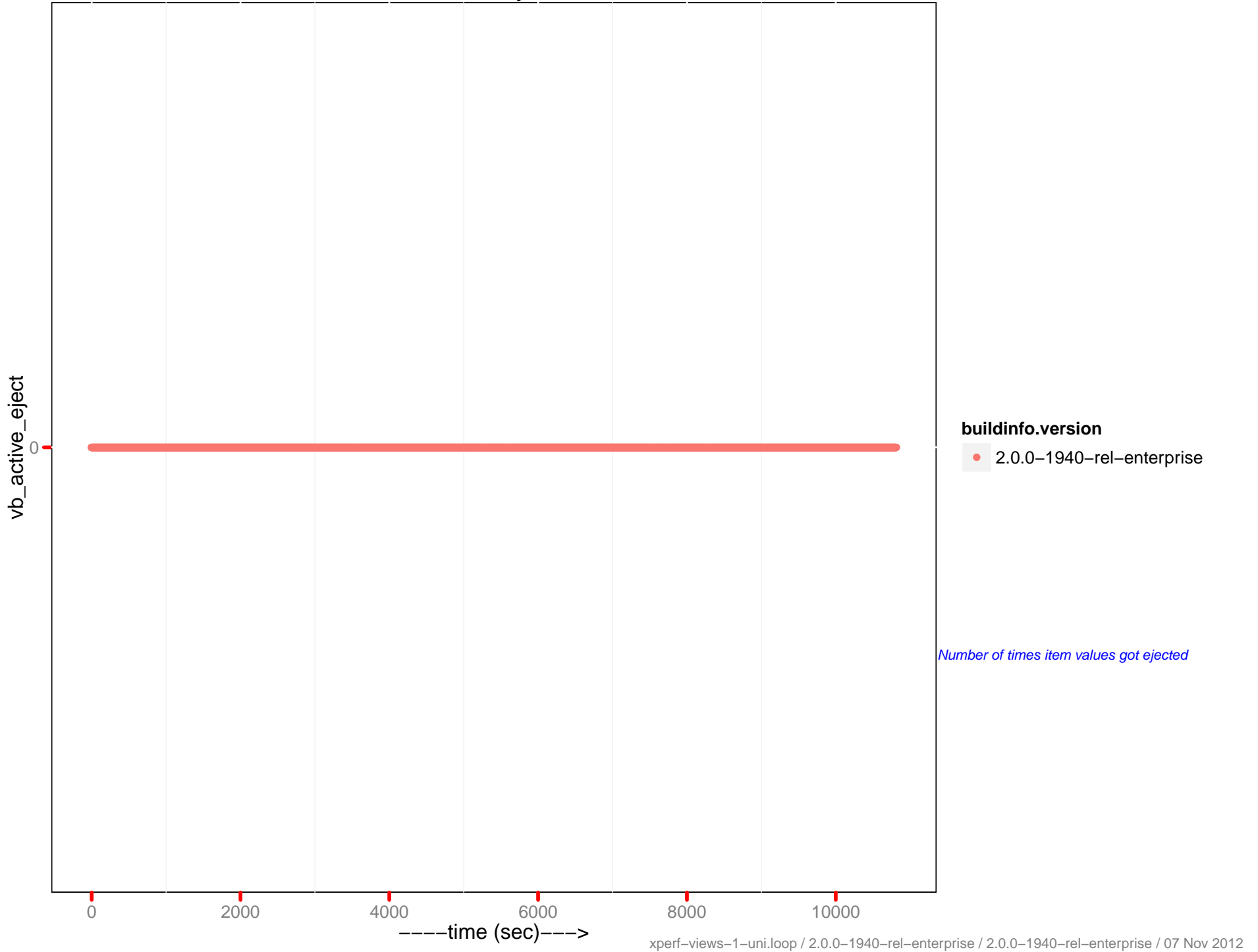
ep_bg_fetched ops/sec



tmp_oom ops/sec



vb_active_eject/sec

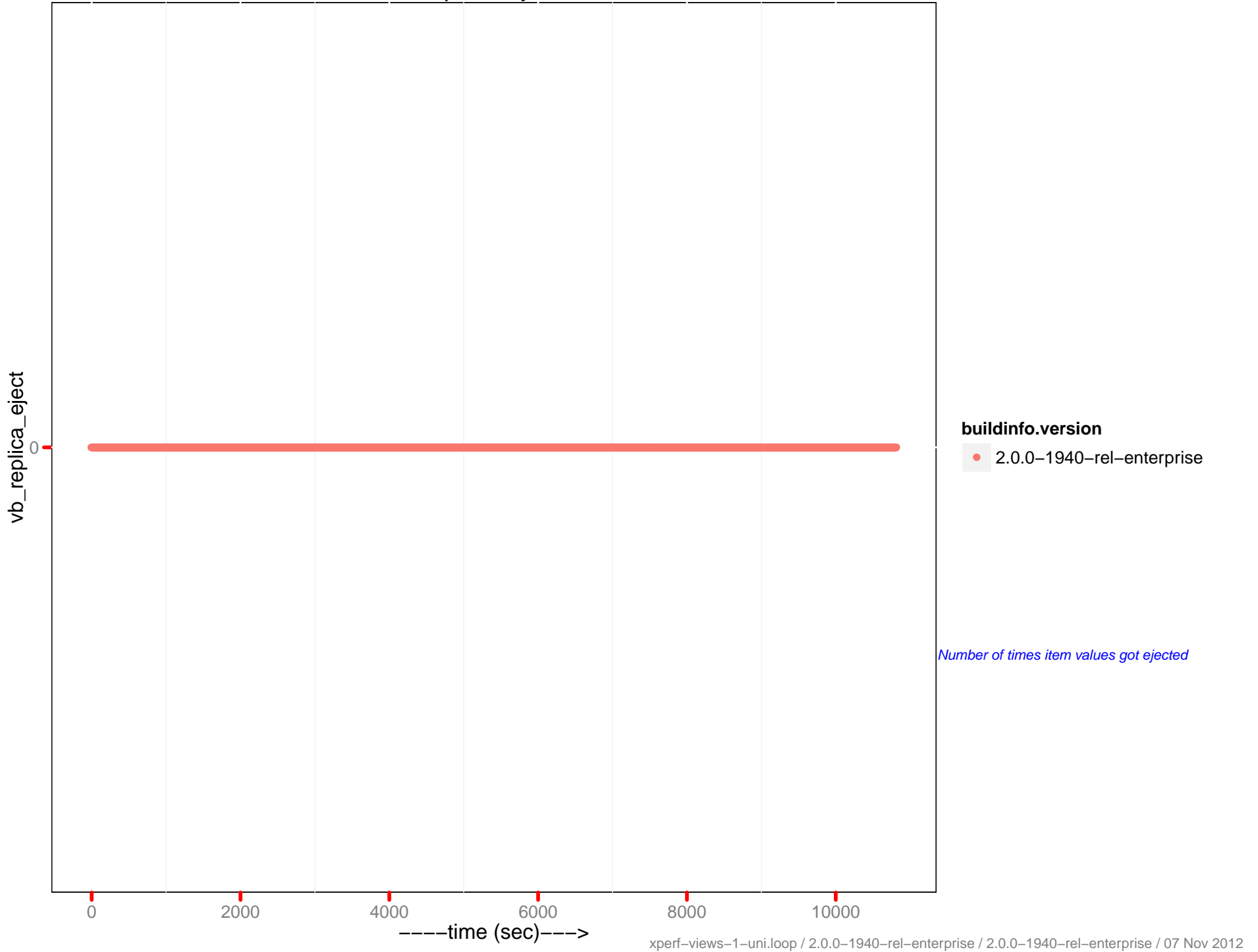


buildinfo.version

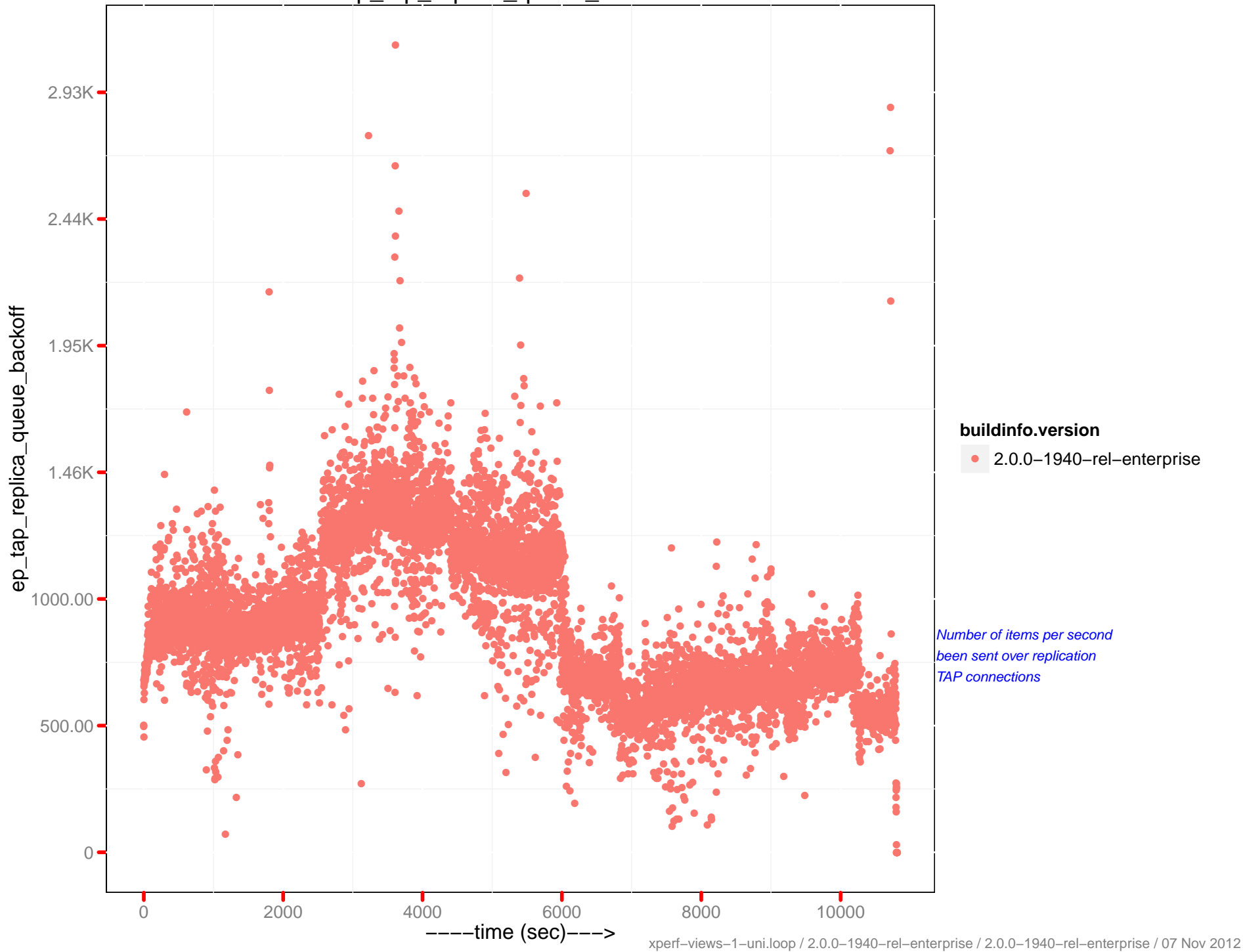
- 2.0.0-1940-rel-enterprise

Number of times item values got ejected

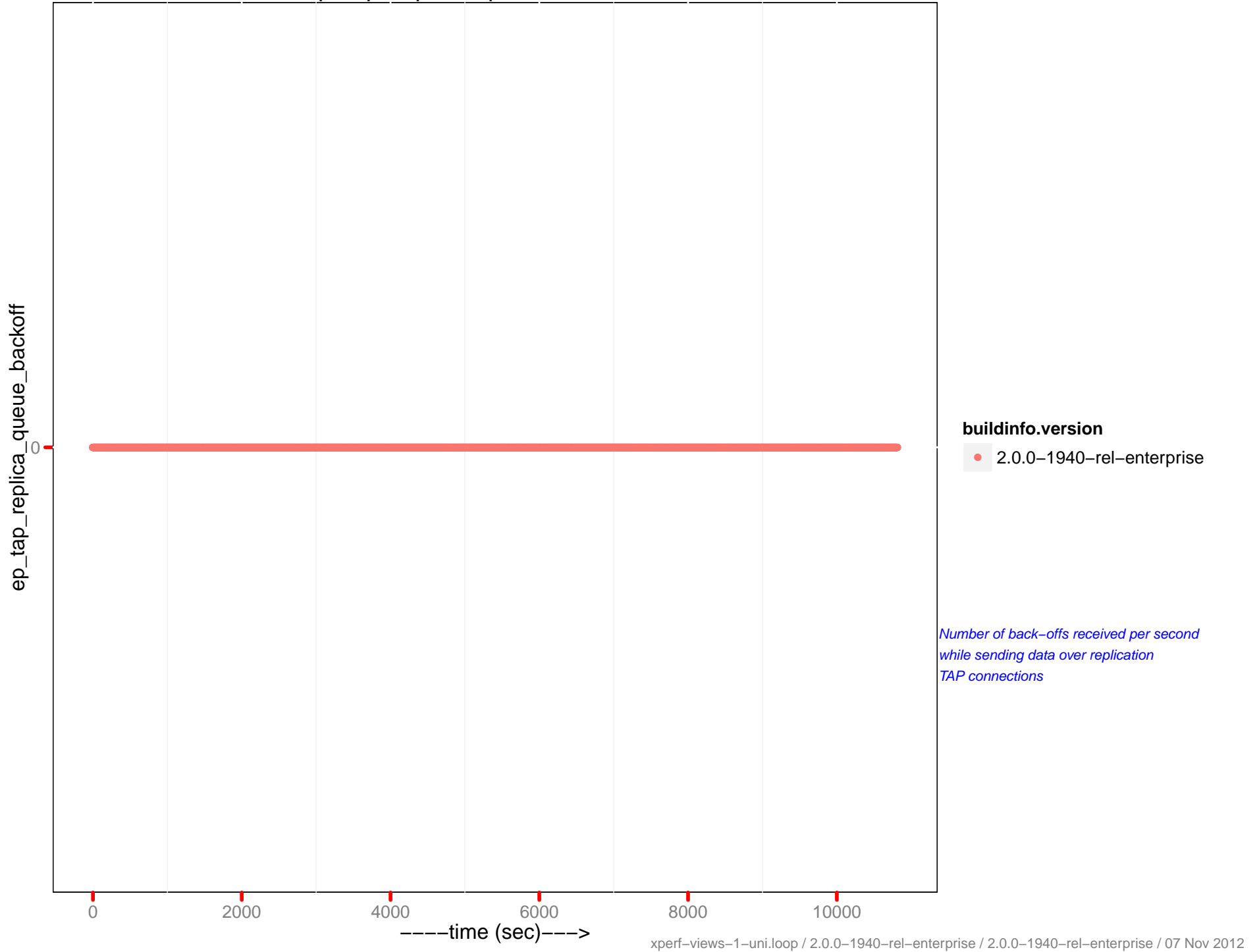
vb_replica_eject/sec



ep_tap_replica_queue_drain/sec



ep_tap_replica_queue_backoff/sec

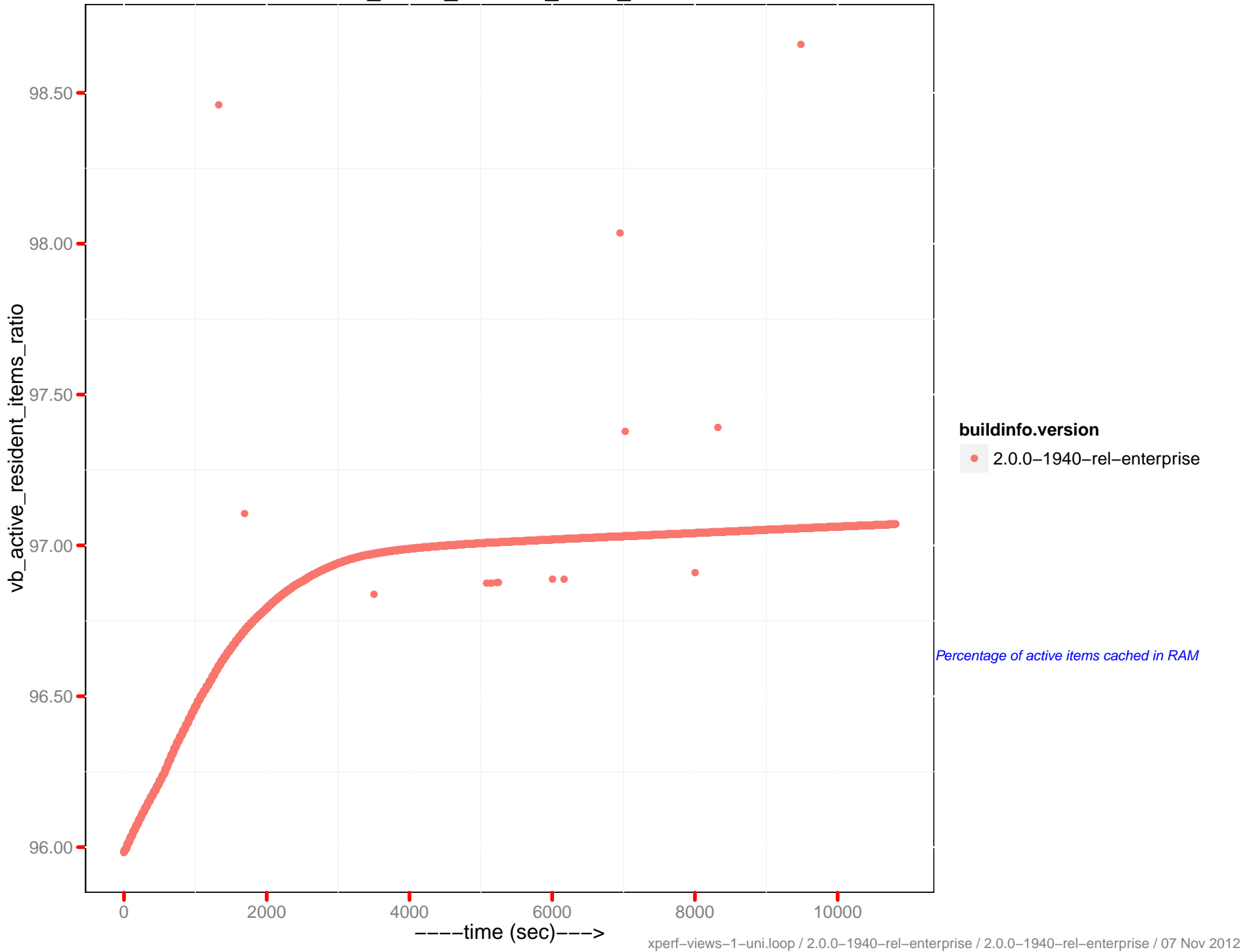


buildinfo.version

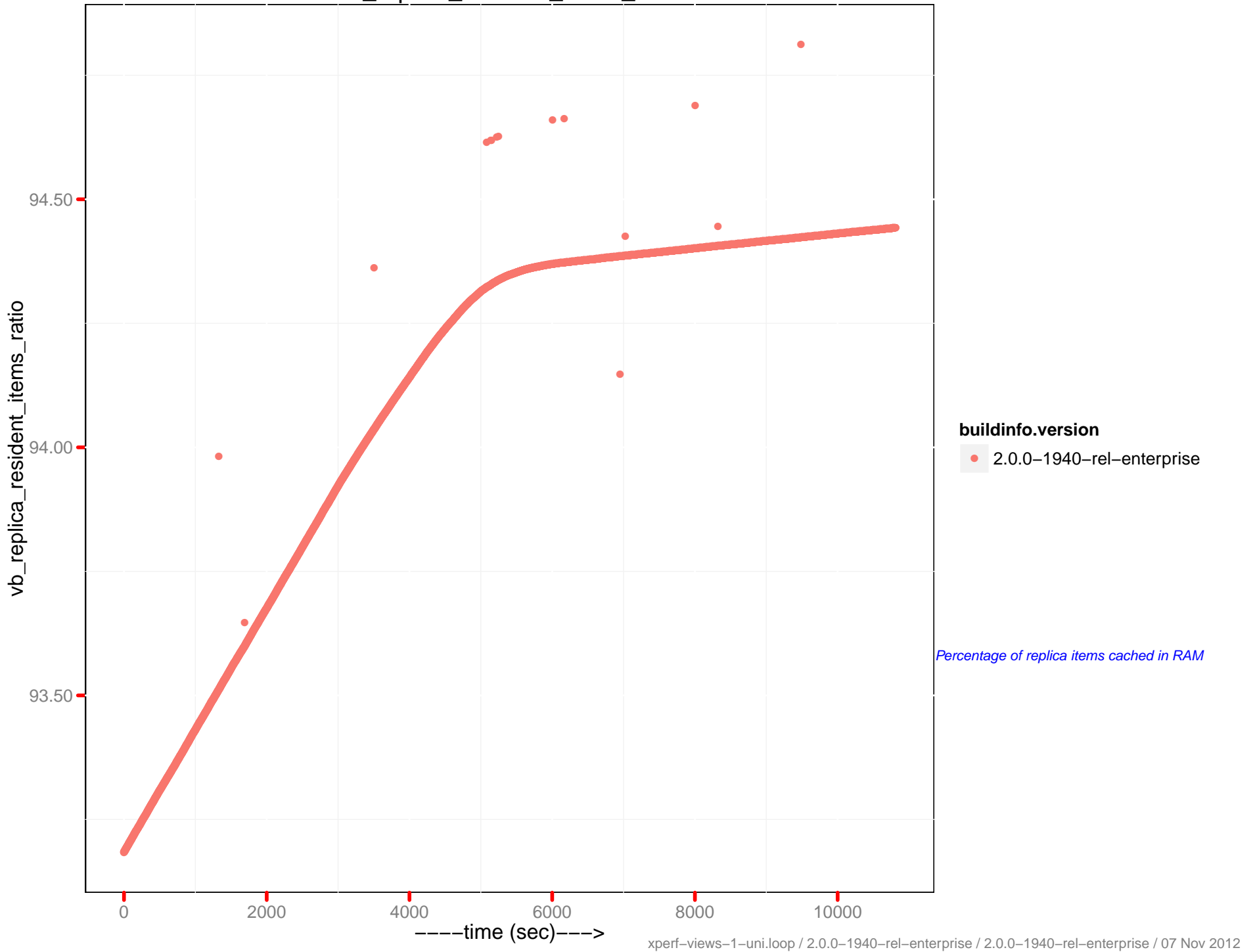
- 2.0.0-1940-rel-enterprise

*Number of back-offs received per second
while sending data over replication
TAP connections*

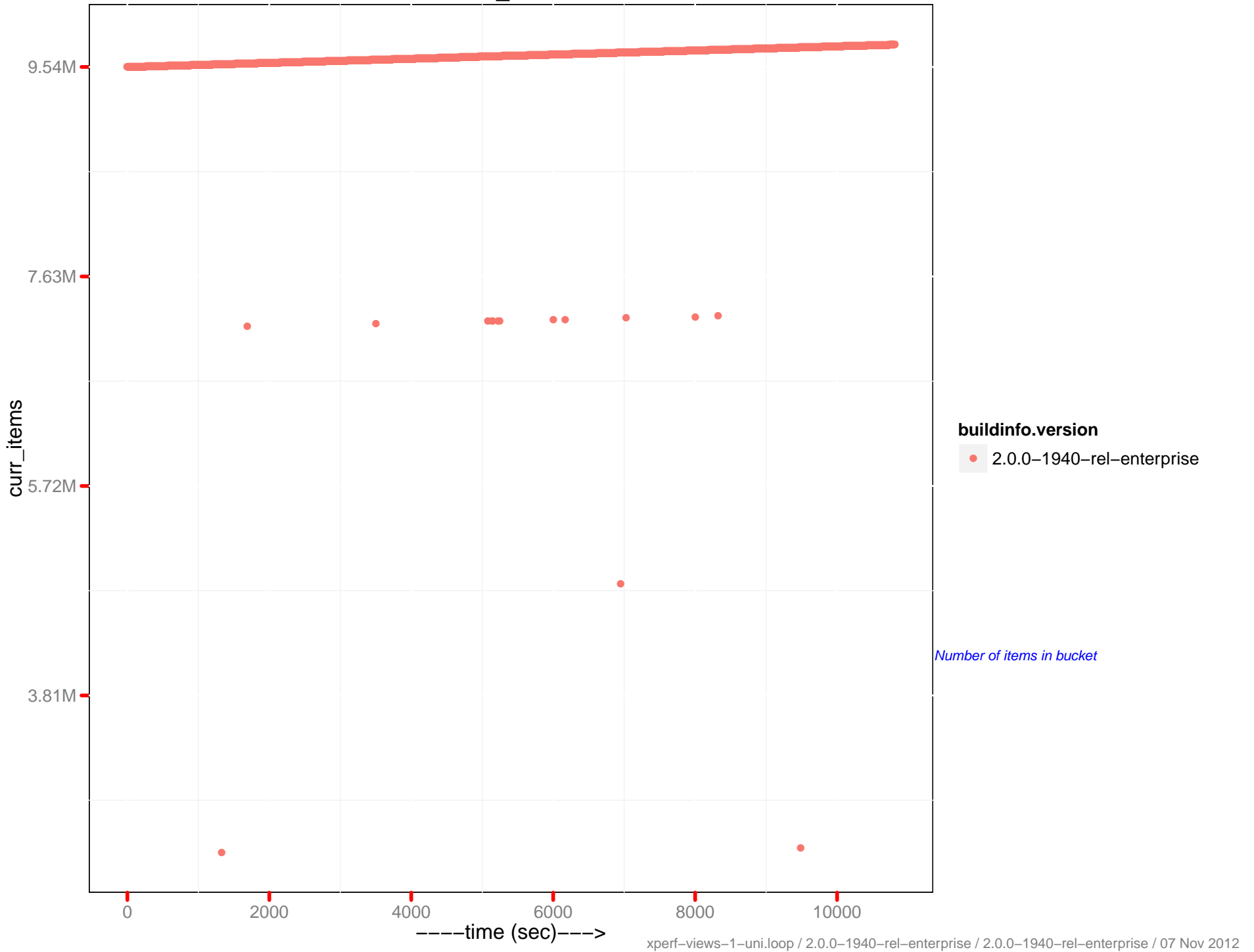
vb_active_resident_items_ratio



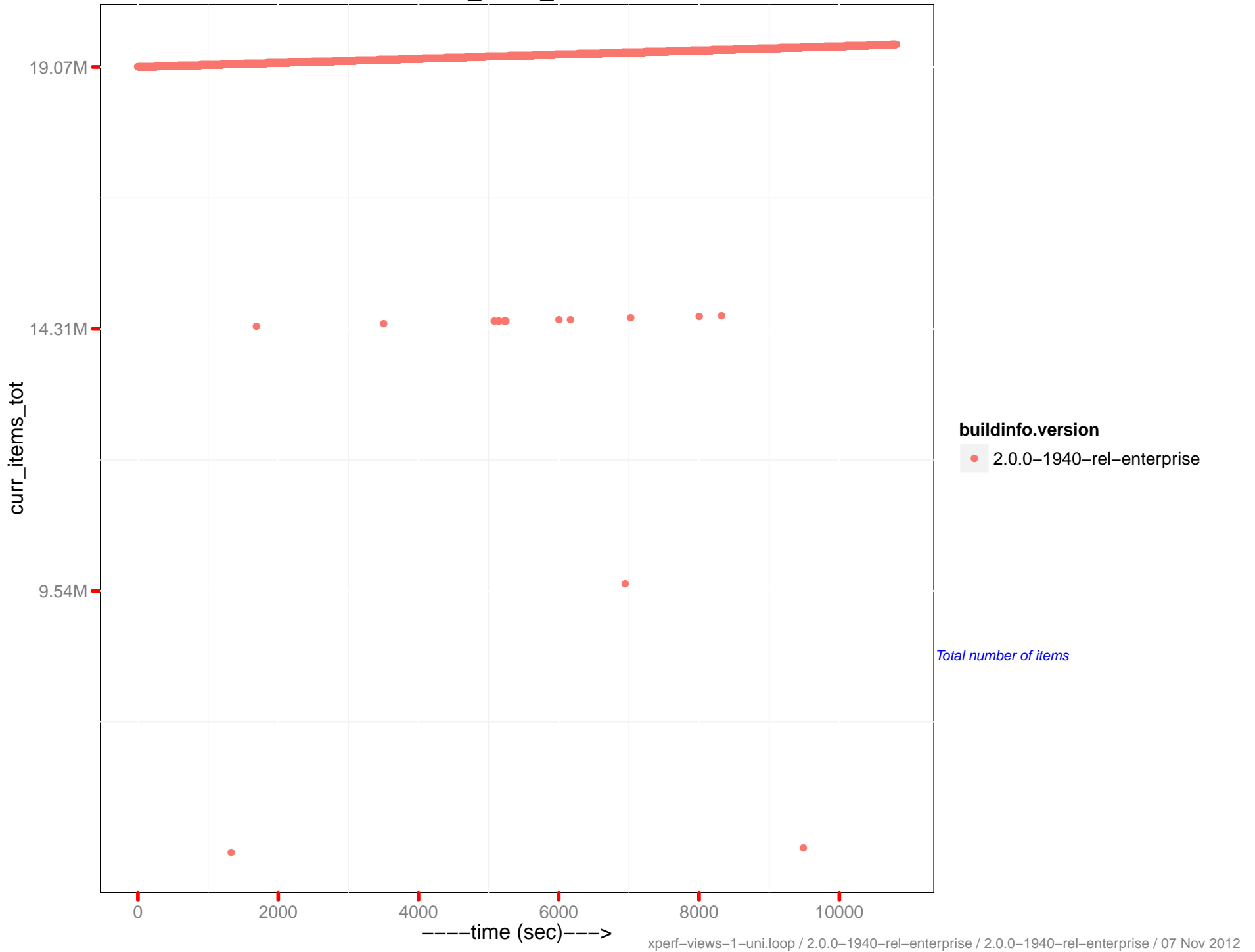
vb_replica_resident_items_ratio



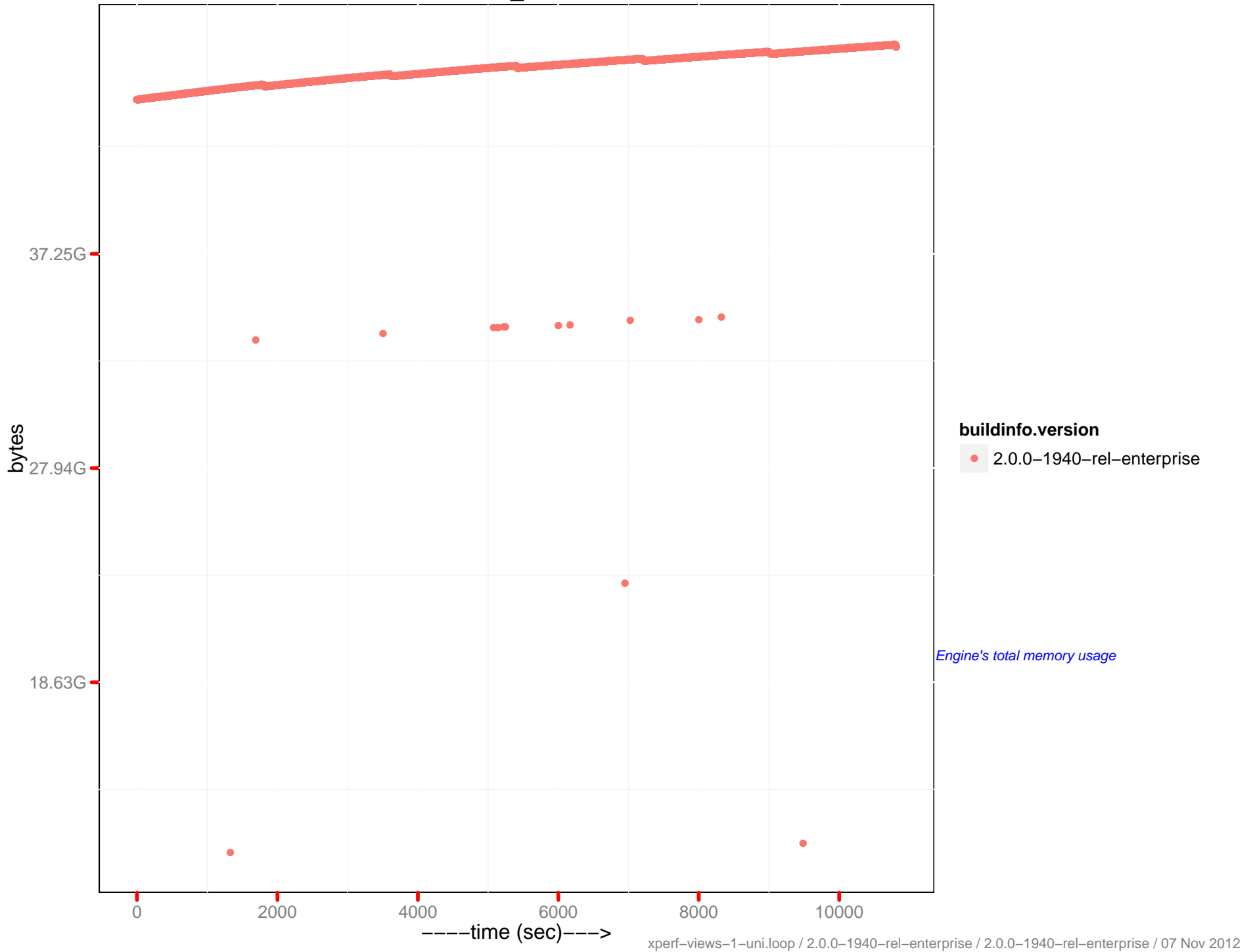
curr_items



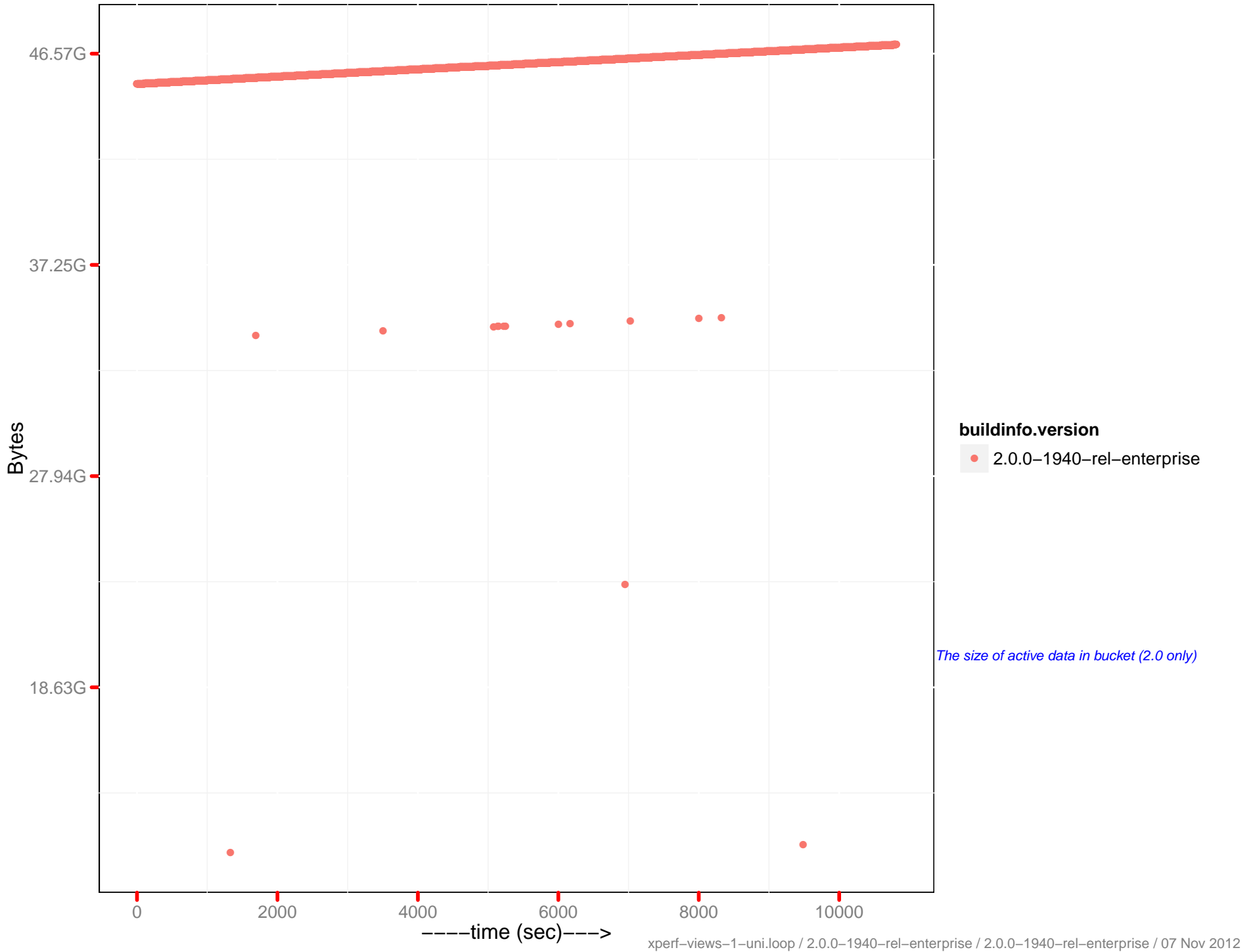
cur_items_total



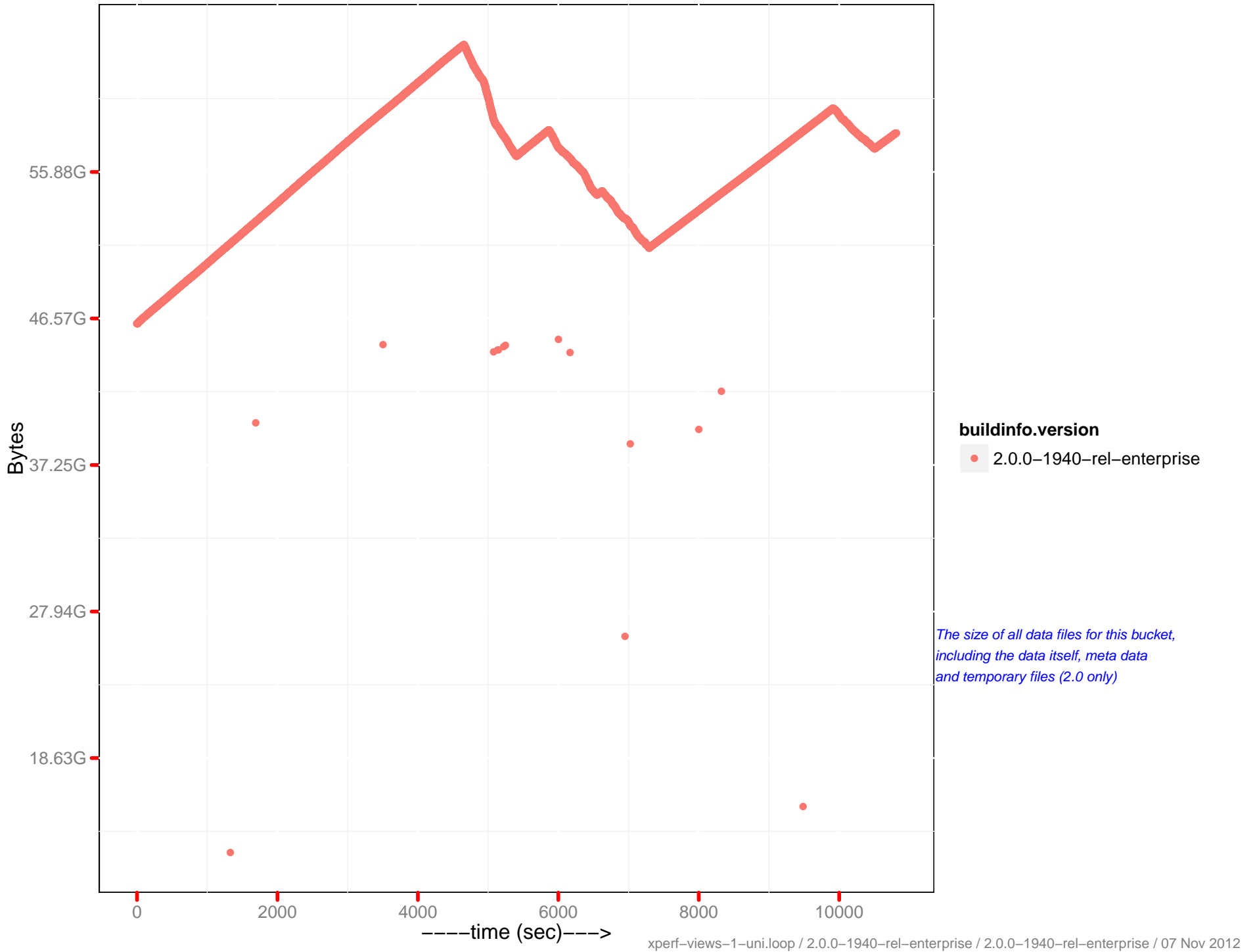
mem_used



Docs data size



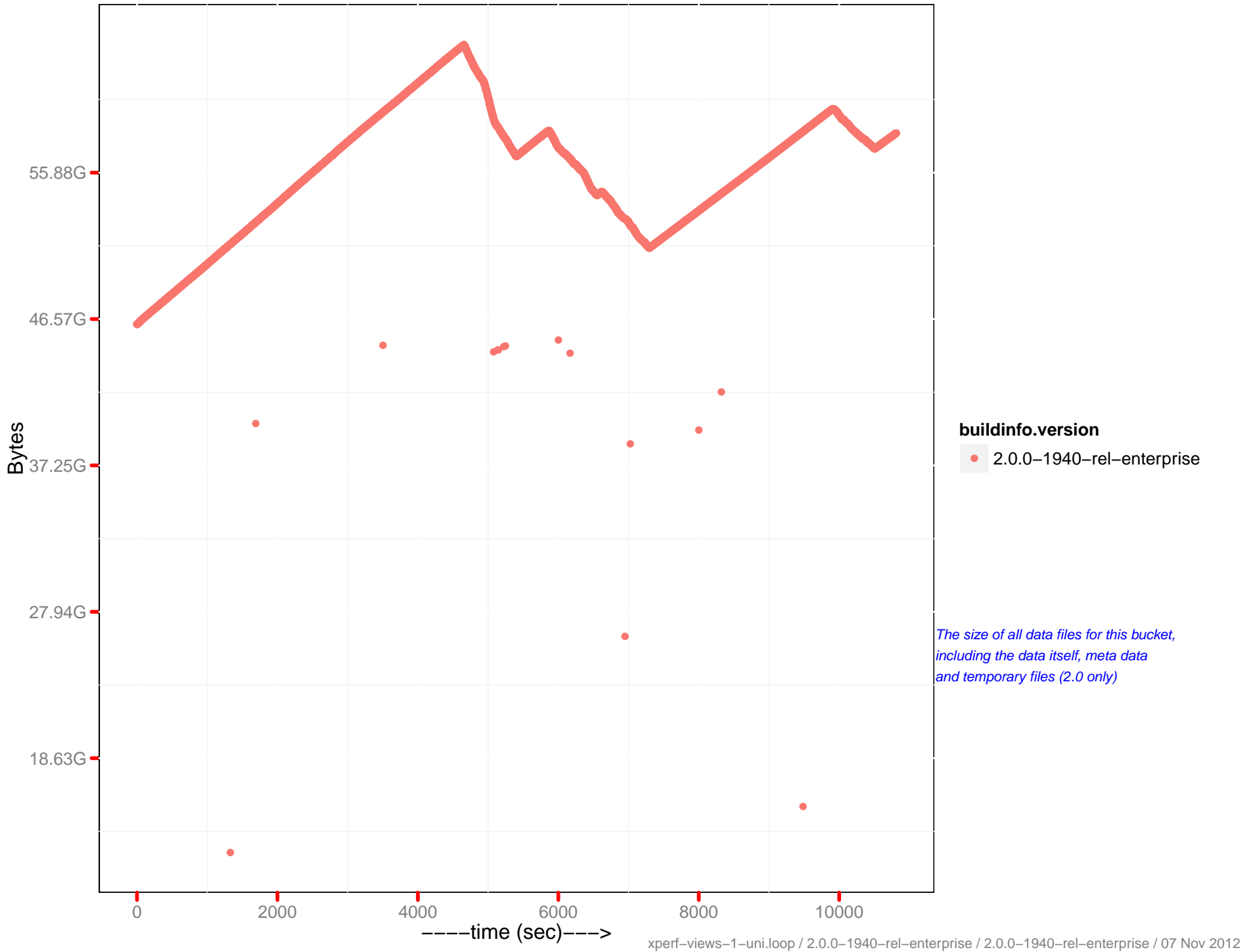
Docs disk size



buildinfo.version
● 2.0.0-1940-rel-enterprise

The size of all data files for this bucket, including the data itself, meta data and temporary files (2.0 only)

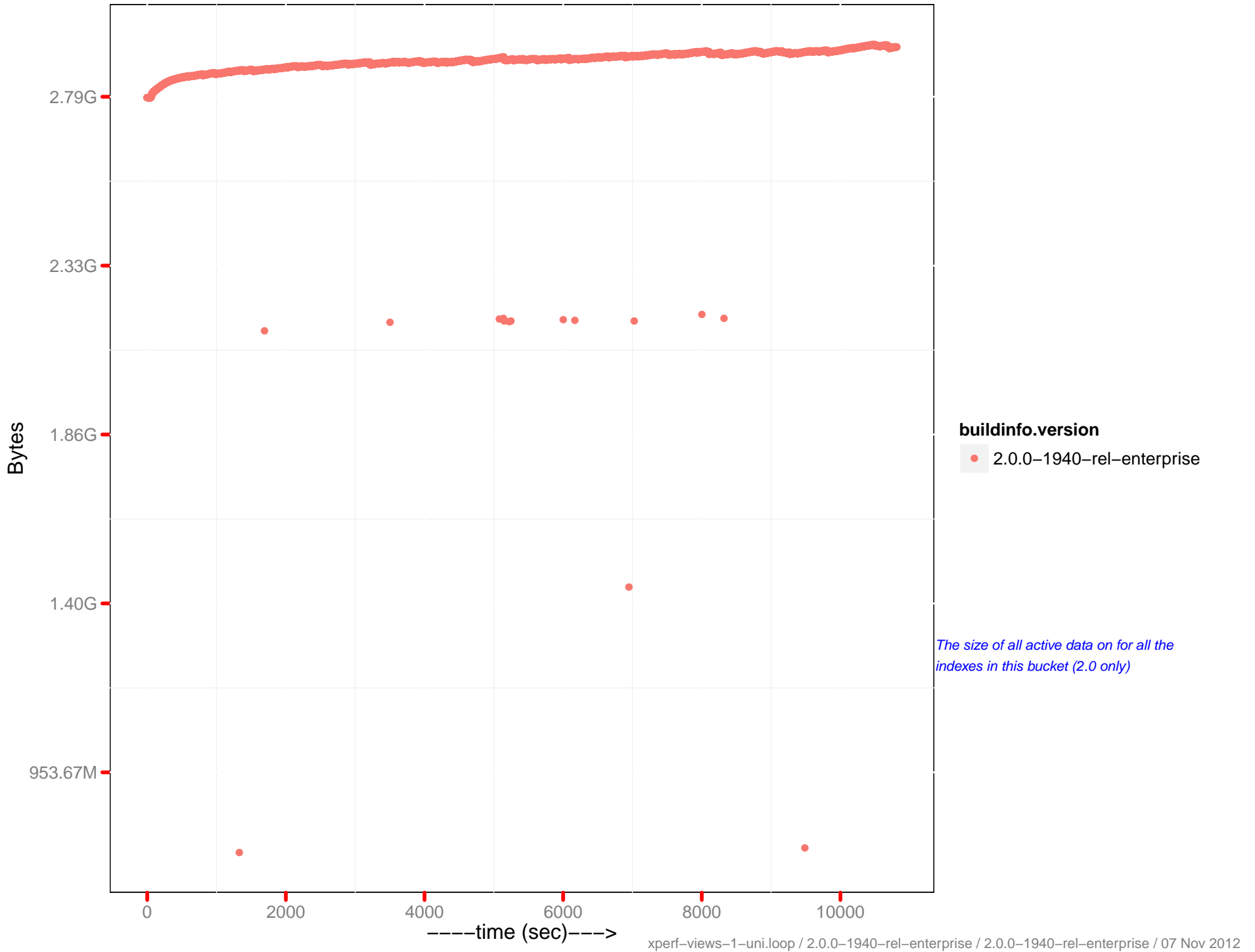
Docs actual disk size



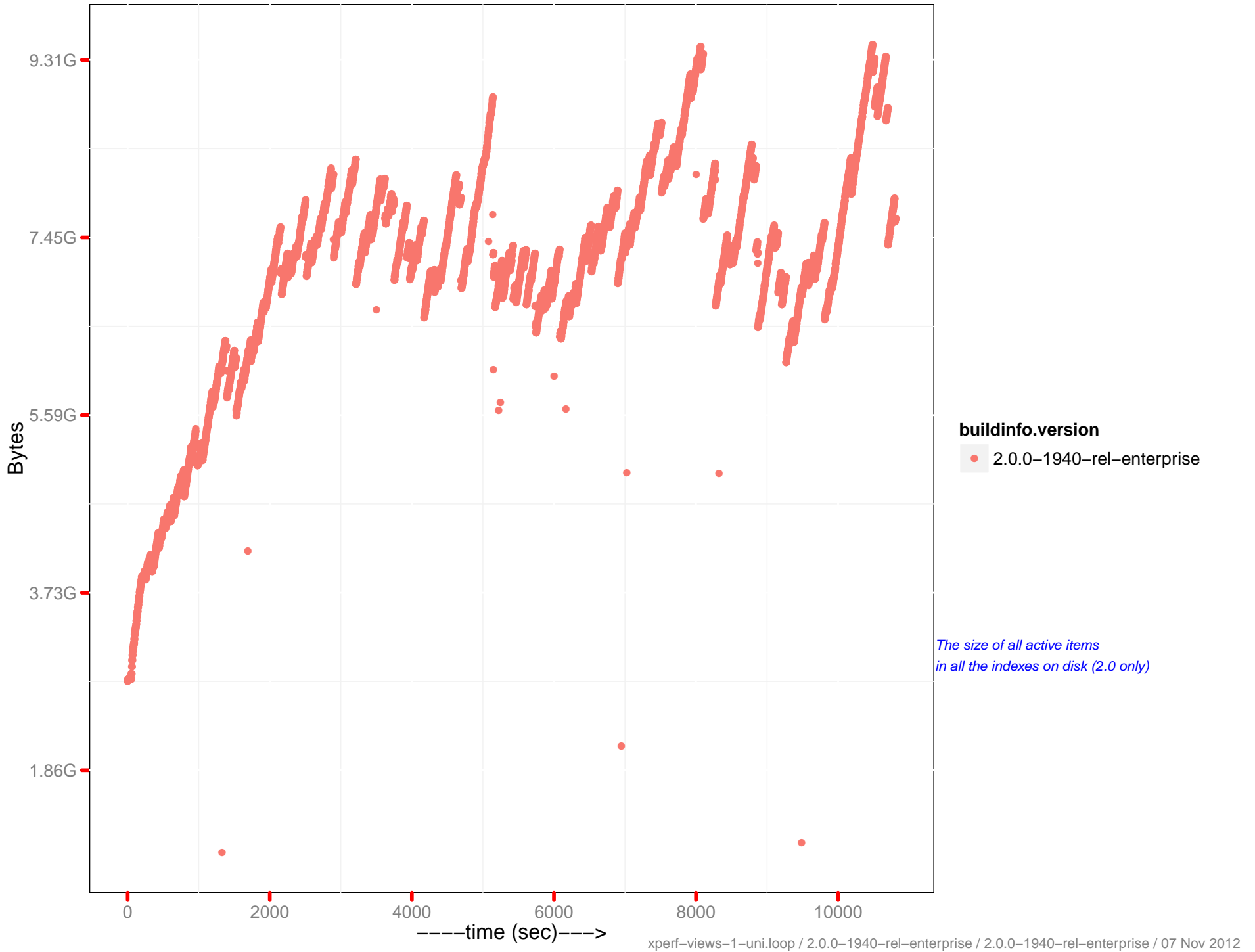
buildinfo.version
● 2.0.0-1940-rel-enterprise

The size of all data files for this bucket, including the data itself, meta data and temporary files (2.0 only)

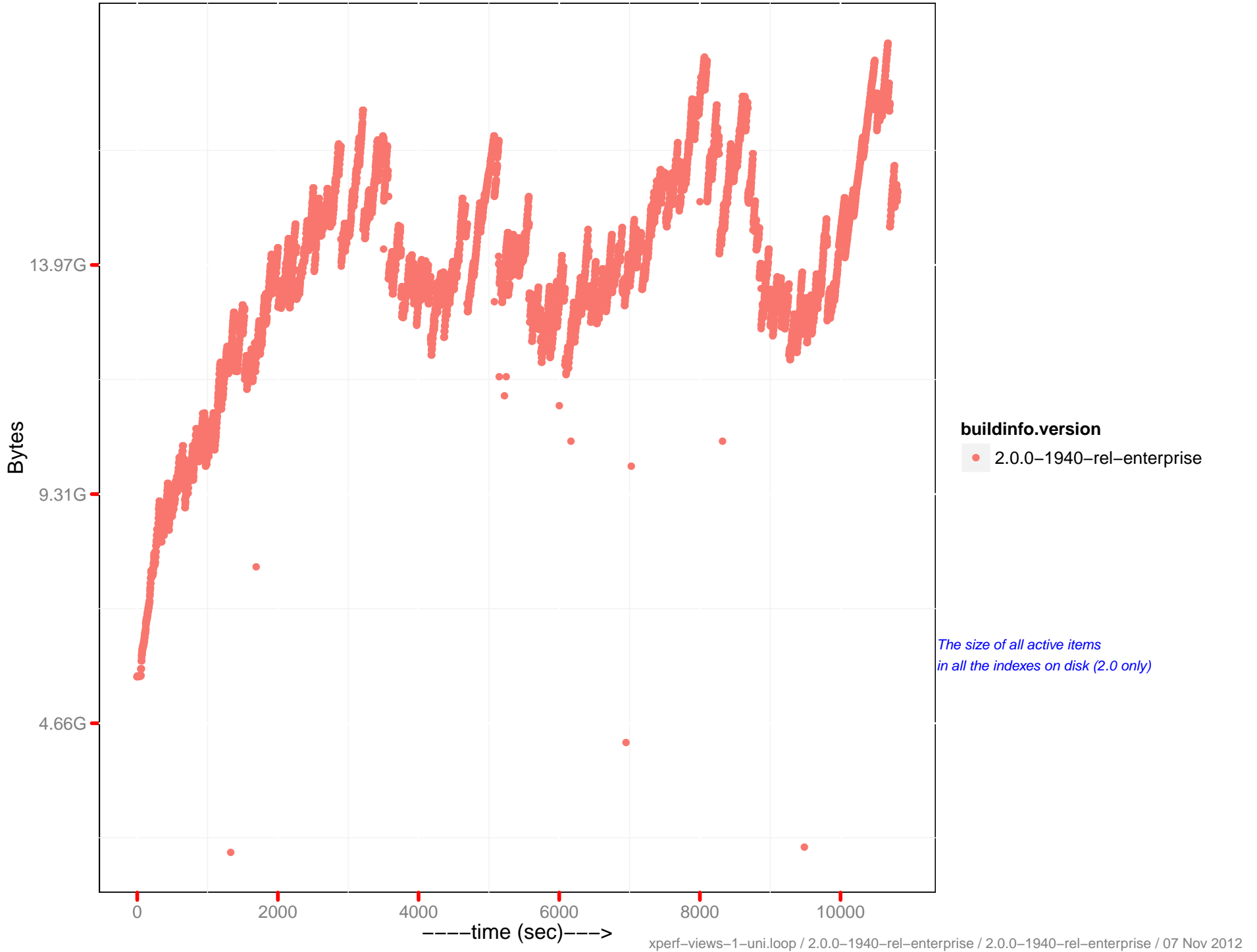
Views data size



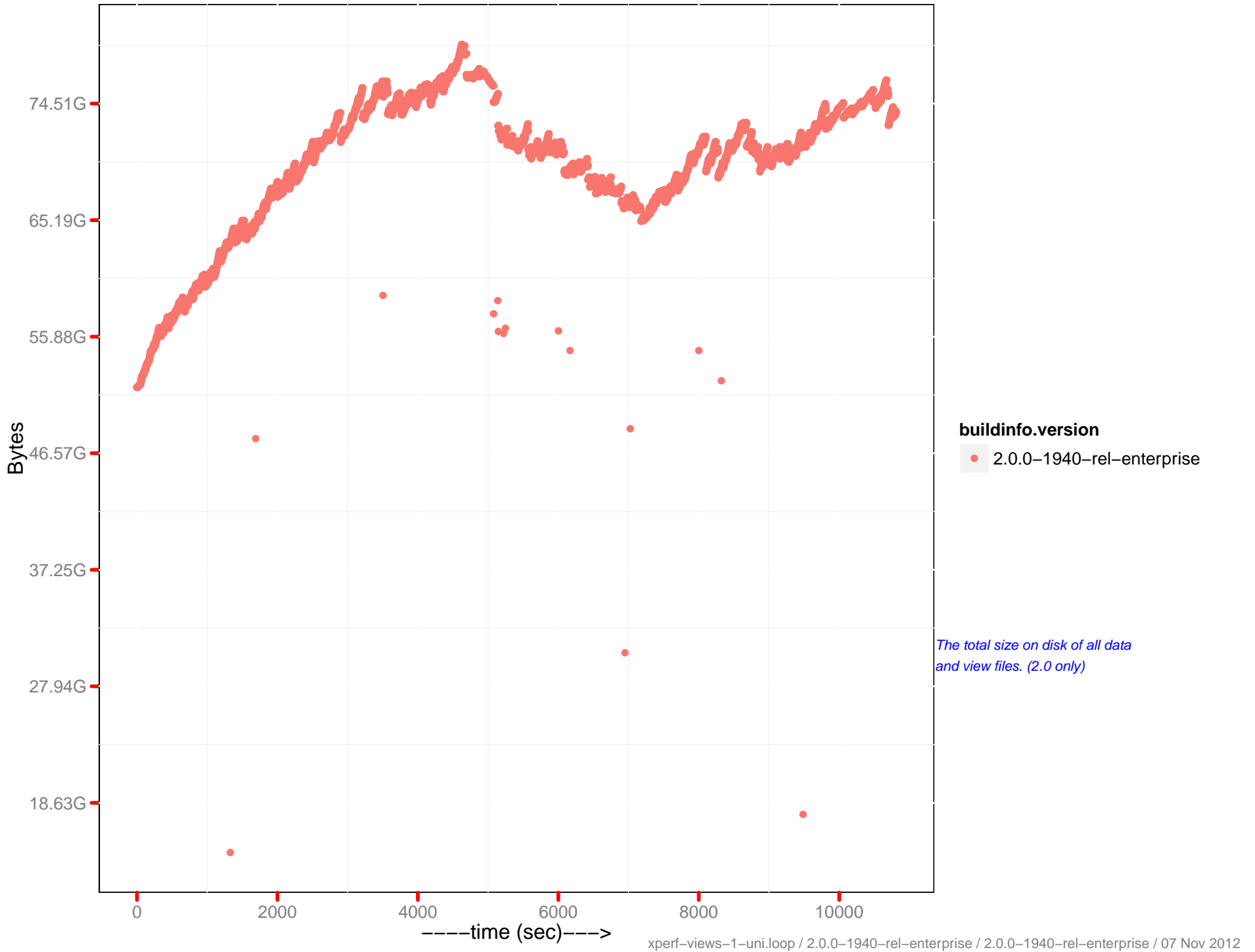
Views disk size



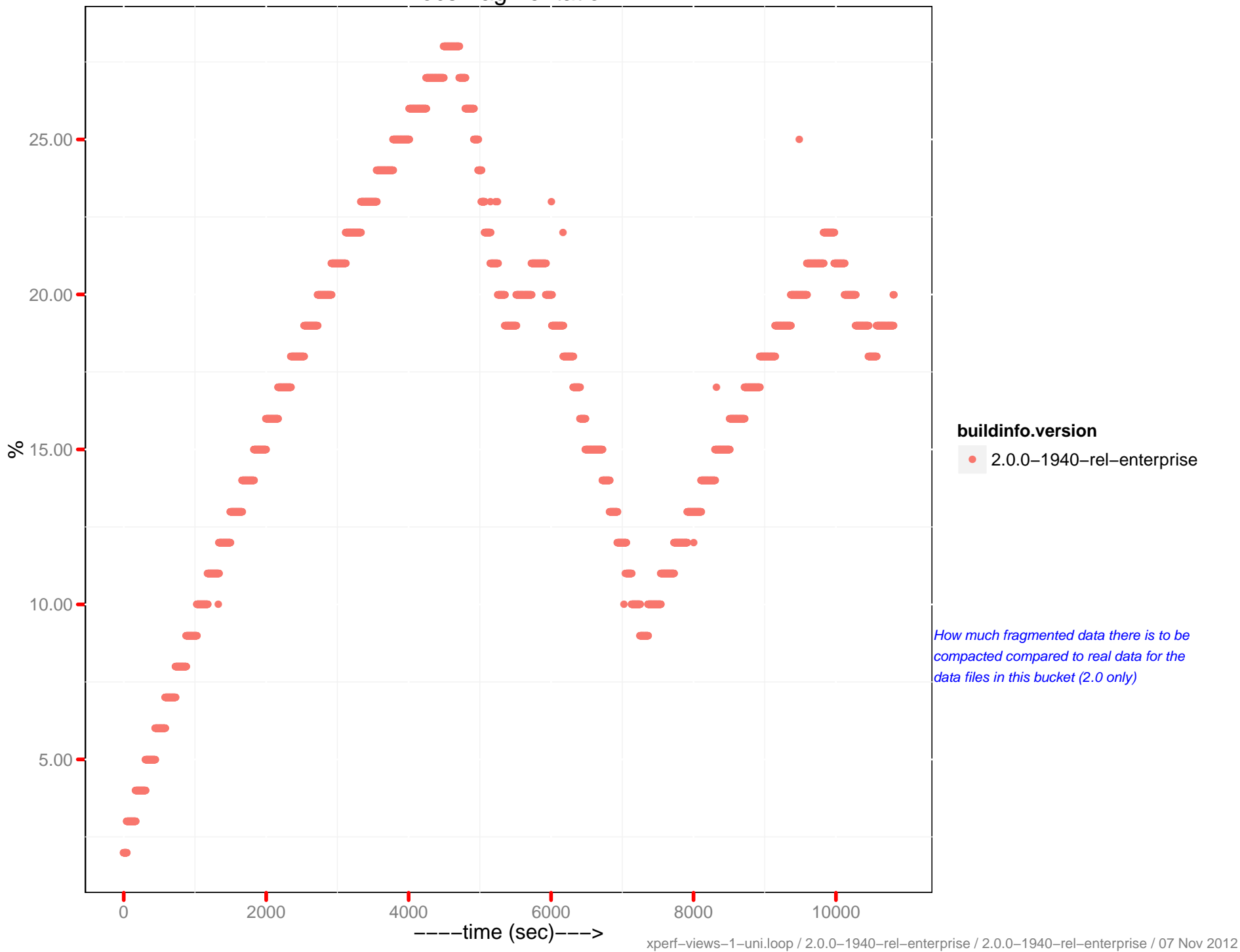
Views actual disk size



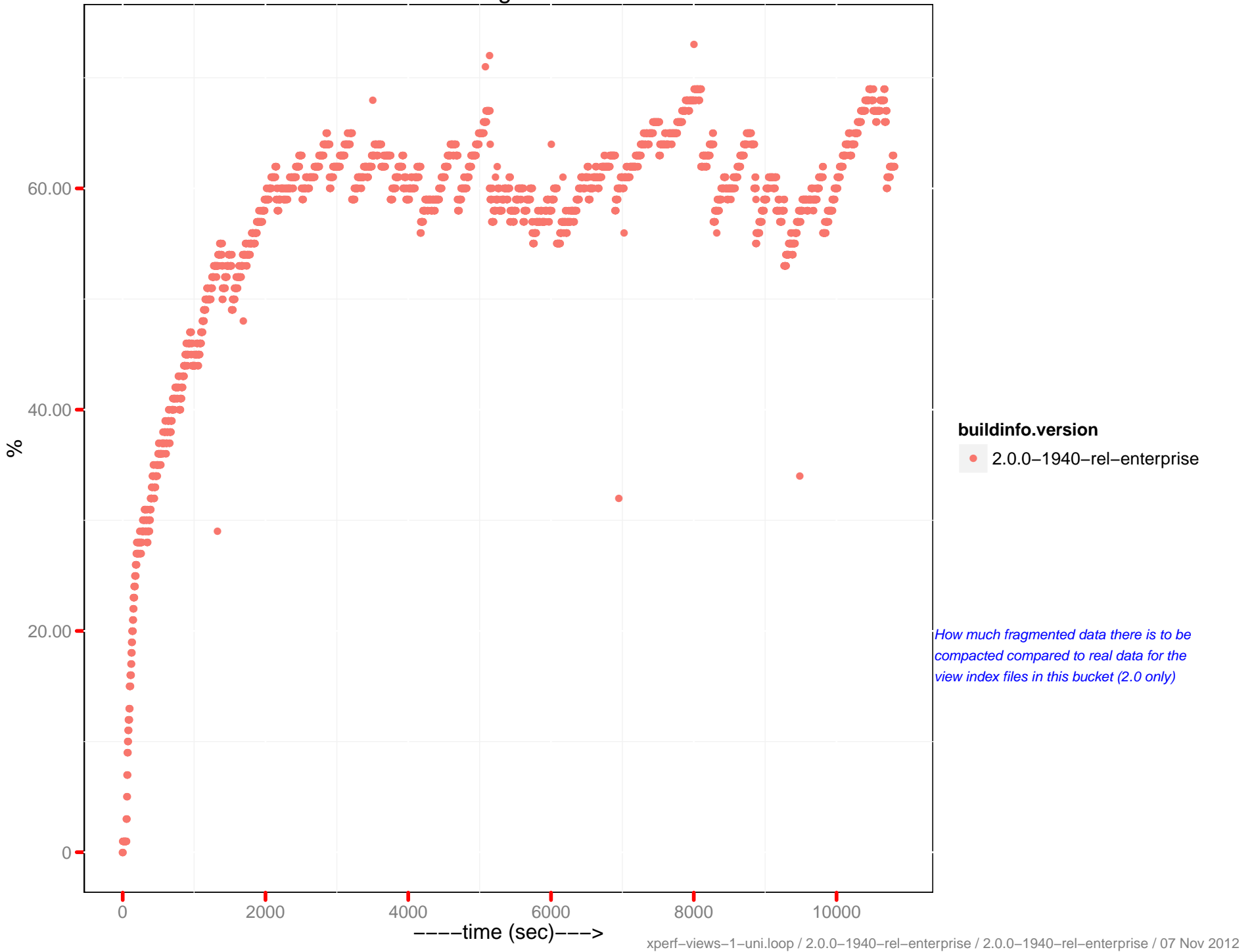
Total disk size



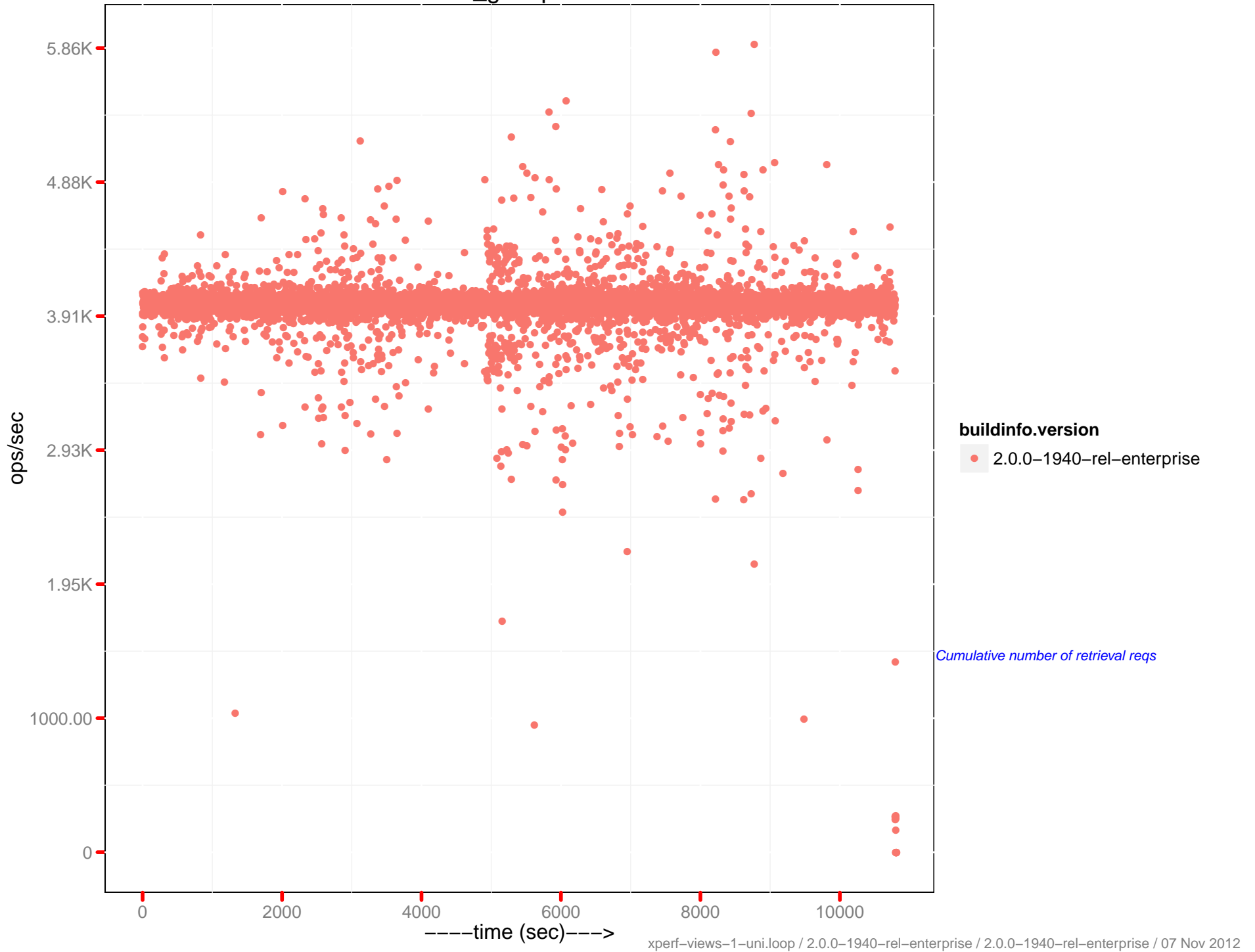
Docs fragmentation



Views fragmentation



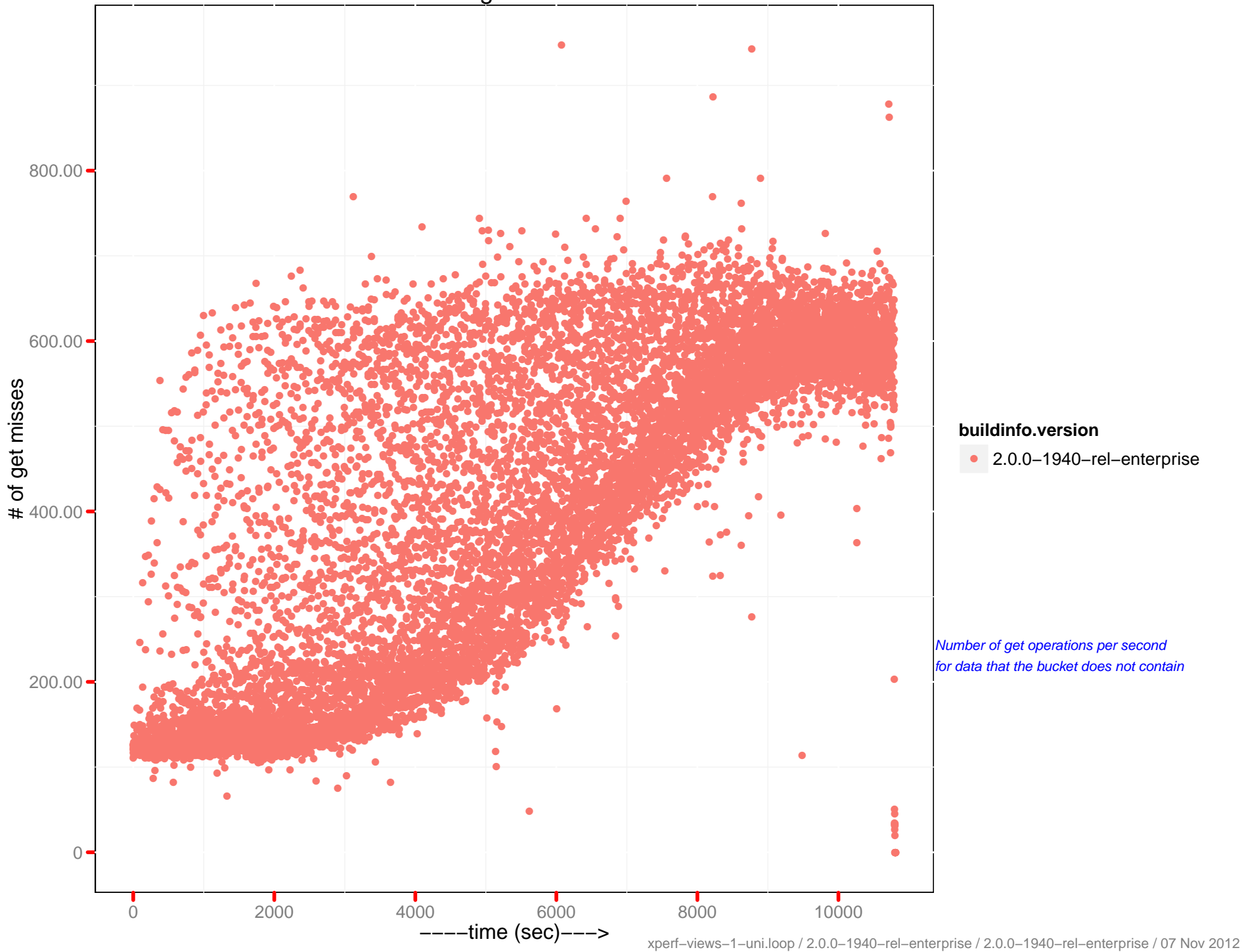
cmd_get ops/sec



cmd_set ops/sec



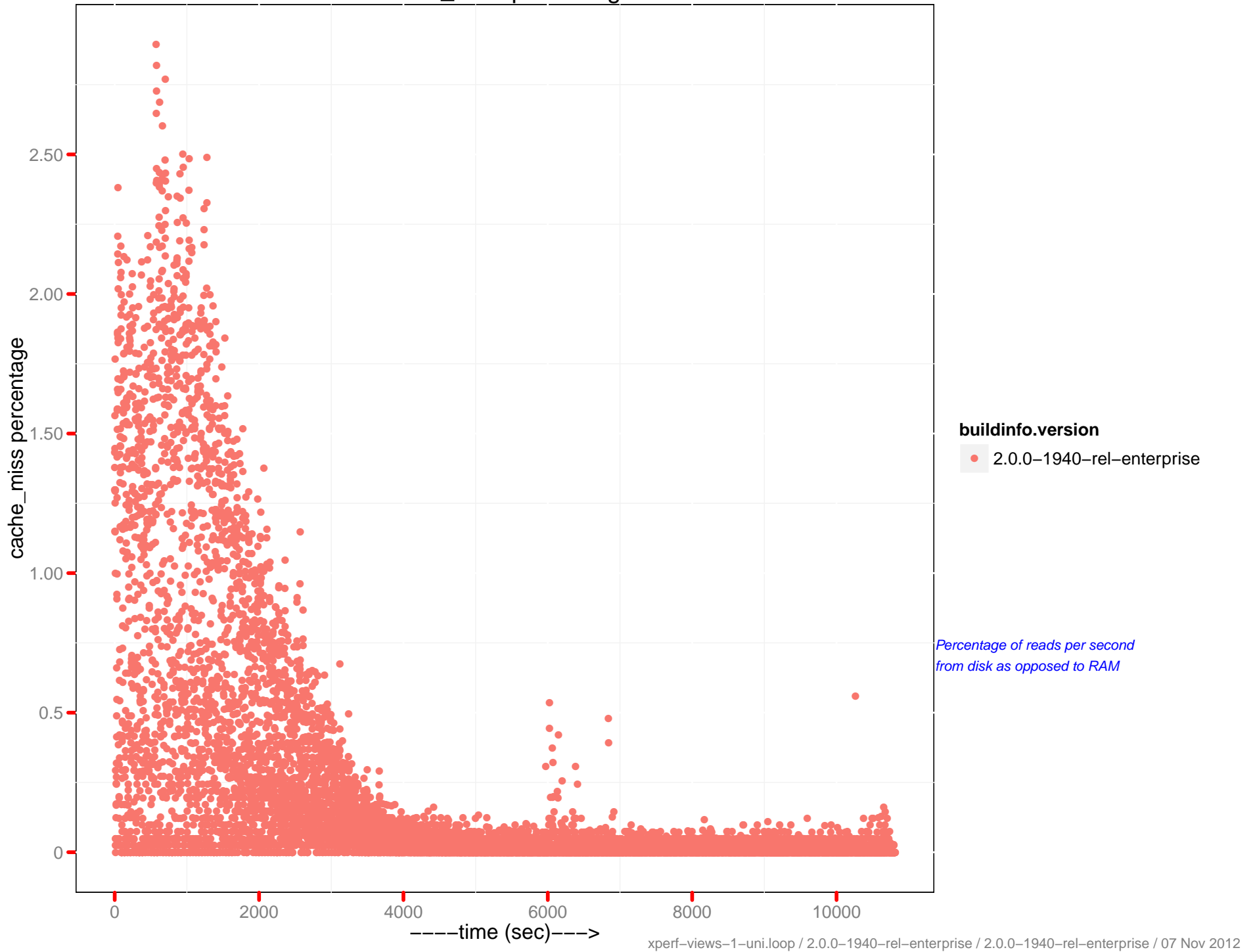
of get misses



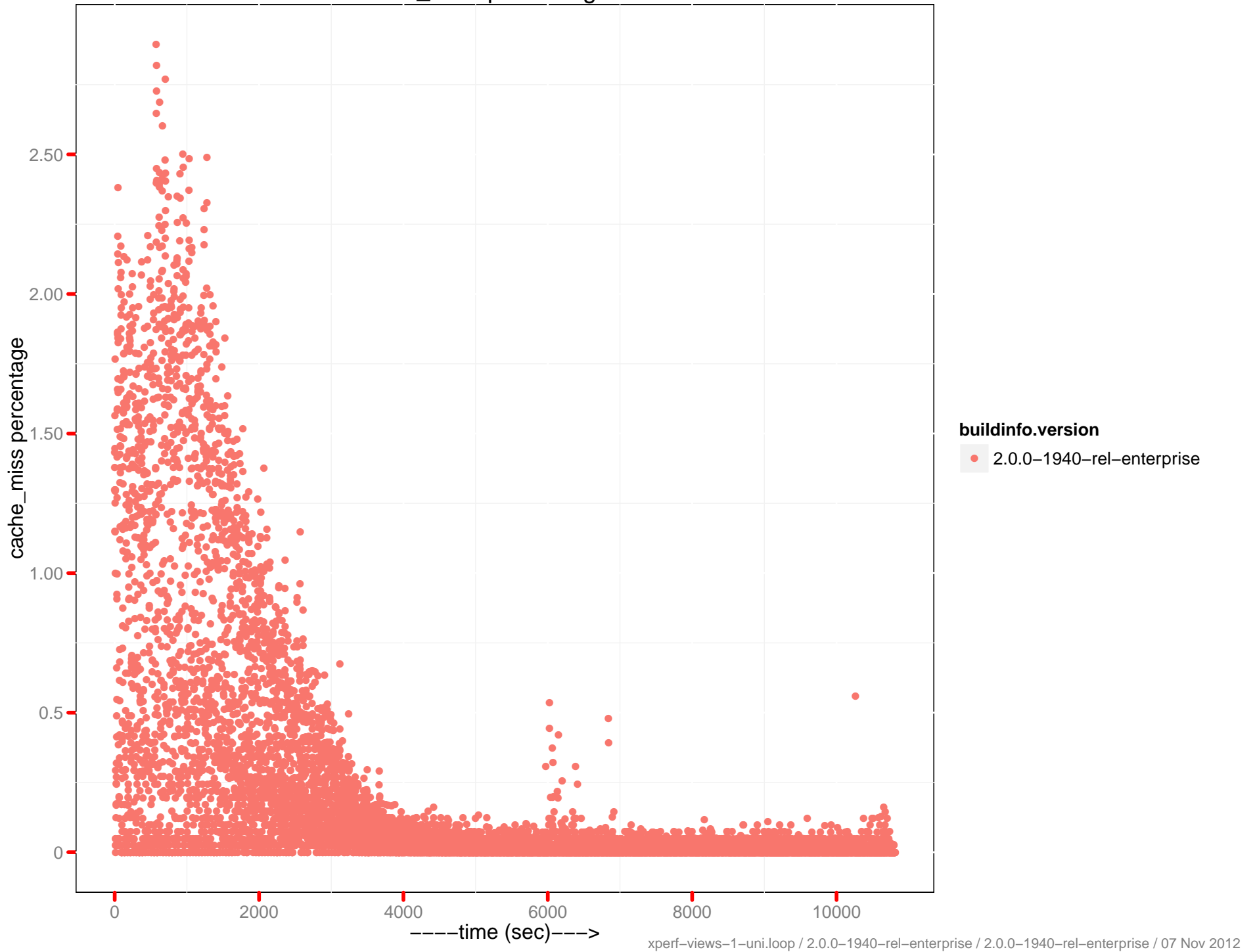
of get hits



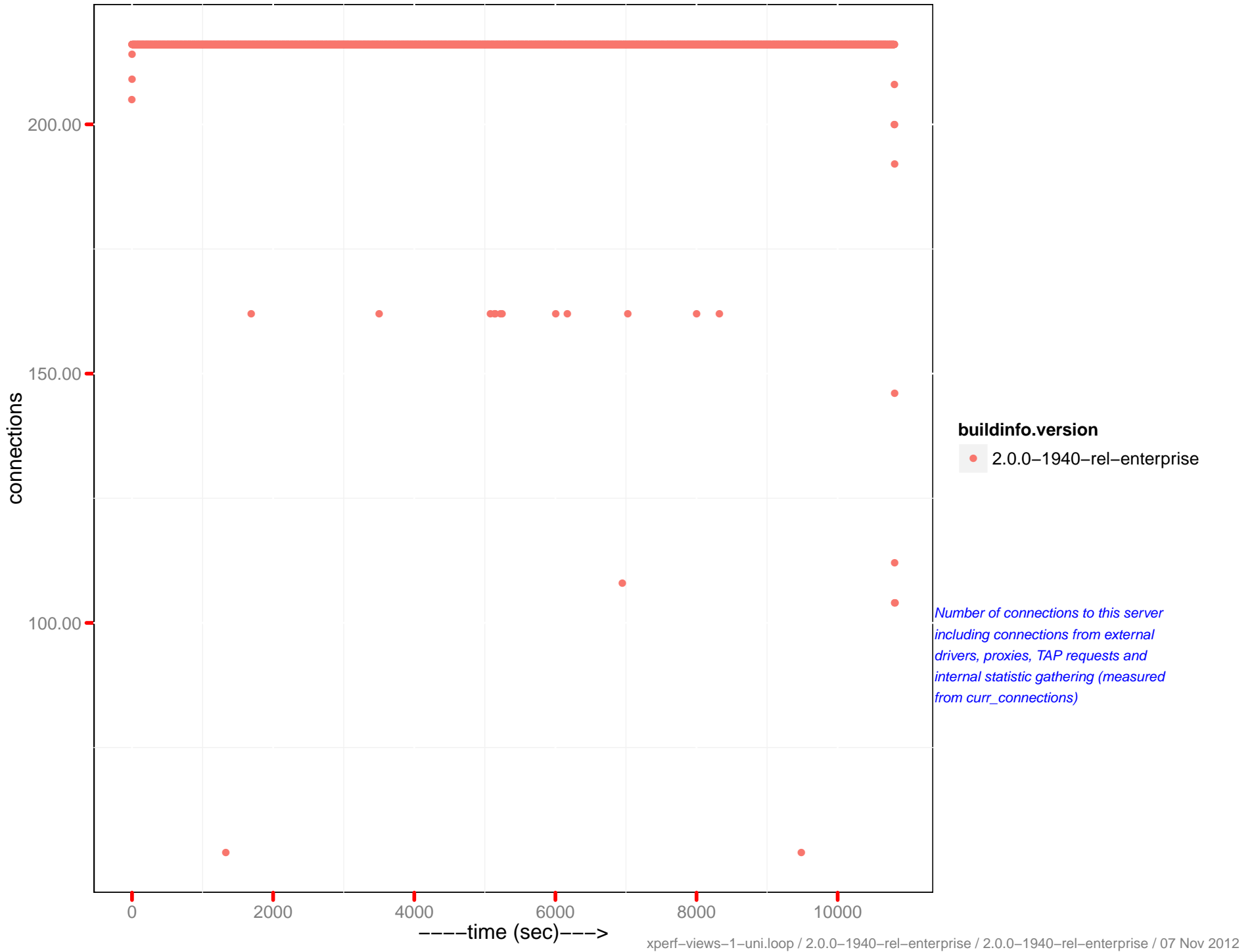
cache_miss percentage



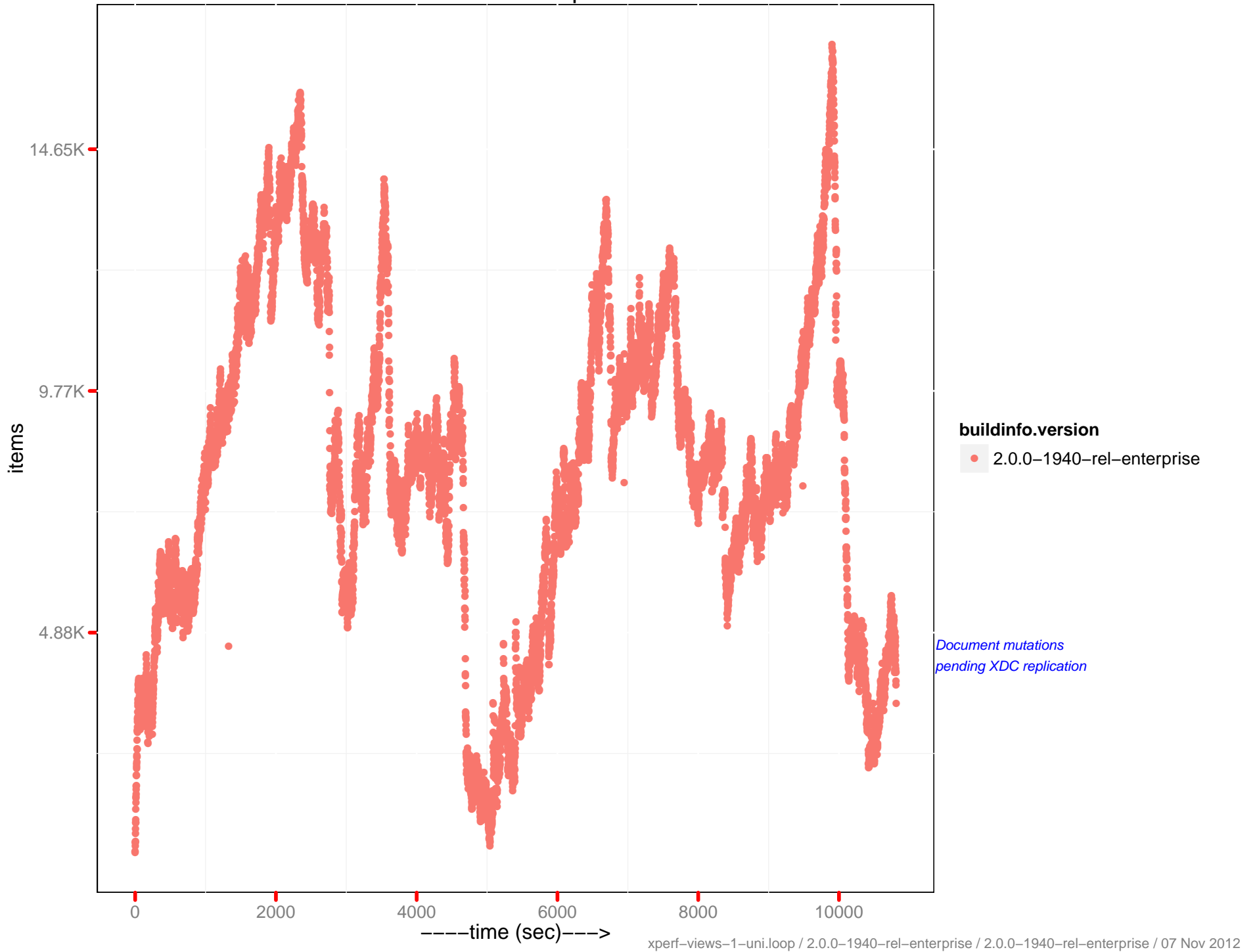
cache_miss percentage 0-5



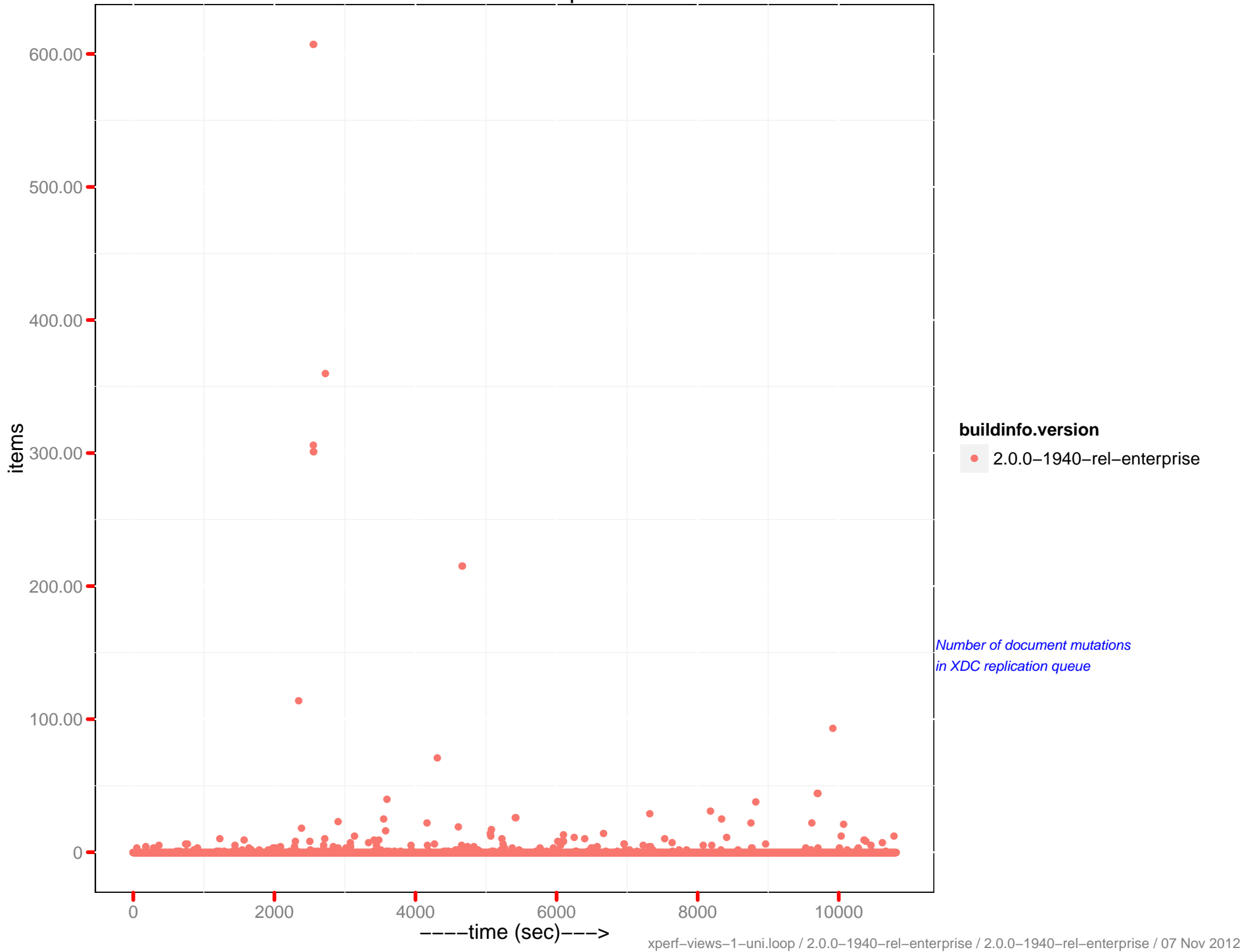
Number of connections



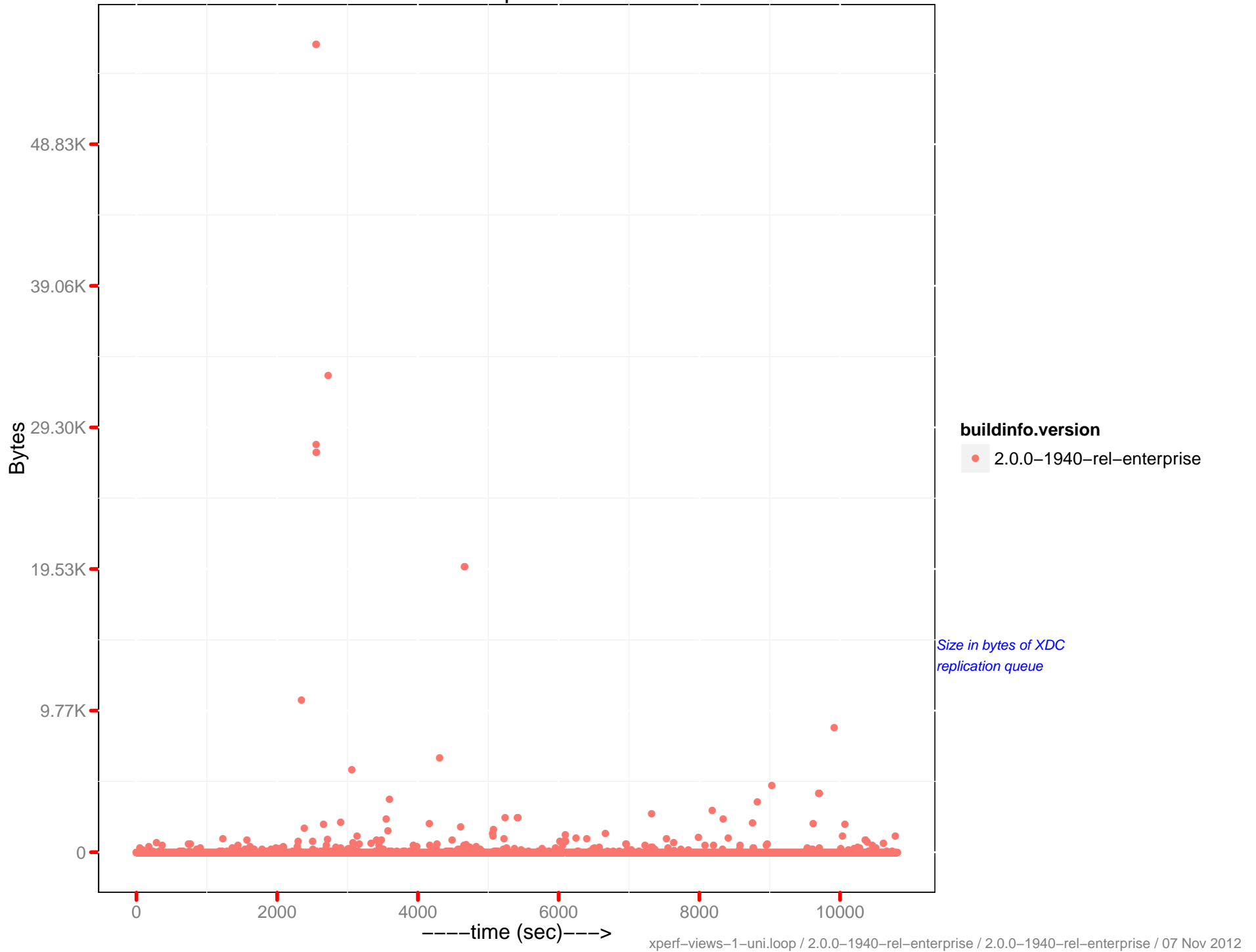
XDCR docs to replicate



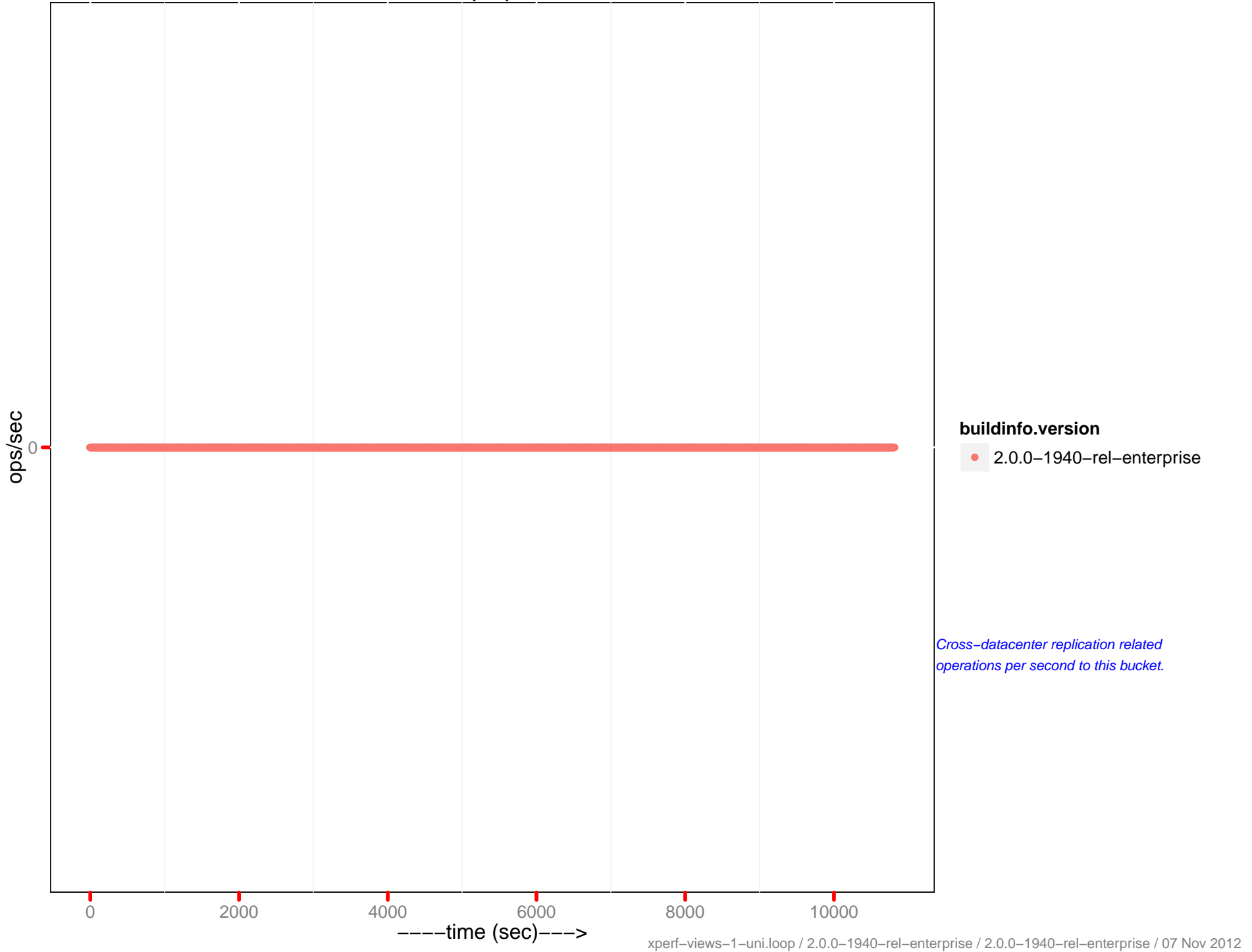
XDCR docs in queue



XDCR queue size



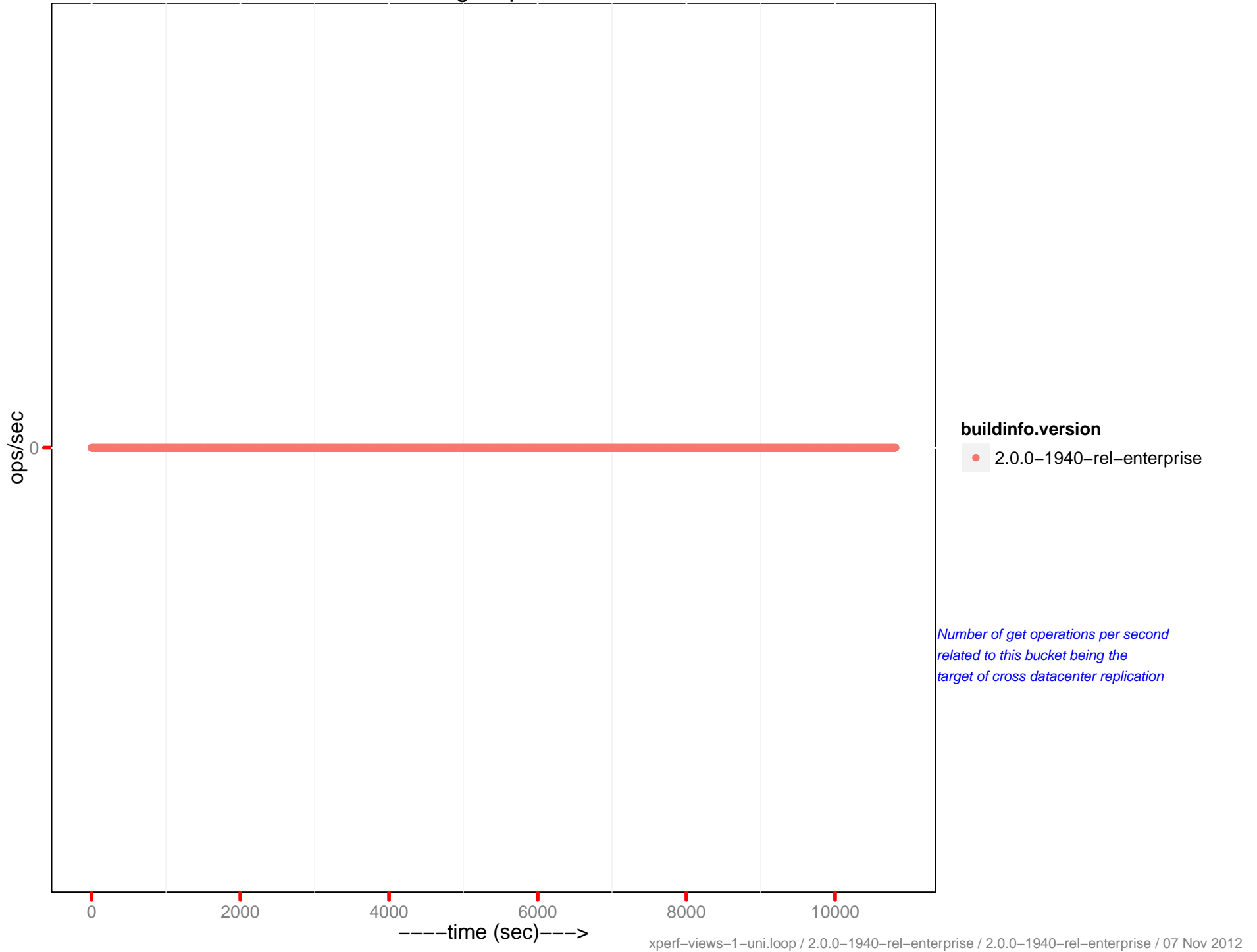
XDC ops per sec



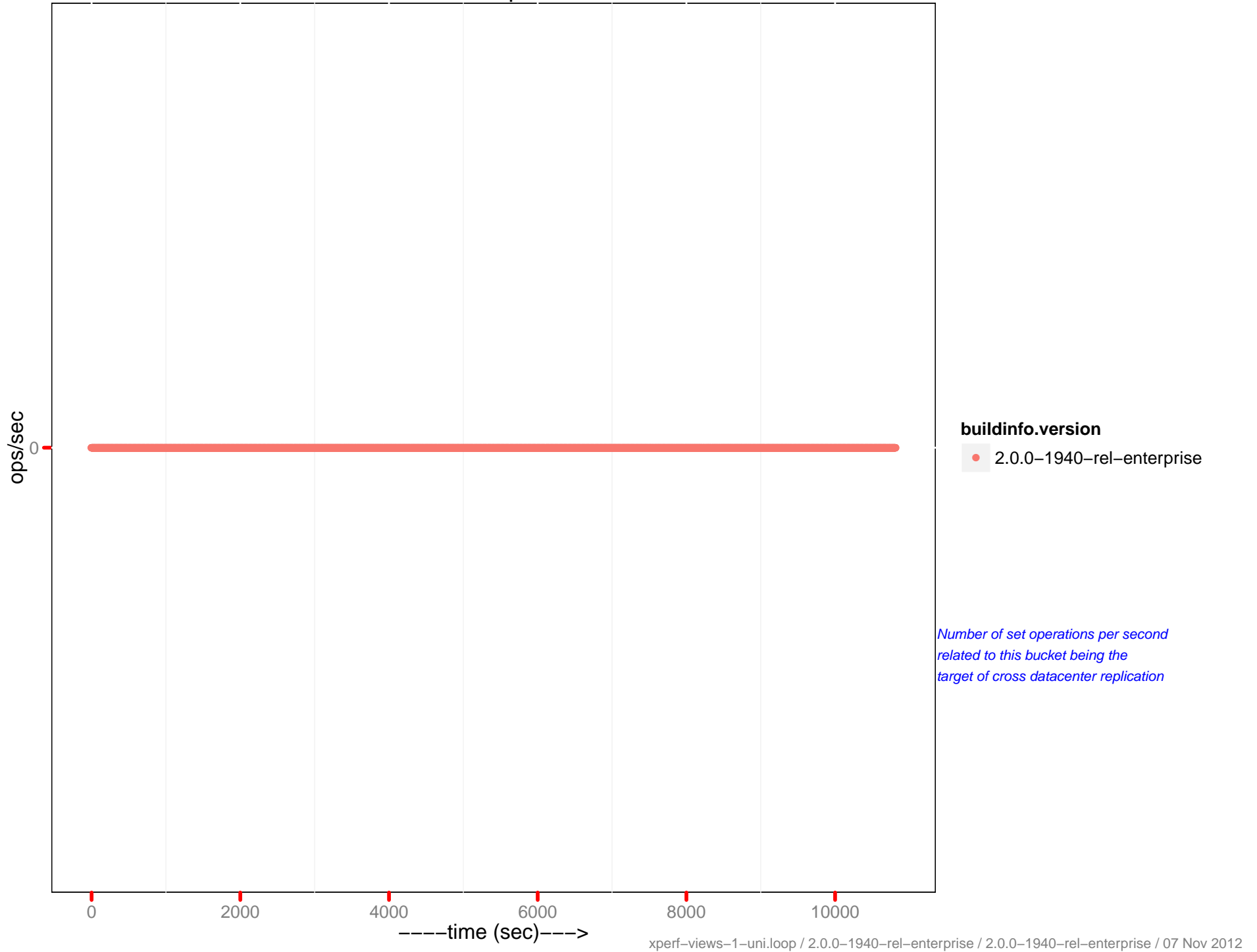
buildinfo.version
● 2.0.0-1940-rel-enterprise

Cross-datacenter replication related operations per second to this bucket.

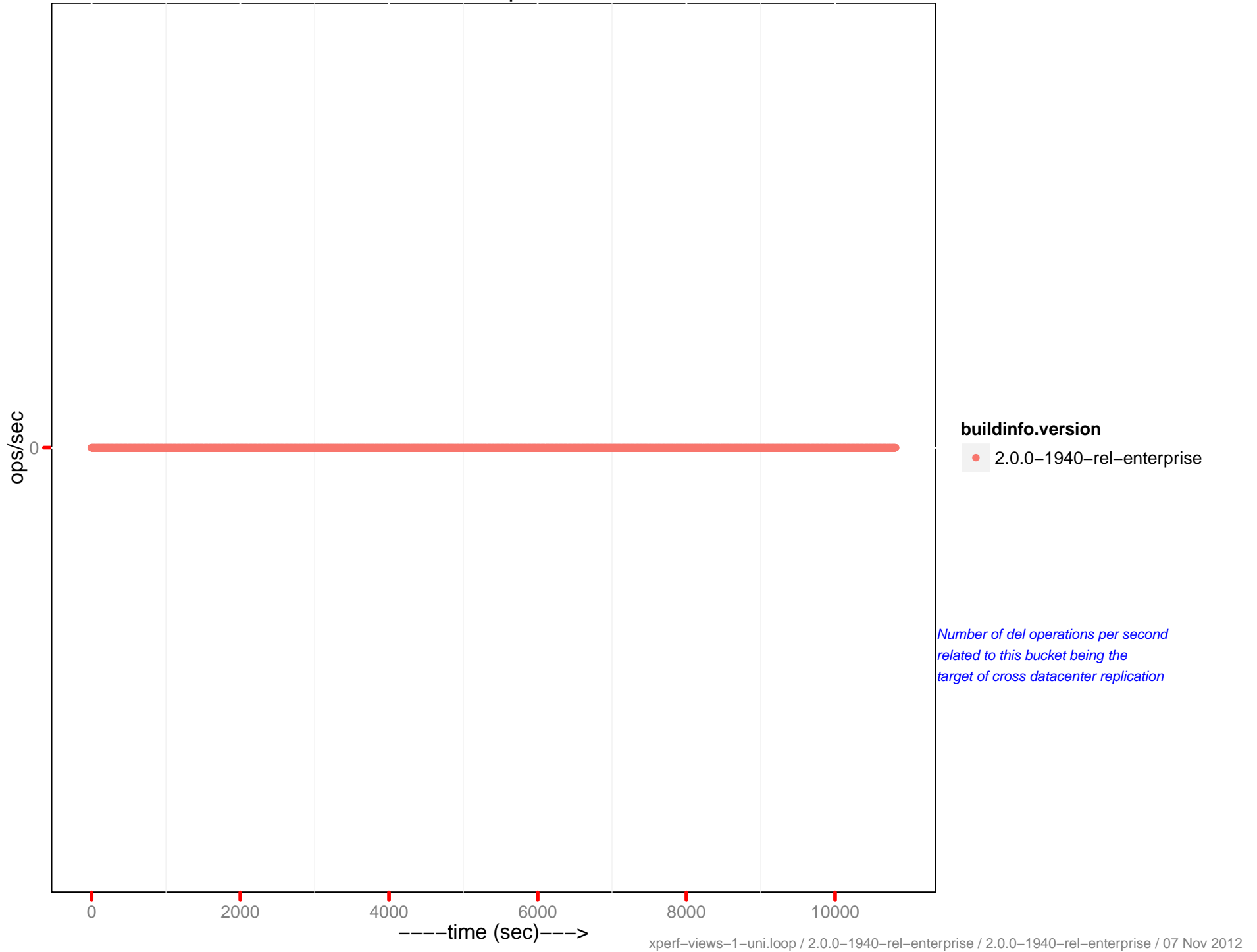
XDC gets per sec



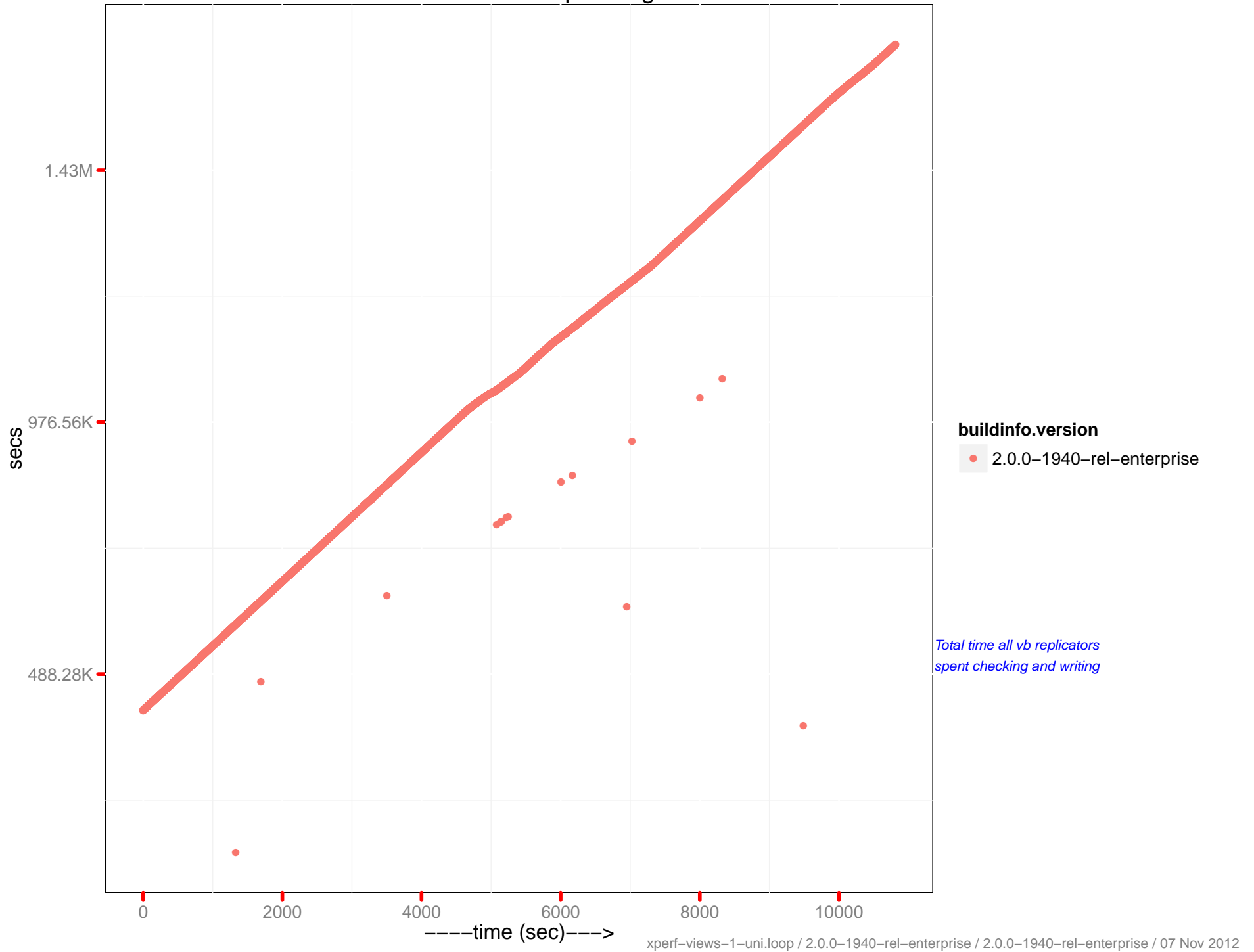
XDC sets per sec



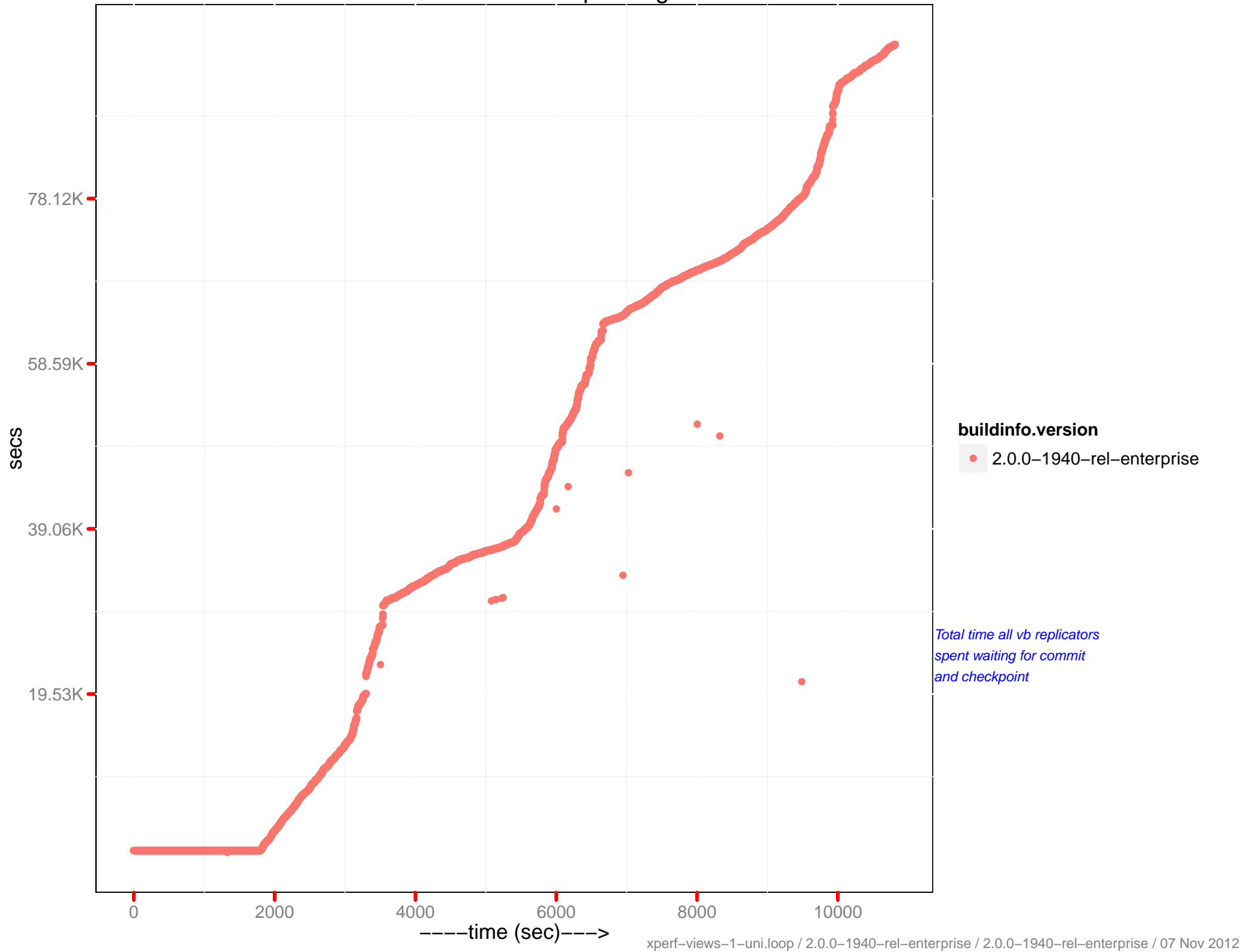
XDC dels per sec



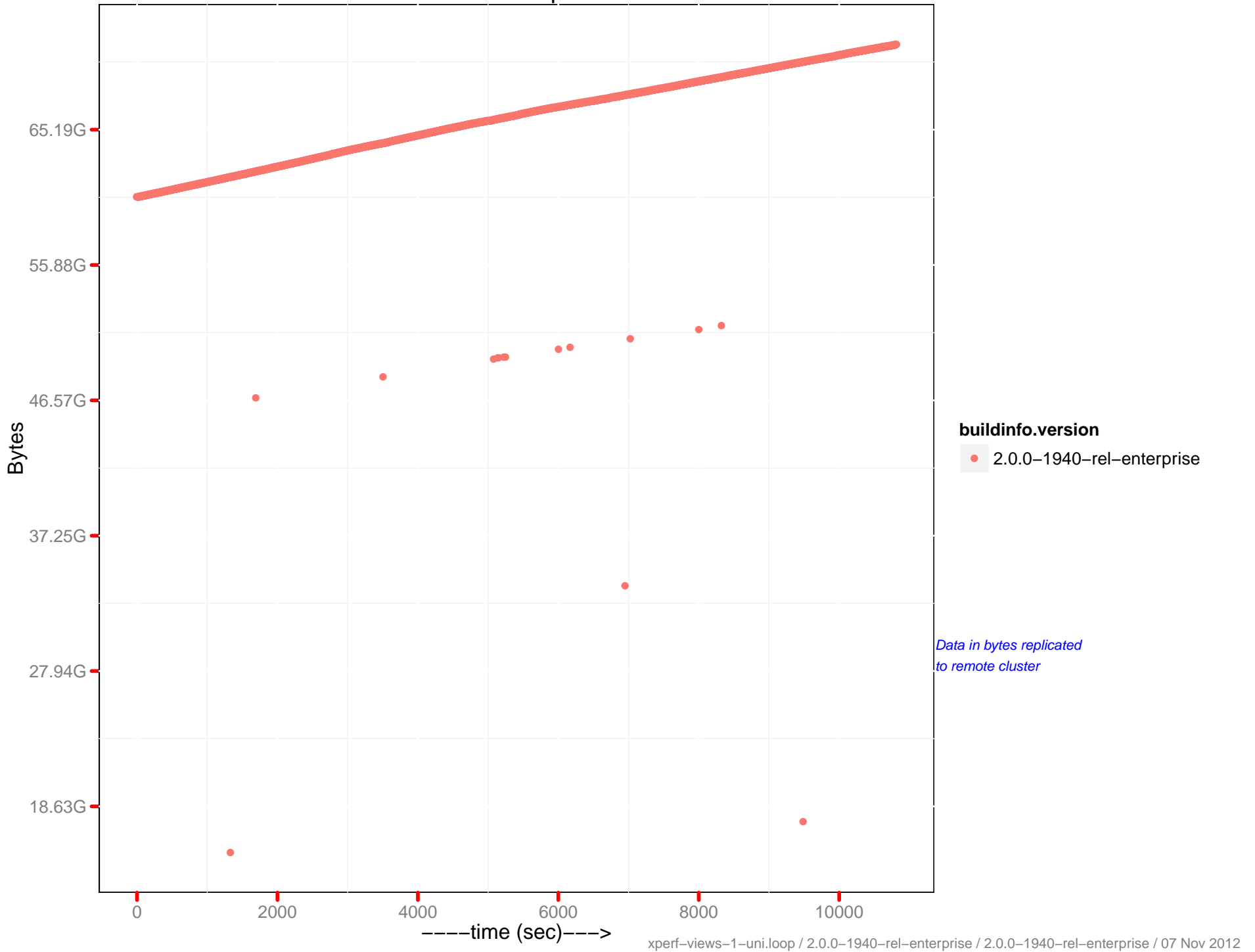
XDCR secs in replicating



XDCR secs in checkpointing



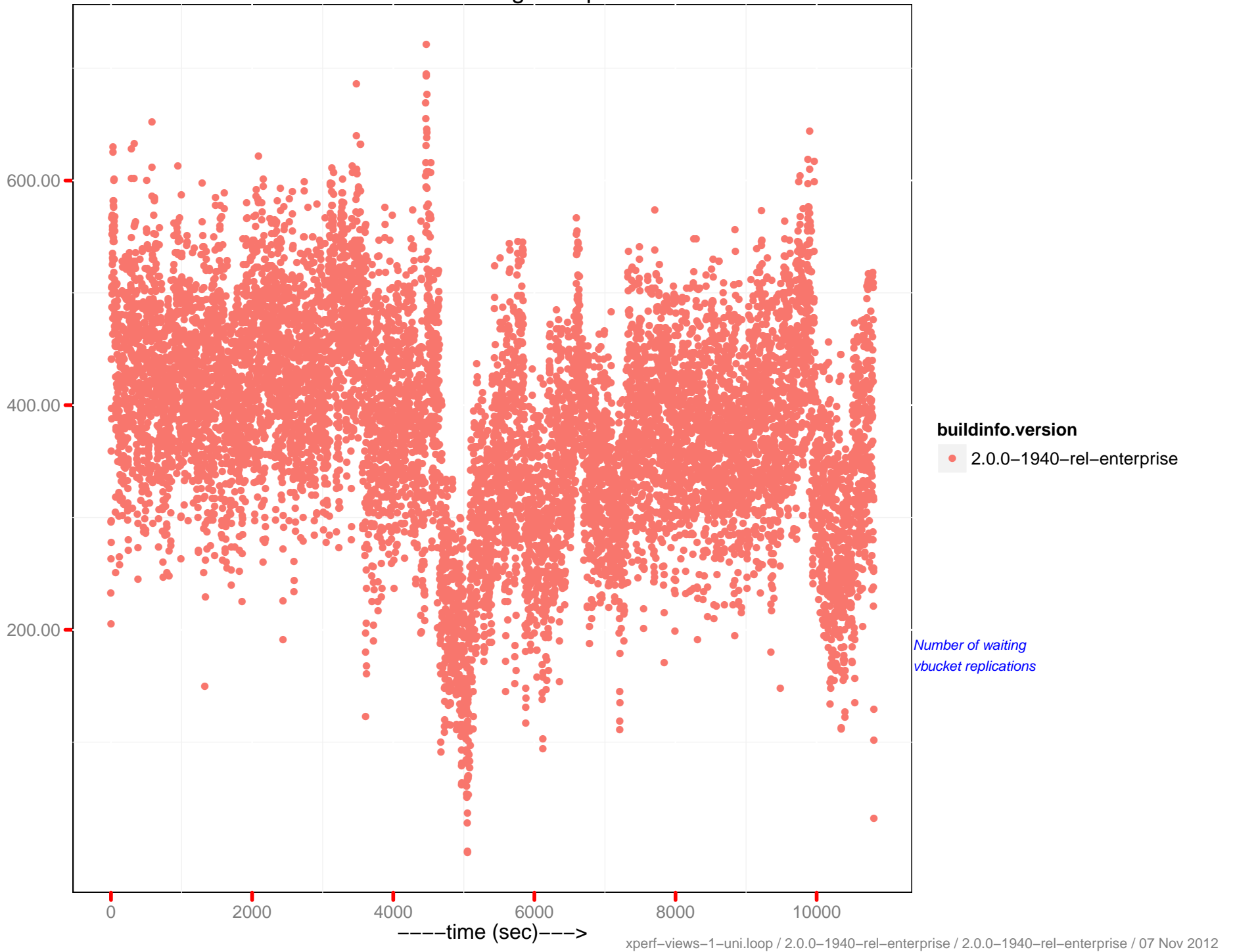
XDCR data replicated



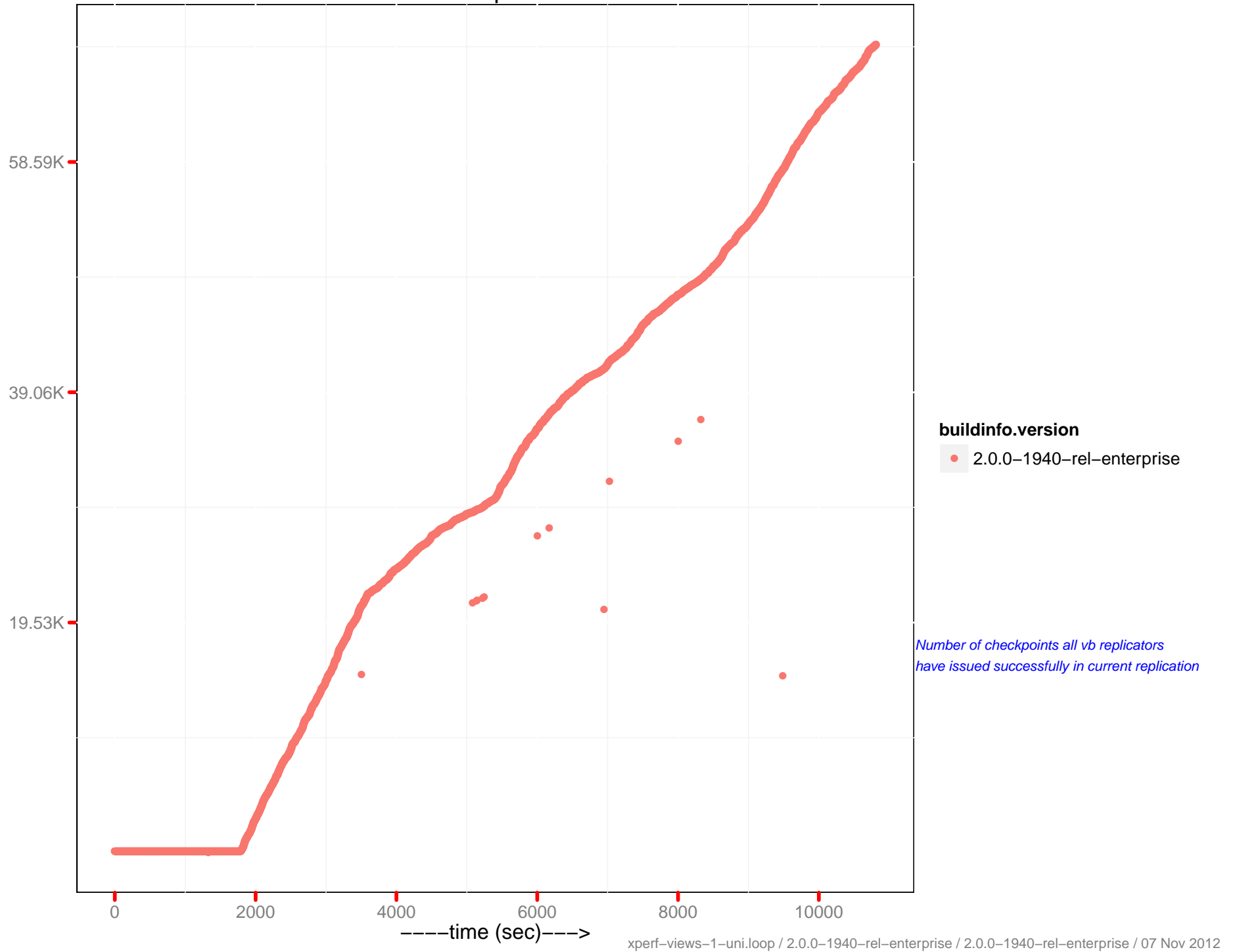
XDCR active vb reps



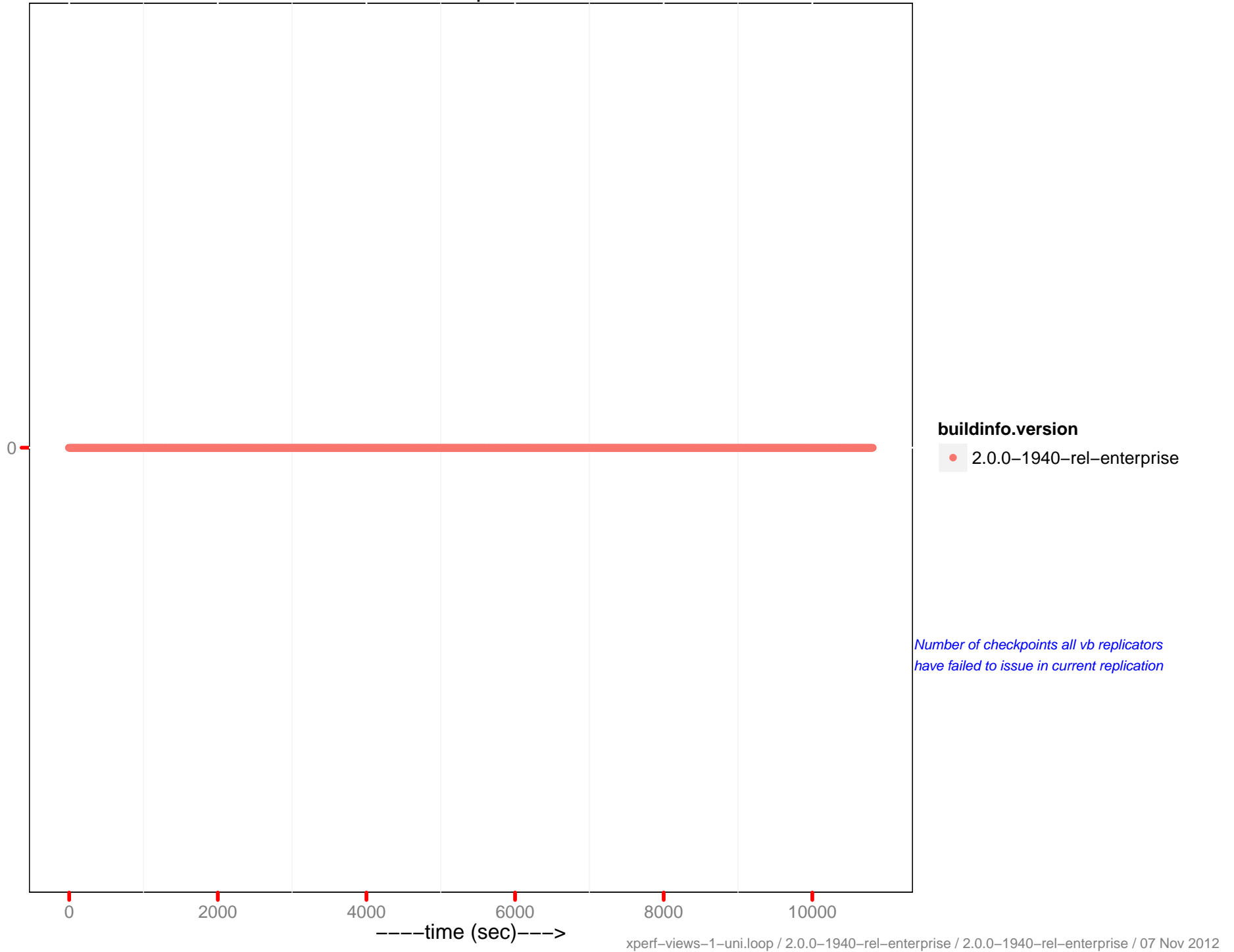
XDCR waiting vb reps



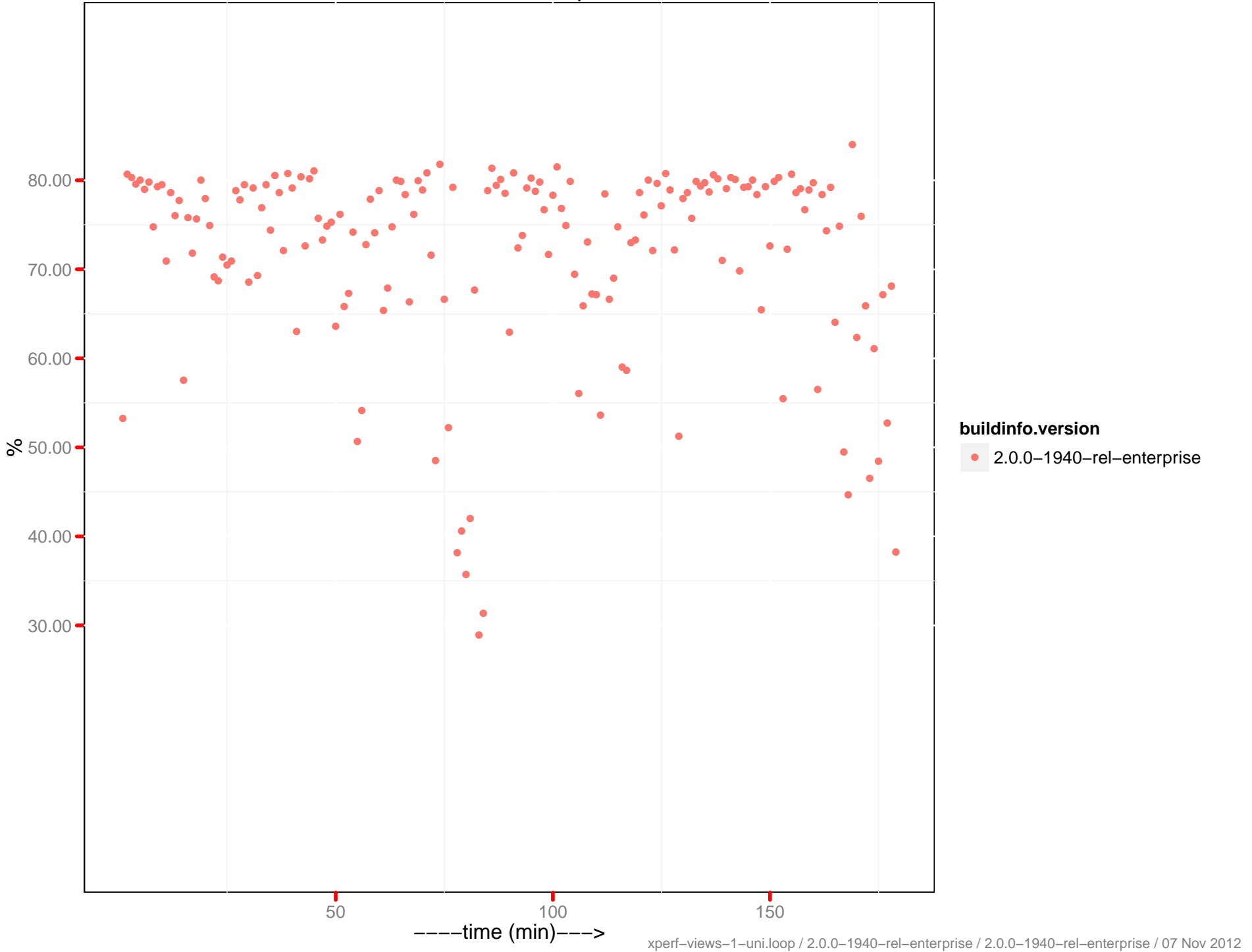
XDCR checkpoints issued



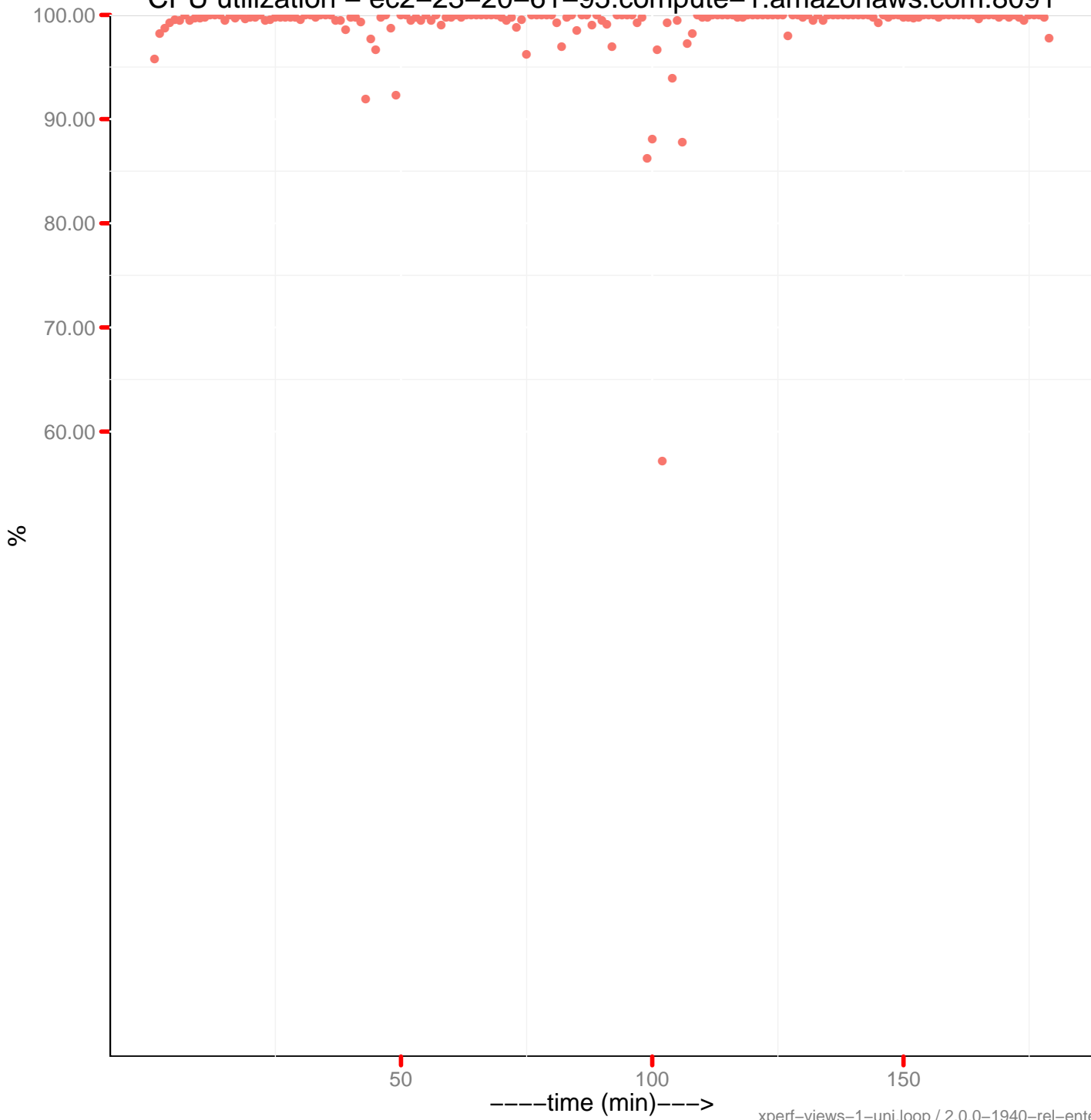
XDCR checkpoints failed



CPU utilization – ec2-107-20-6-150.compute-1.amazonaws.com:8091

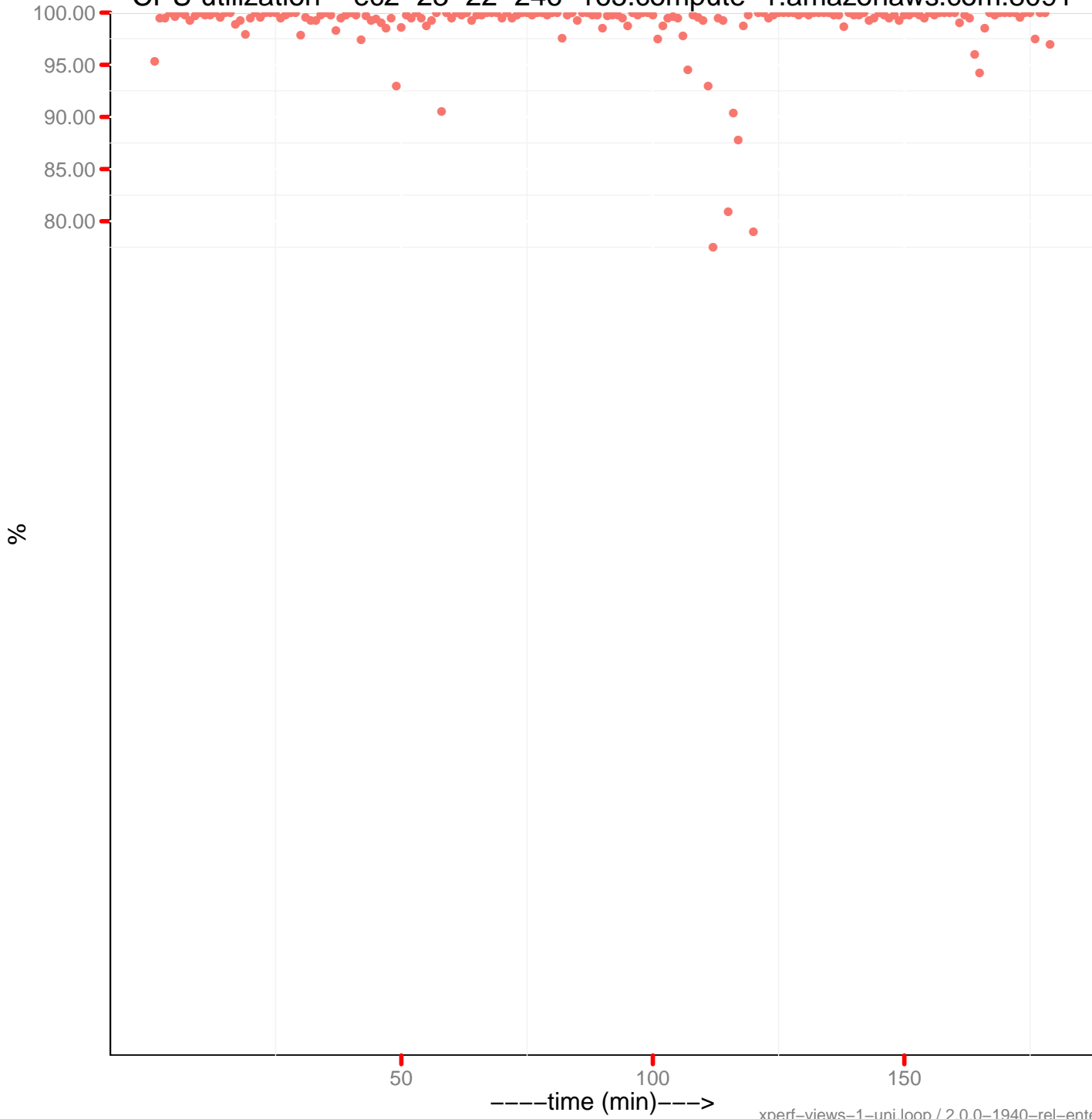


CPU utilization – ec2-23-20-61-95.compute-1.amazonaws.com:8091



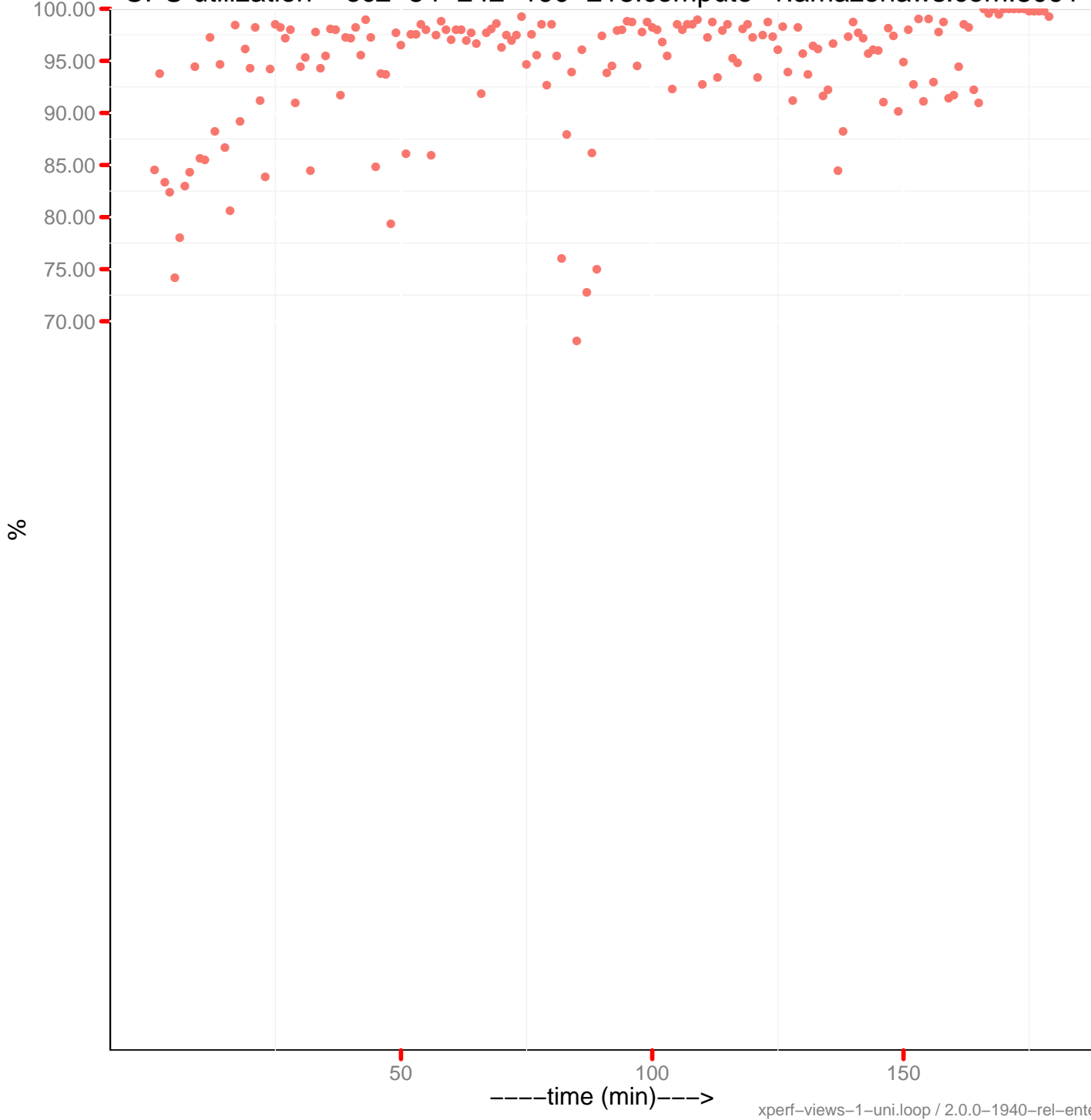
buildinfo.version
● 2.0.0-1940-rel-enterprise

CPU utilization – ec2-23-22-246-165.compute-1.amazonaws.com:8091



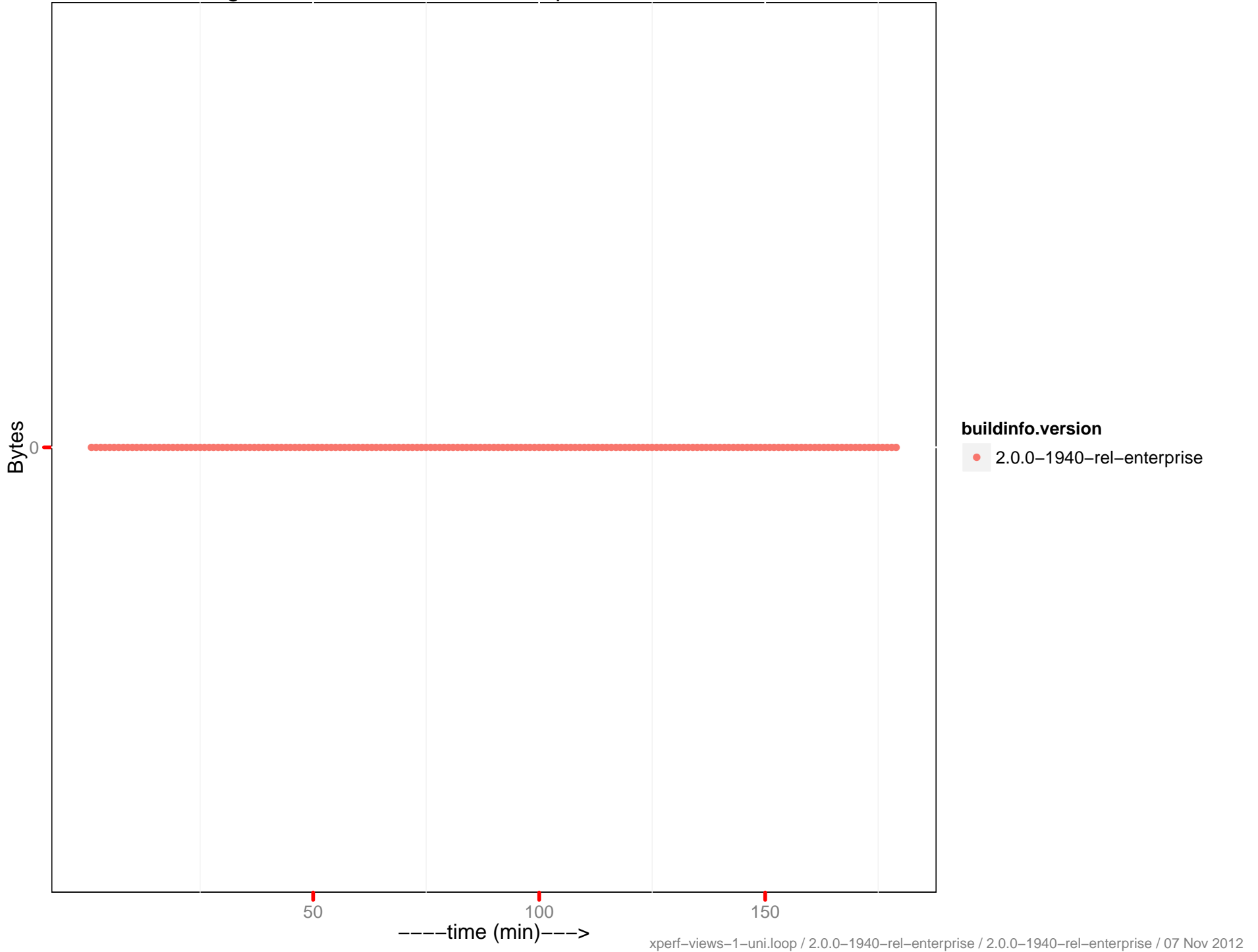
buildinfo.version
● 2.0.0-1940-rel-enterprise

CPU utilization – ec2-54-242-190-218.compute-1.amazonaws.com:8091

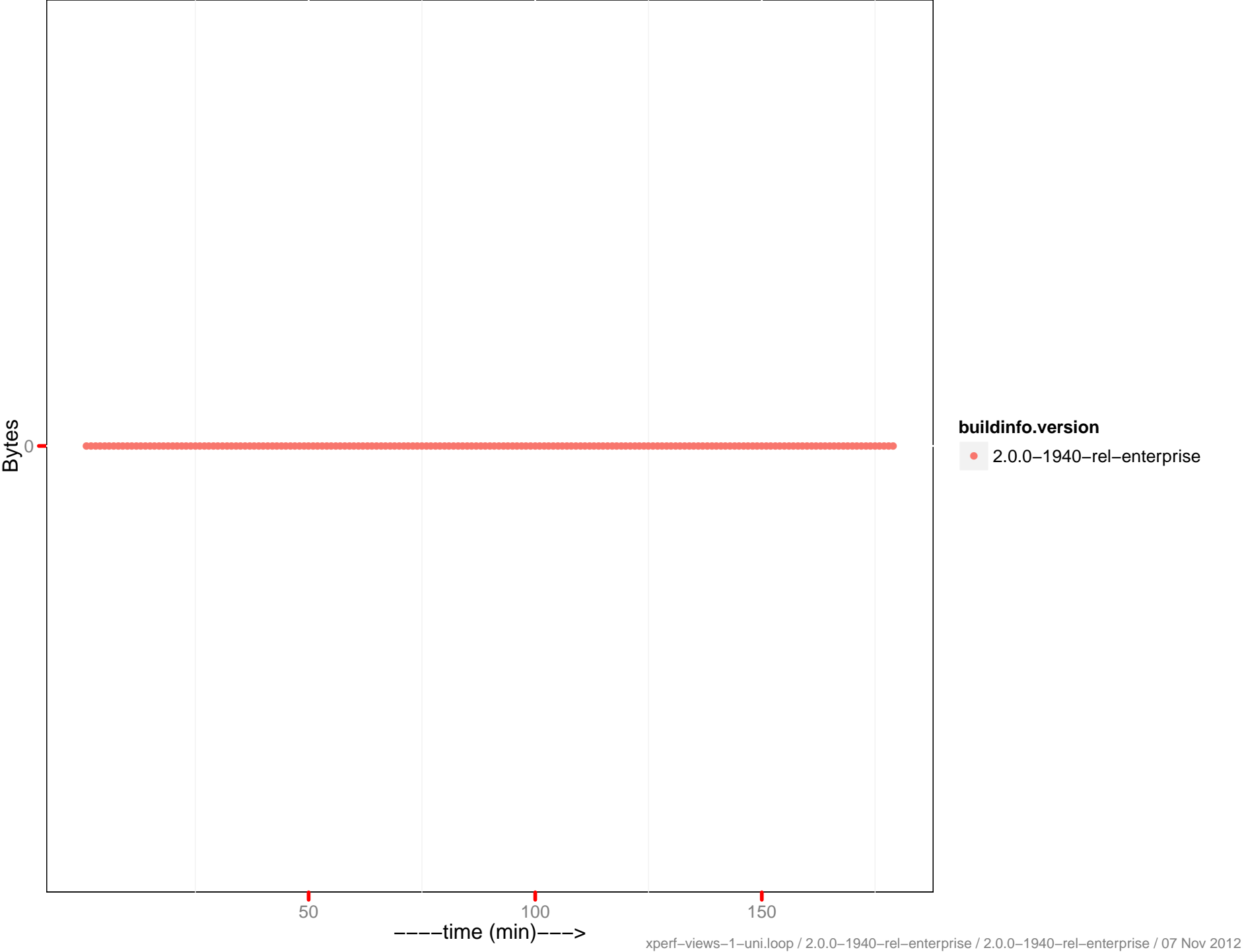


buildinfo.version
● 2.0.0-1940-rel-enterprise

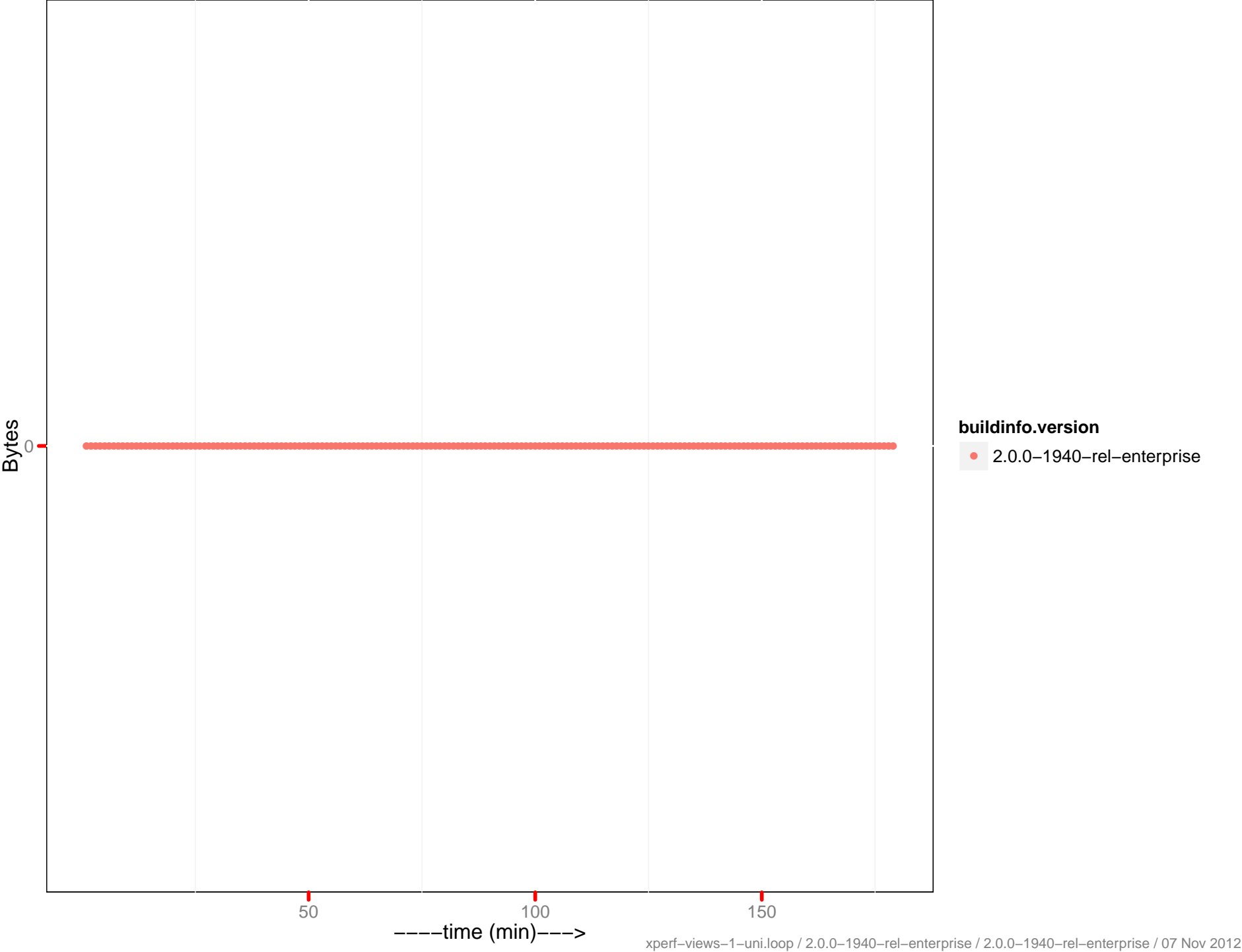
SWAP Usage - ec2-107-20-6-150.compute-1.amazonaws.com:8091



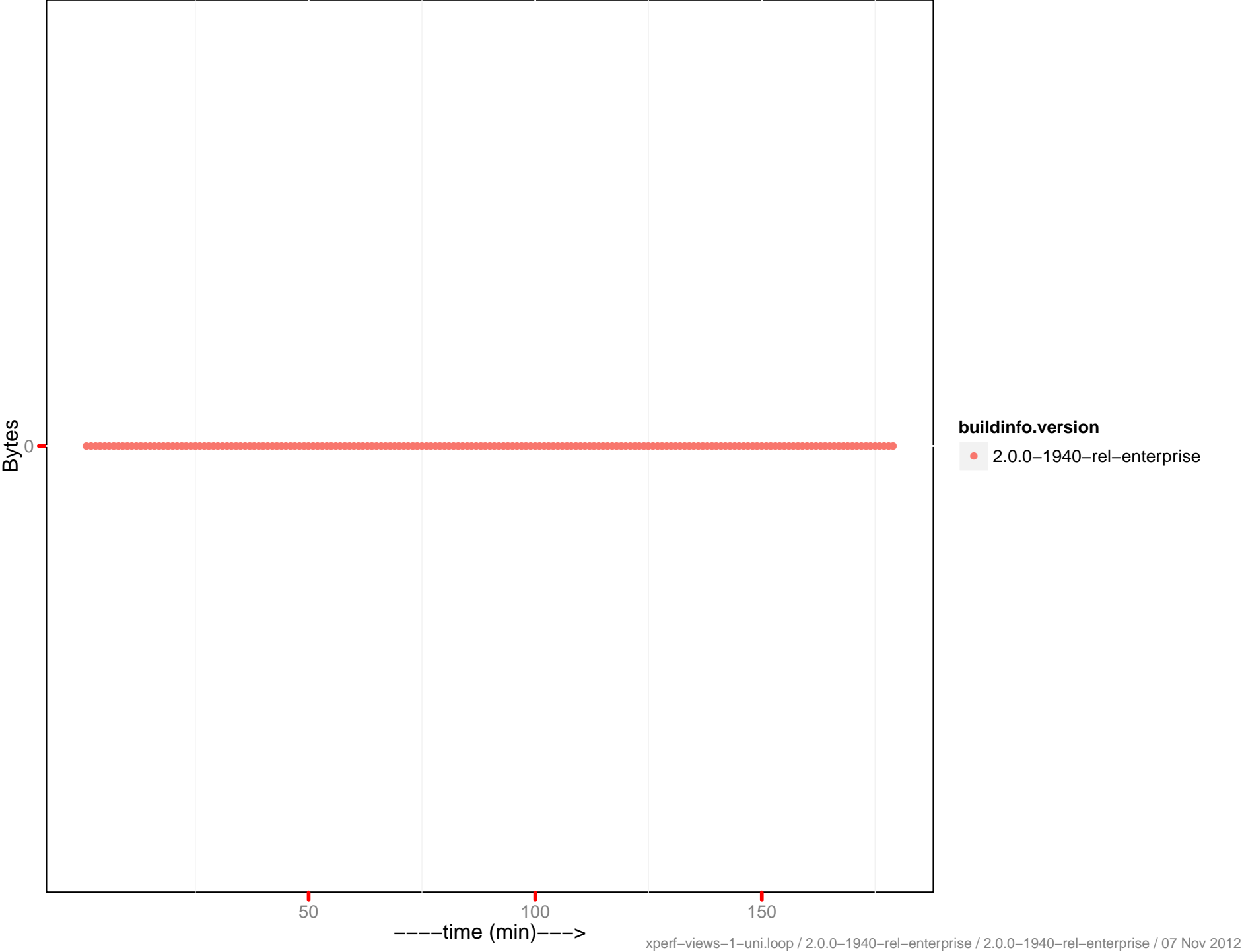
SWAP Usage - ec2-23-20-61-95.compute-1.amazonaws.com:8091



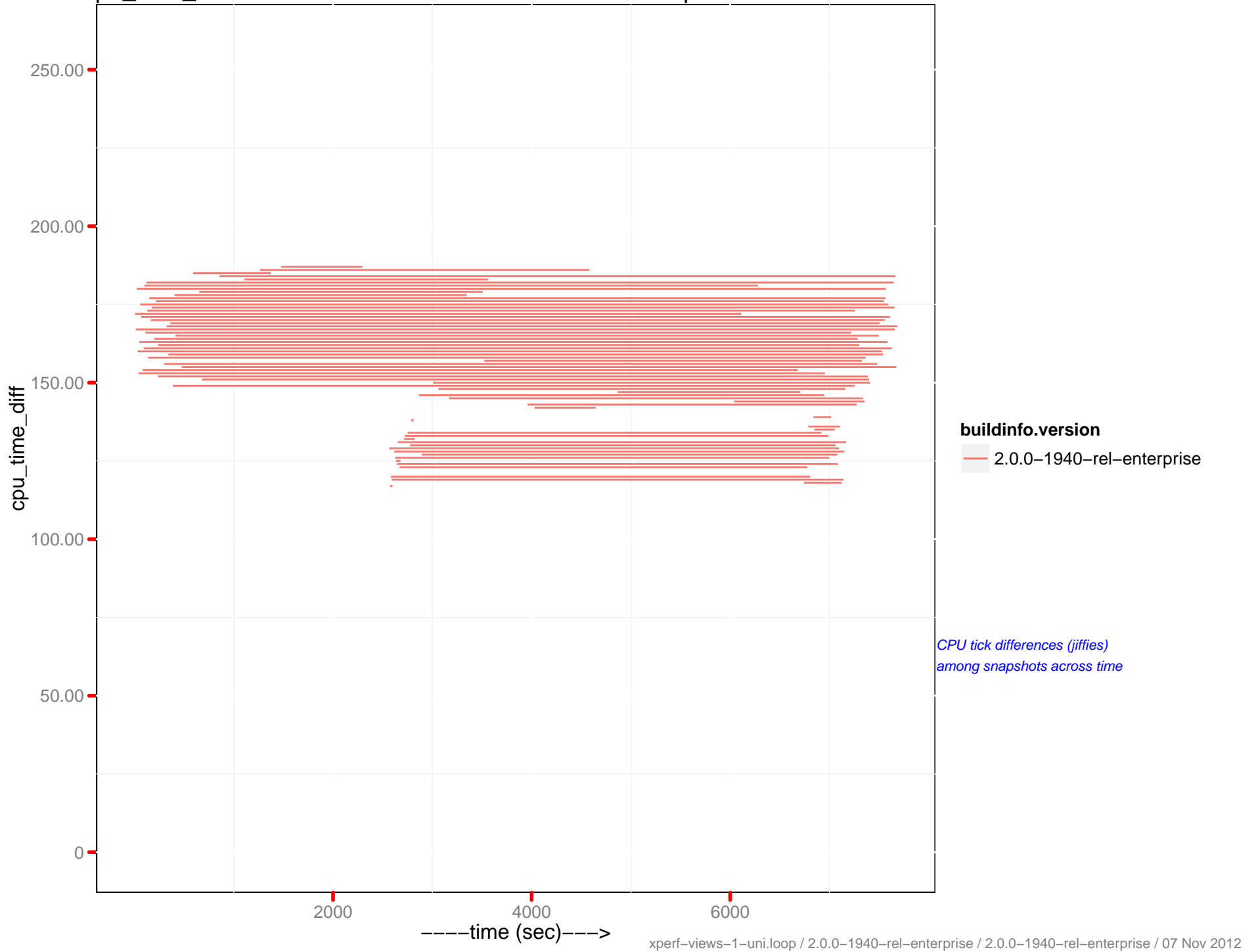
SWAP Usage - ec2-23-22-246-165.compute-1.amazonaws.com:8091



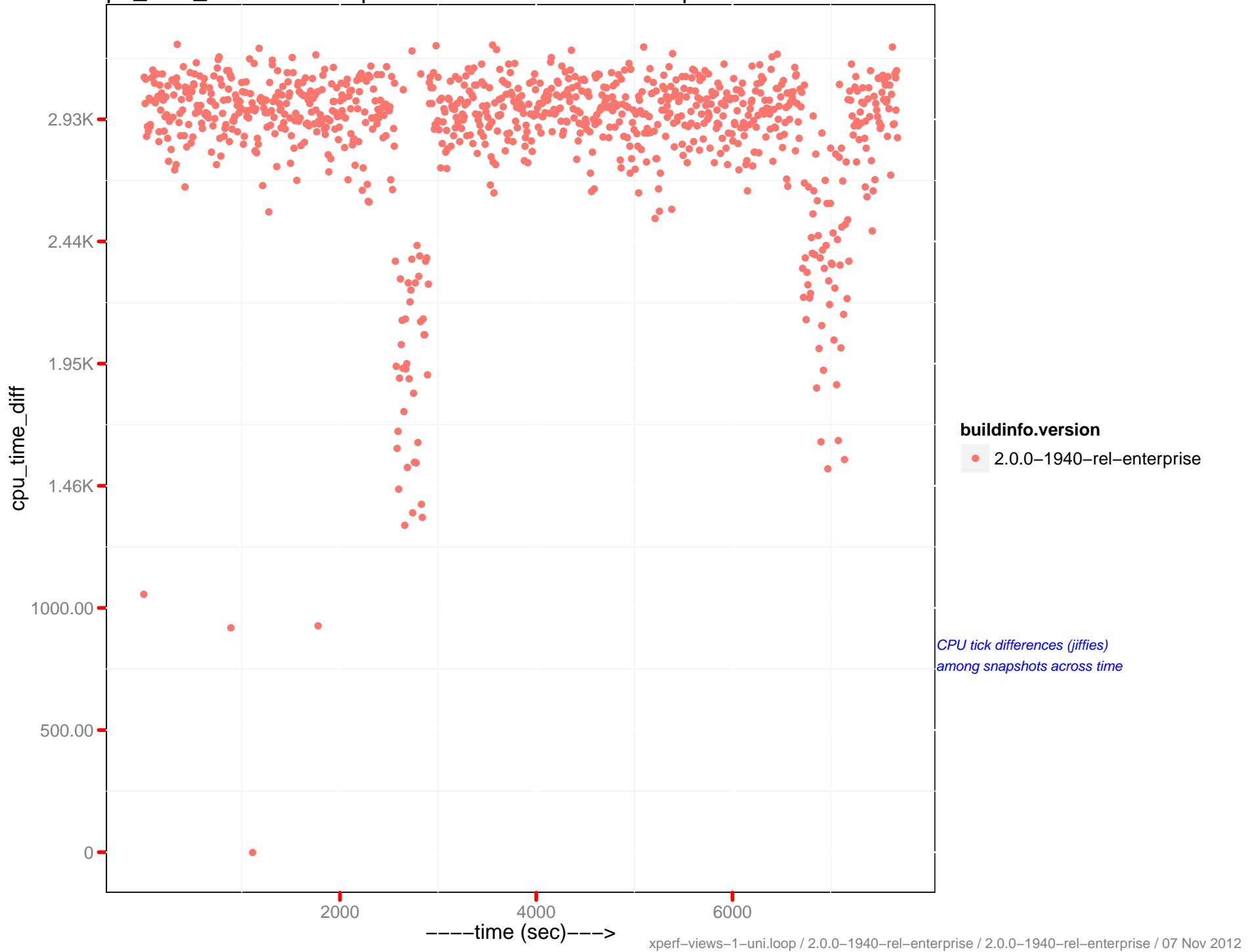
SWAP Usage - ec2-54-242-190-218.compute-1.amazonaws.com:8091



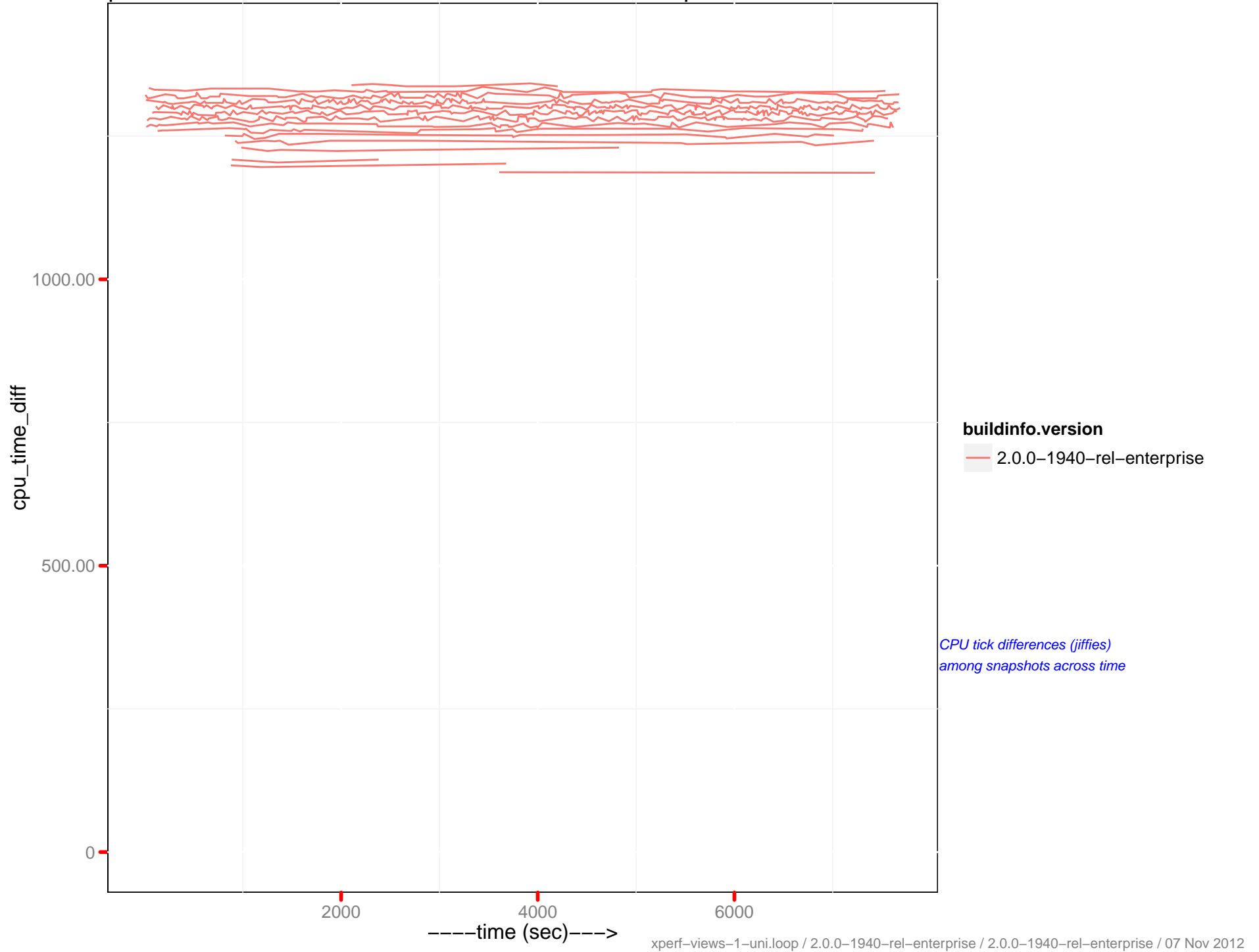
cpu_time_diff: memcached - ec2-107-20-6-150.compute-1.amazonaws.com



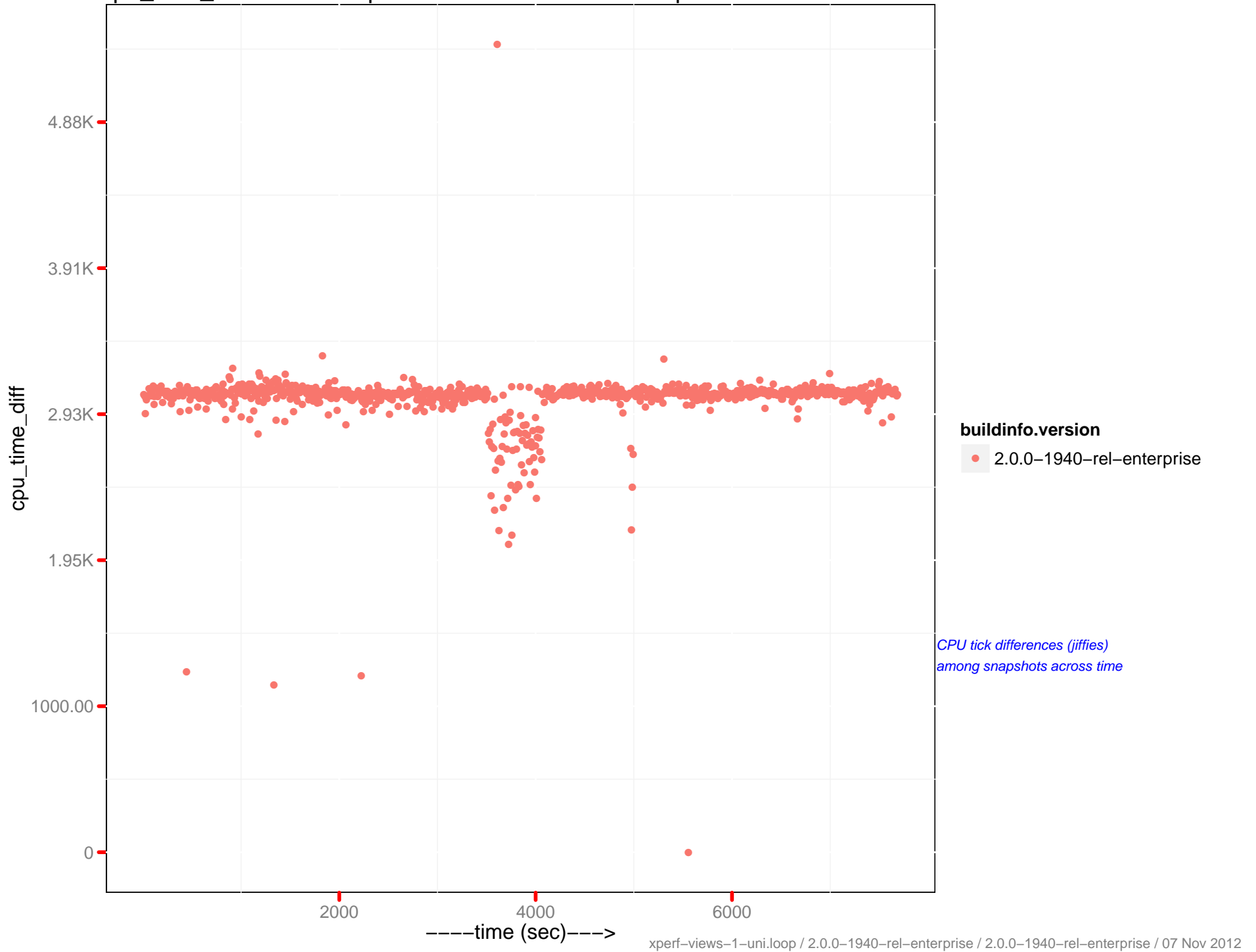
cpu_time_diff : beam.smp - ec2-107-20-6-150.compute-1.amazonaws.com



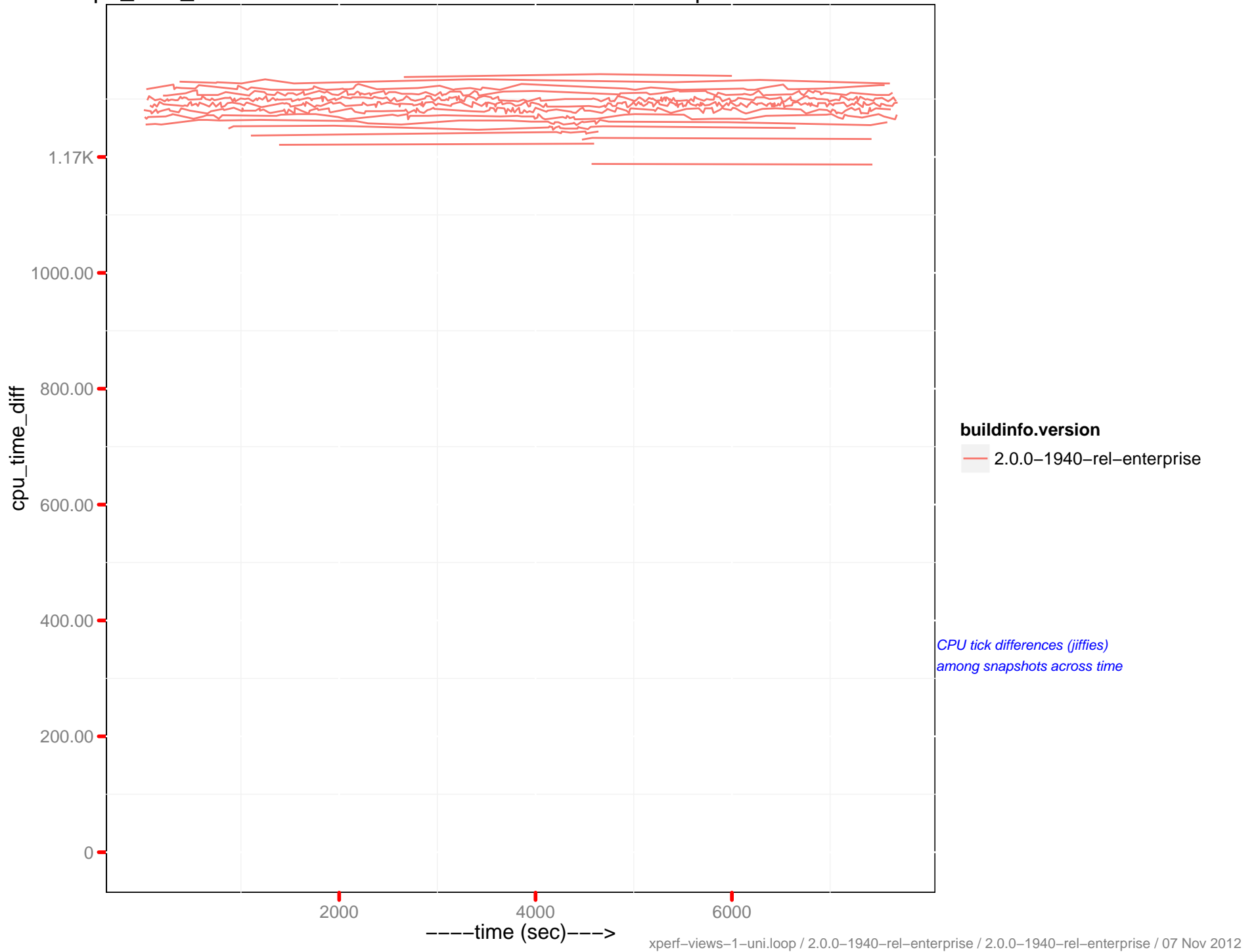
cpu_time_diff: memcached - ec2-23-20-61-95.compute-1.amazonaws.com



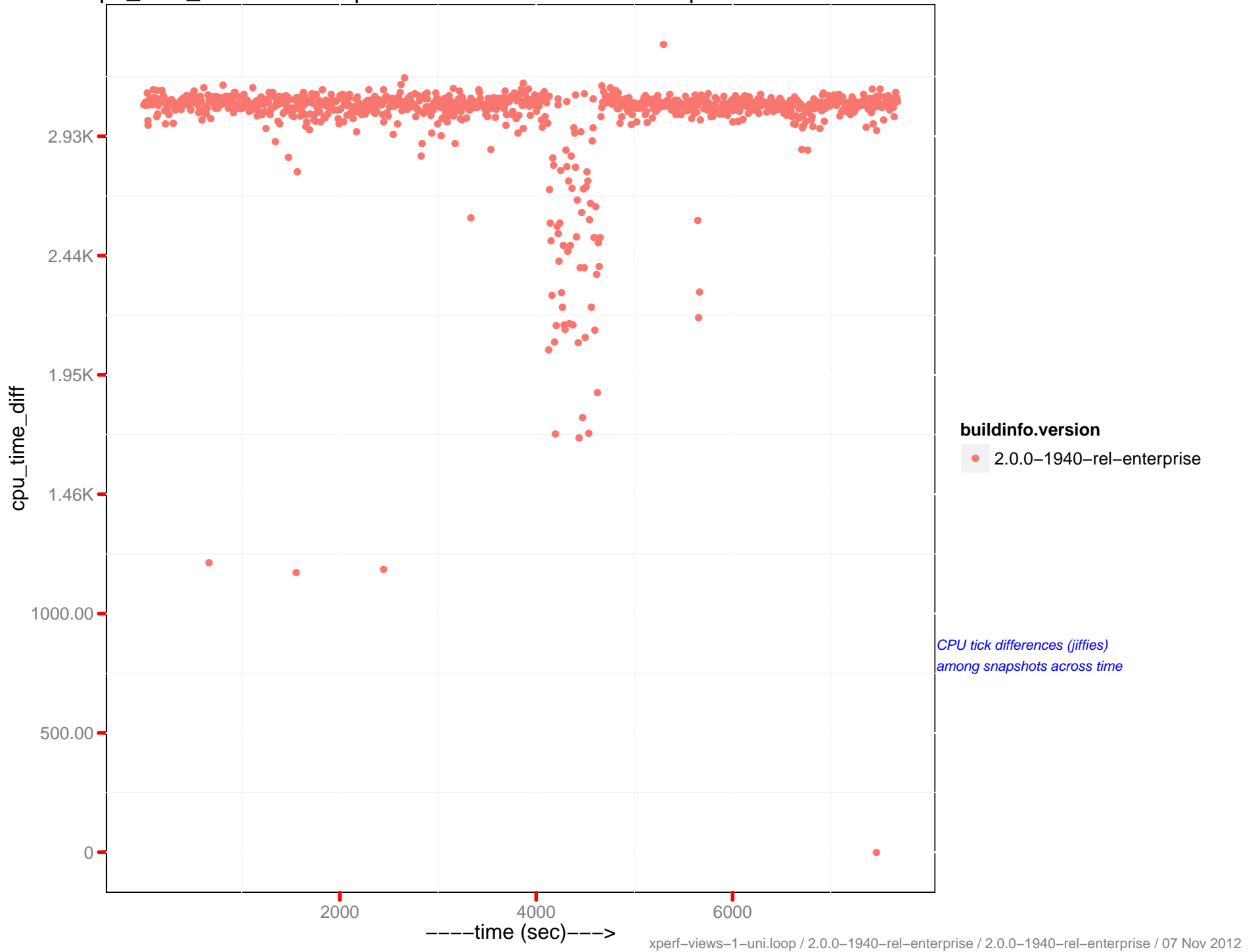
cpu_time_diff : beam.smp - ec2-23-20-61-95.compute-1.amazonaws.com



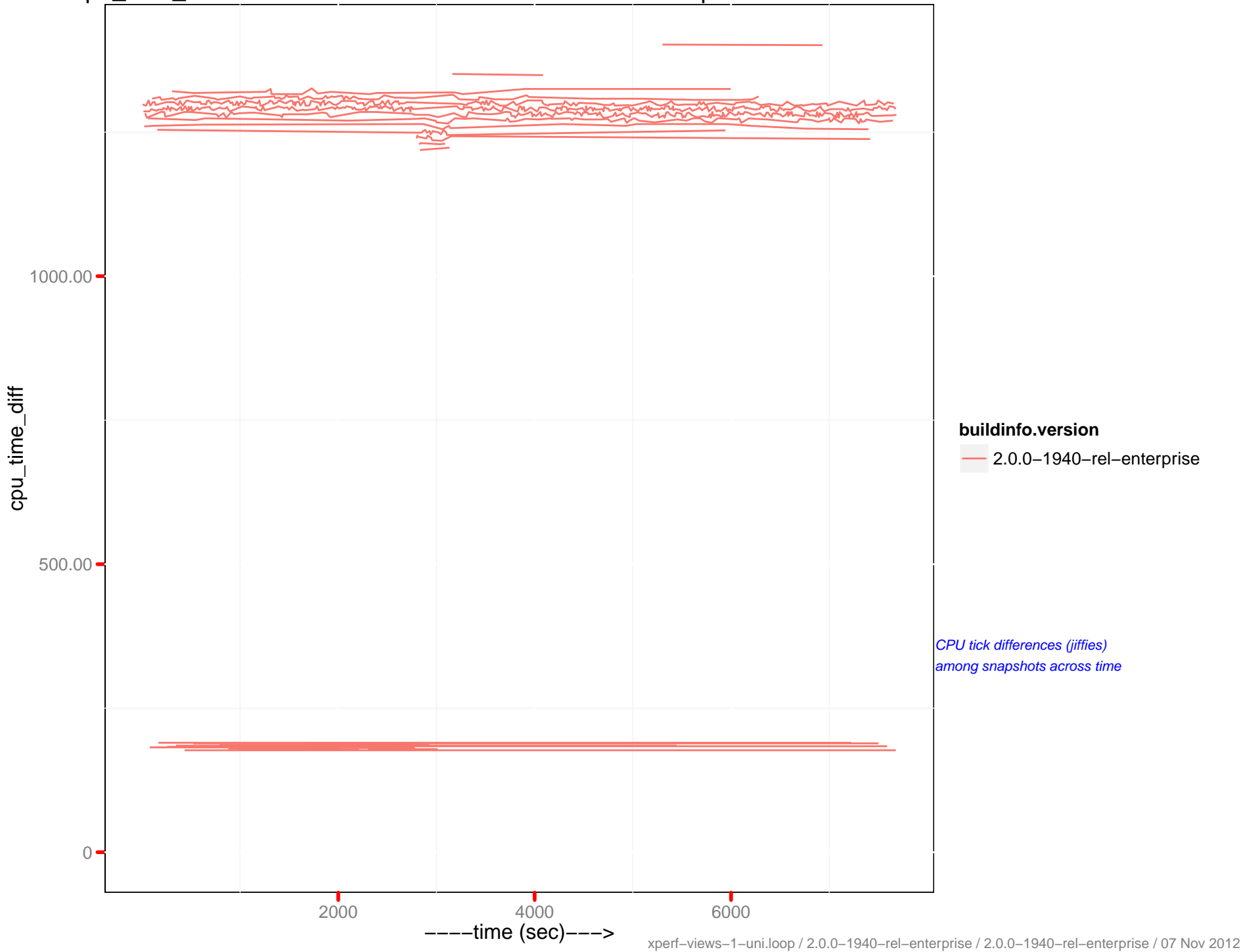
cpu_time_diff: memcached - ec2-23-22-246-165.compute-1.amazonaws.com



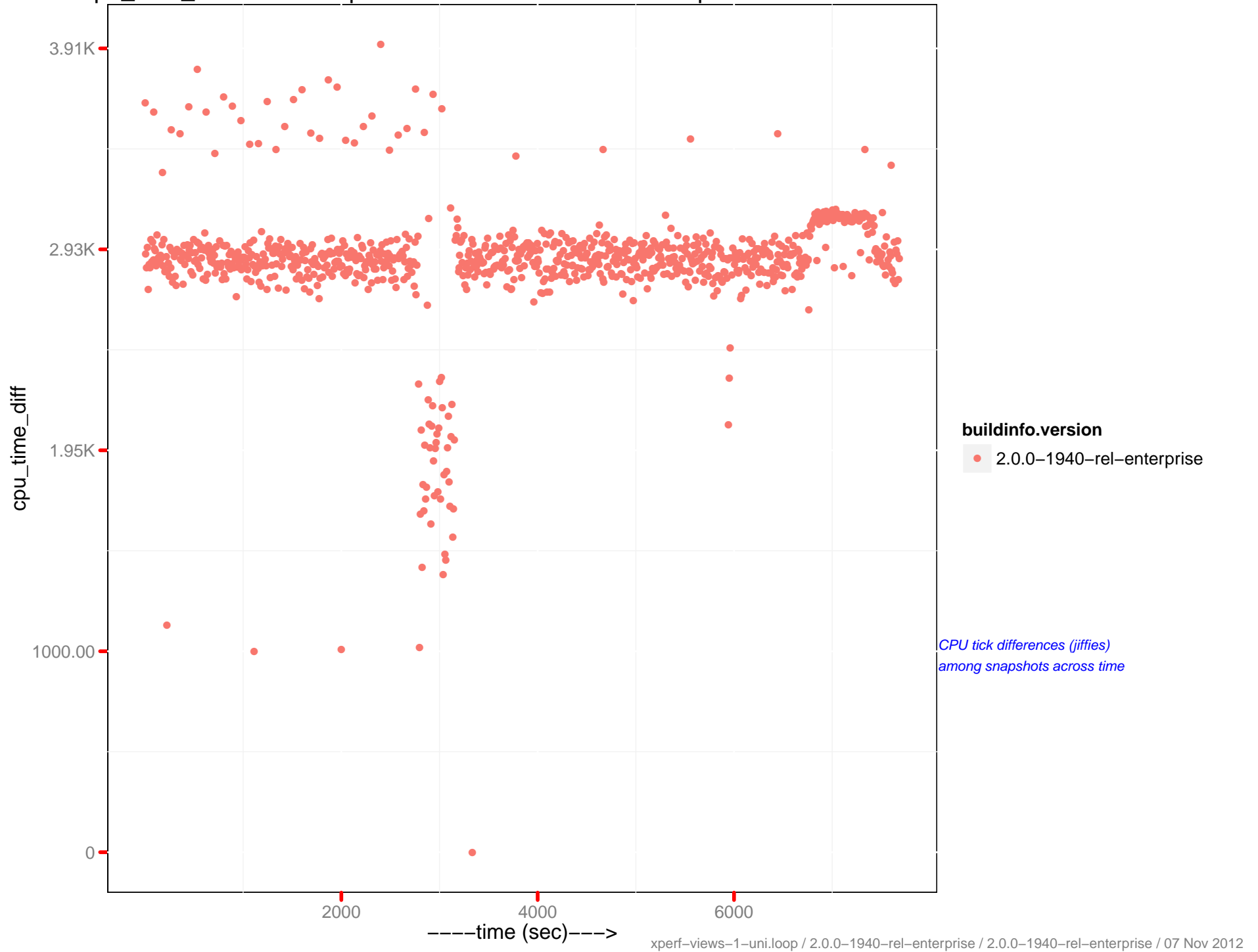
cpu_time_diff : beam.smp - ec2-23-22-246-165.compute-1.amazonaws.com



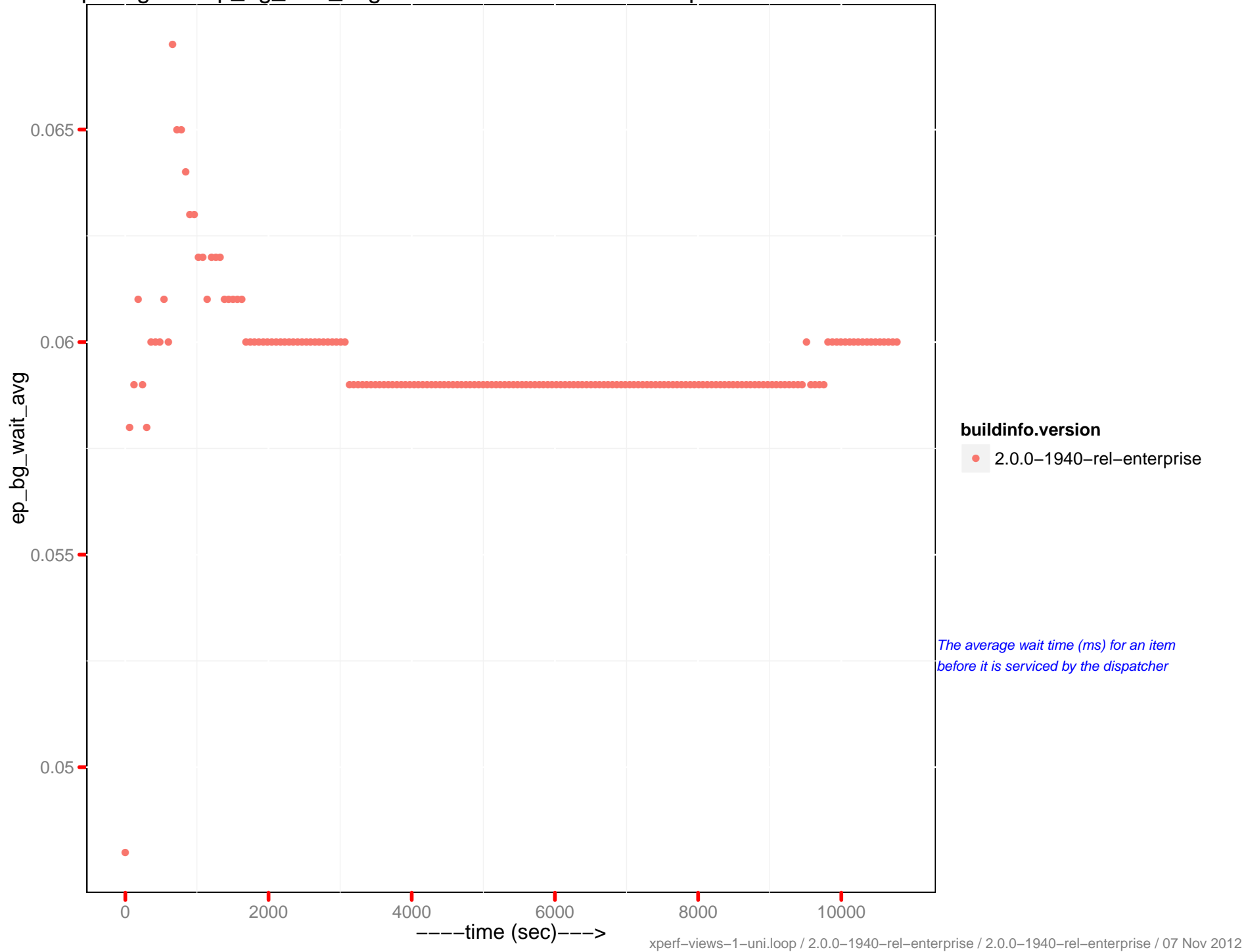
cpu_time_diff: memcached - ec2-54-242-190-218.compute-1.amazonaws.com



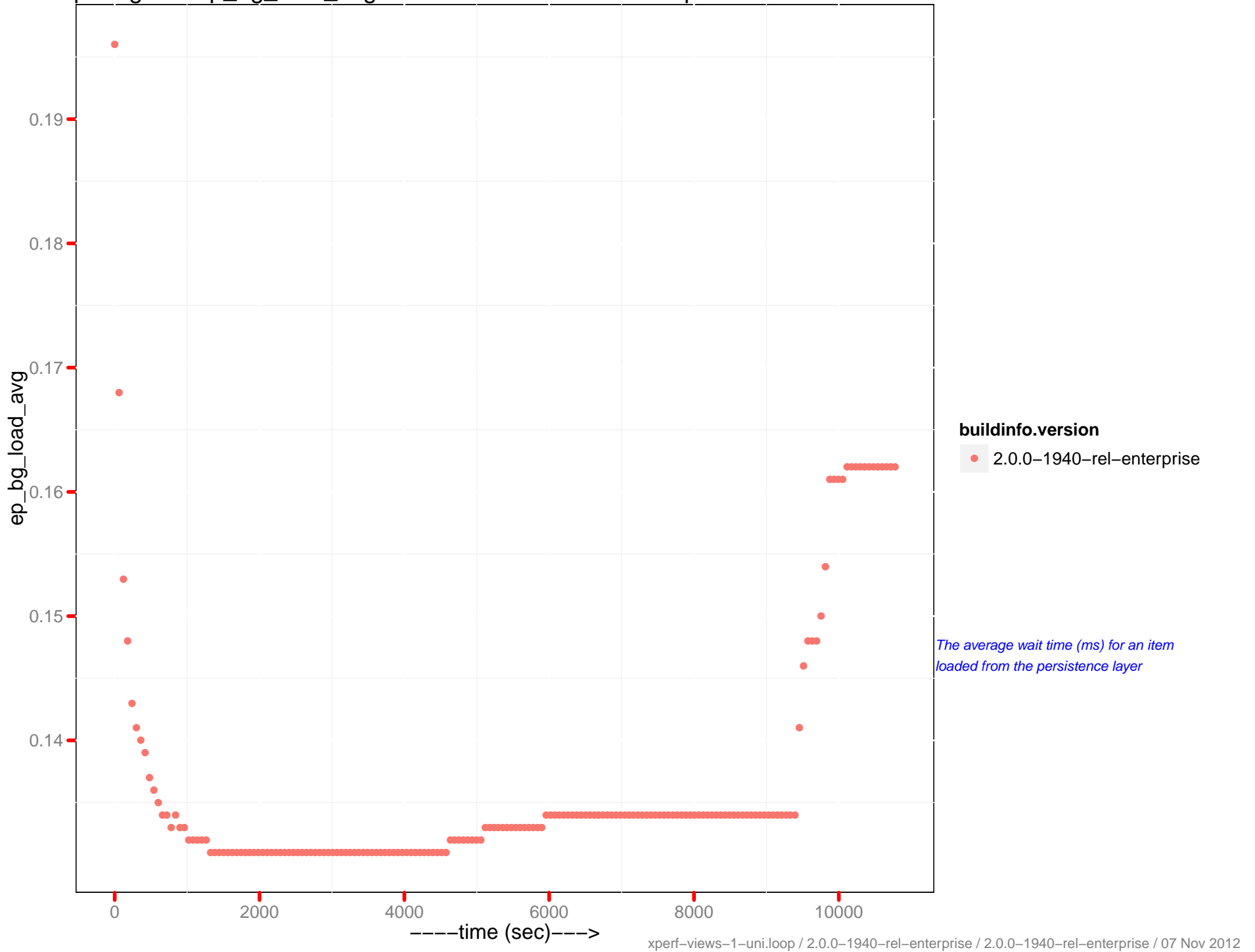
cpu_time_diff : beam.smp - ec2-54-242-190-218.compute-1.amazonaws.com



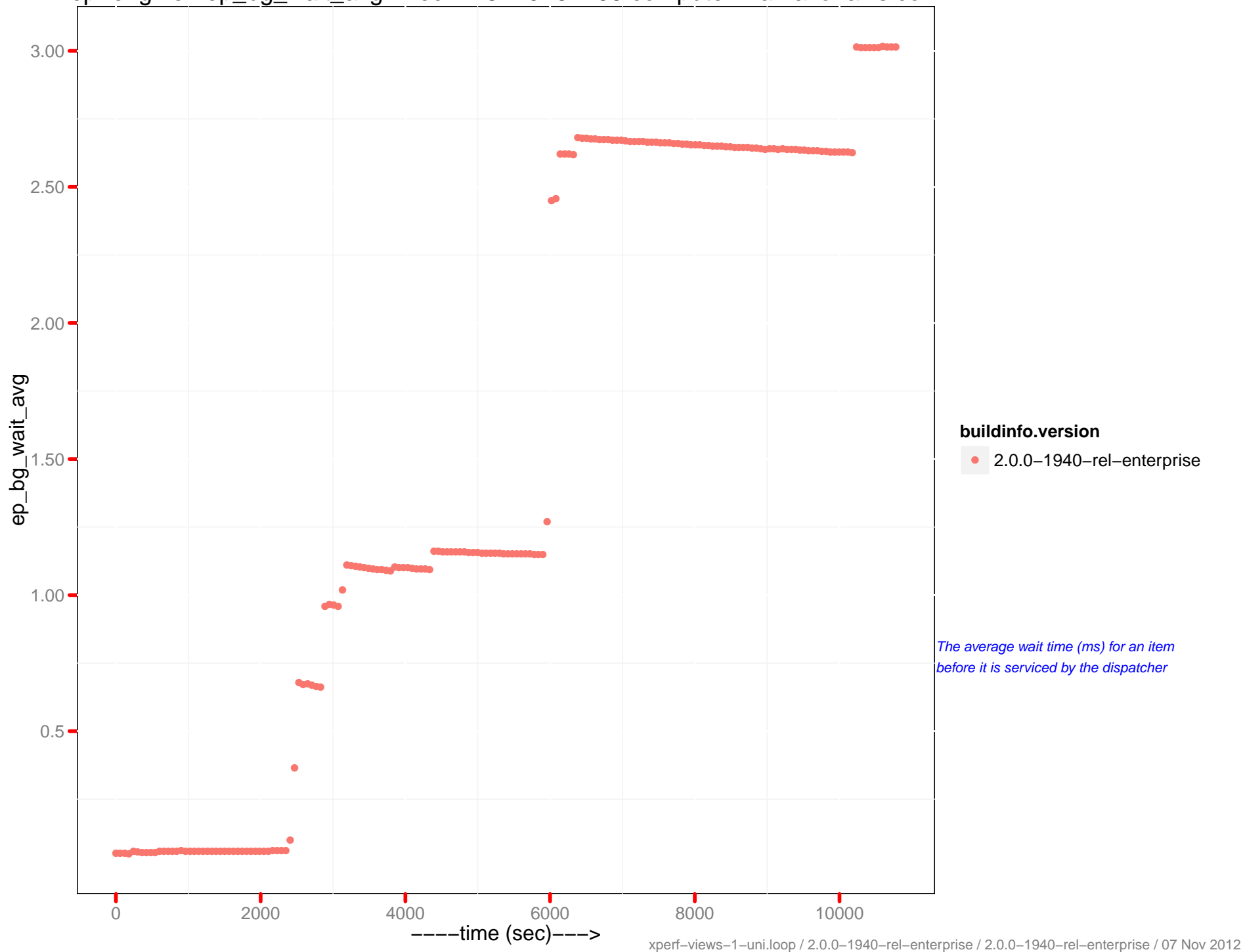
ep-engine : ep_bg_wait_avg - ec2-107-20-6-150.compute-1.amazonaws.com

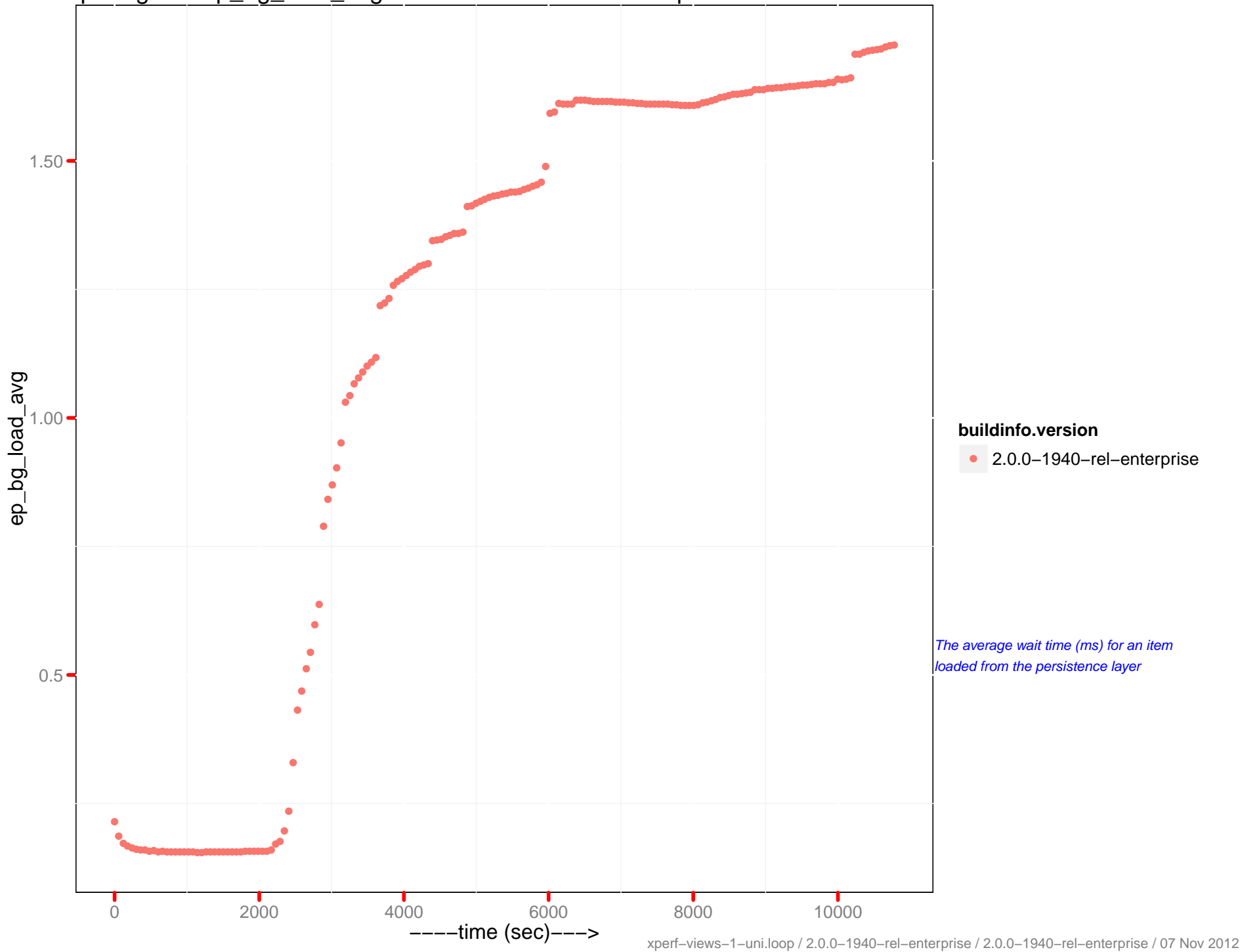


ep-engine : ep_bg_load_avg - ec2-107-20-6-150.compute-1.amazonaws.com

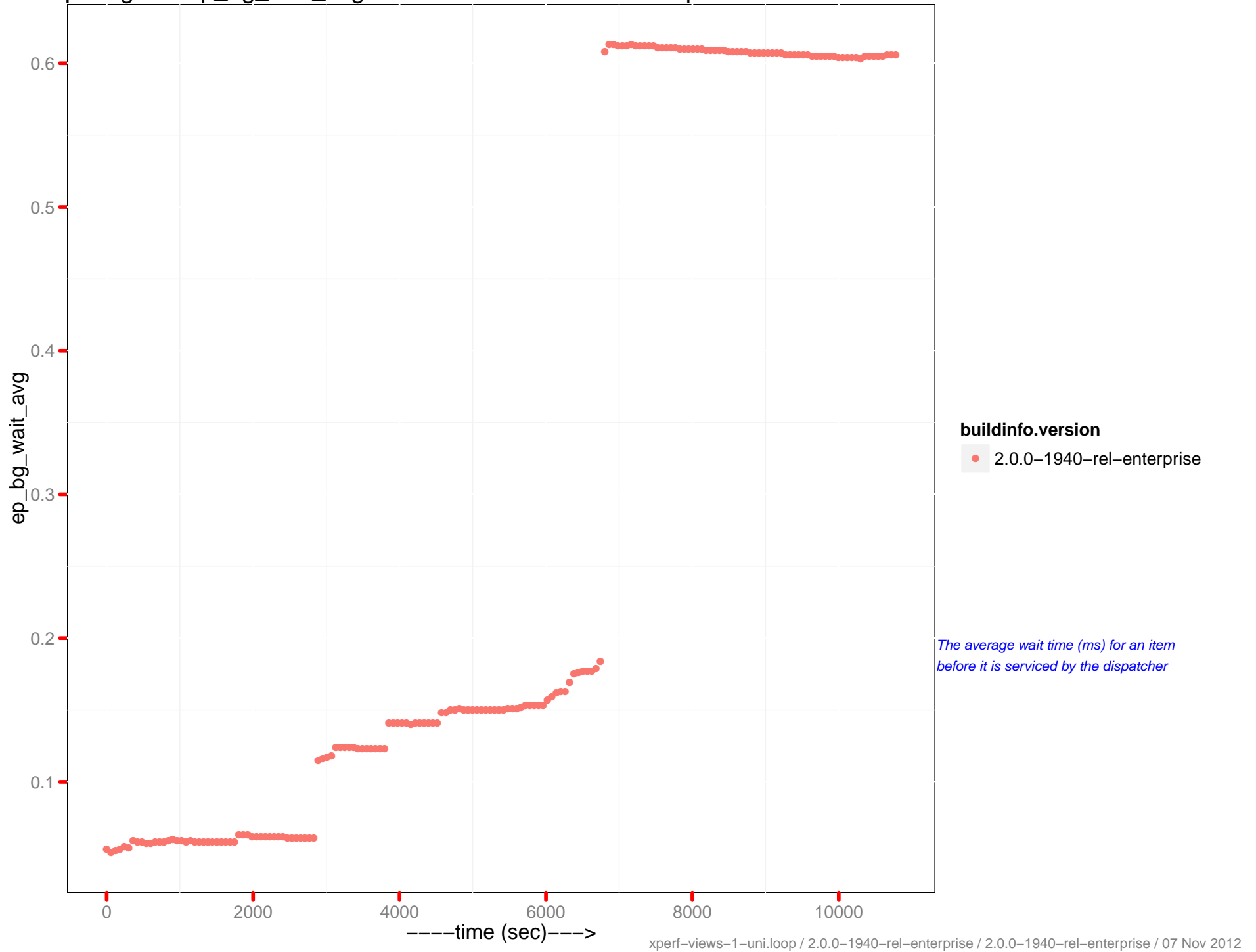


ep-engine : ep_bg_wait_avg - ec2-23-20-61-95.compute-1.amazonaws.com

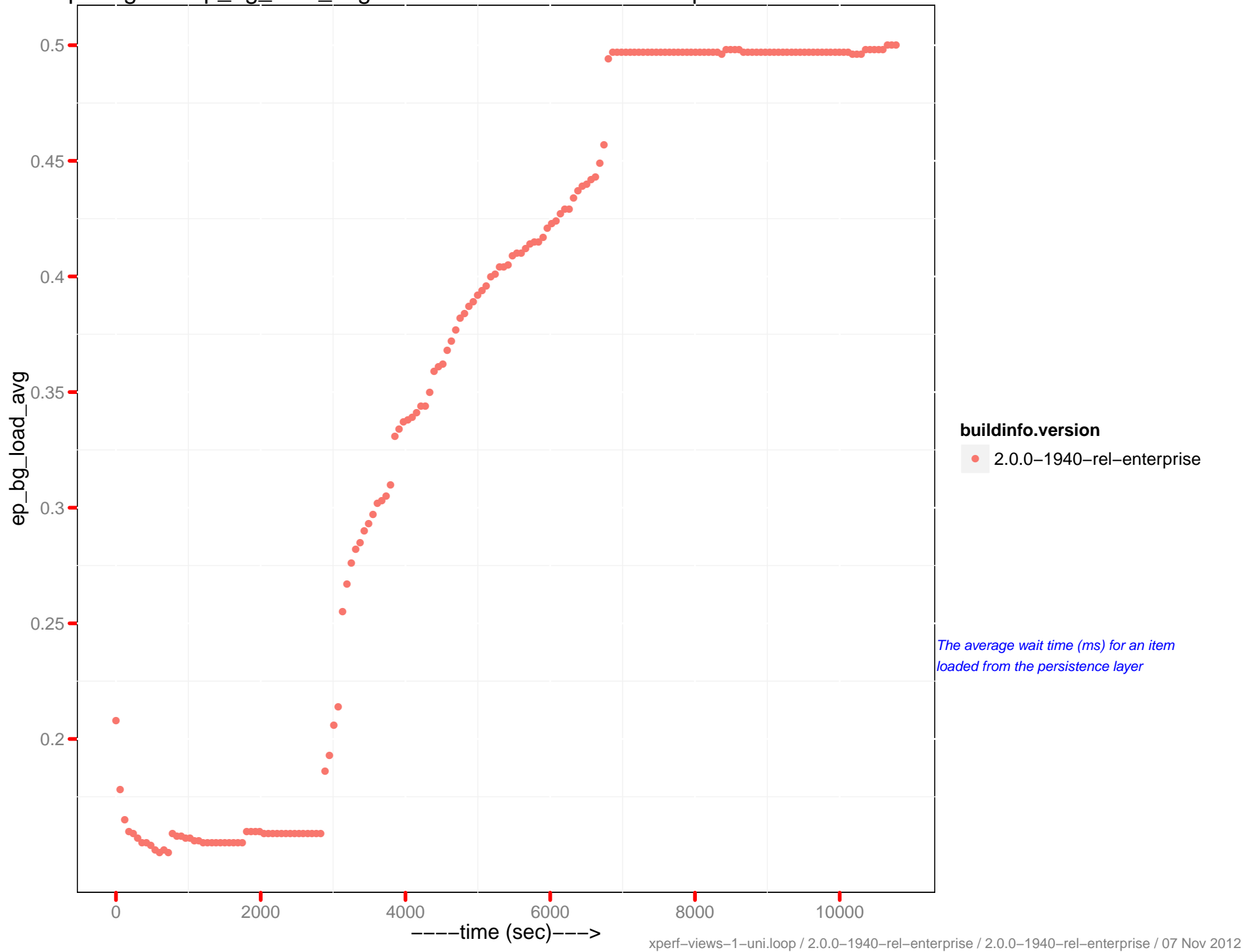




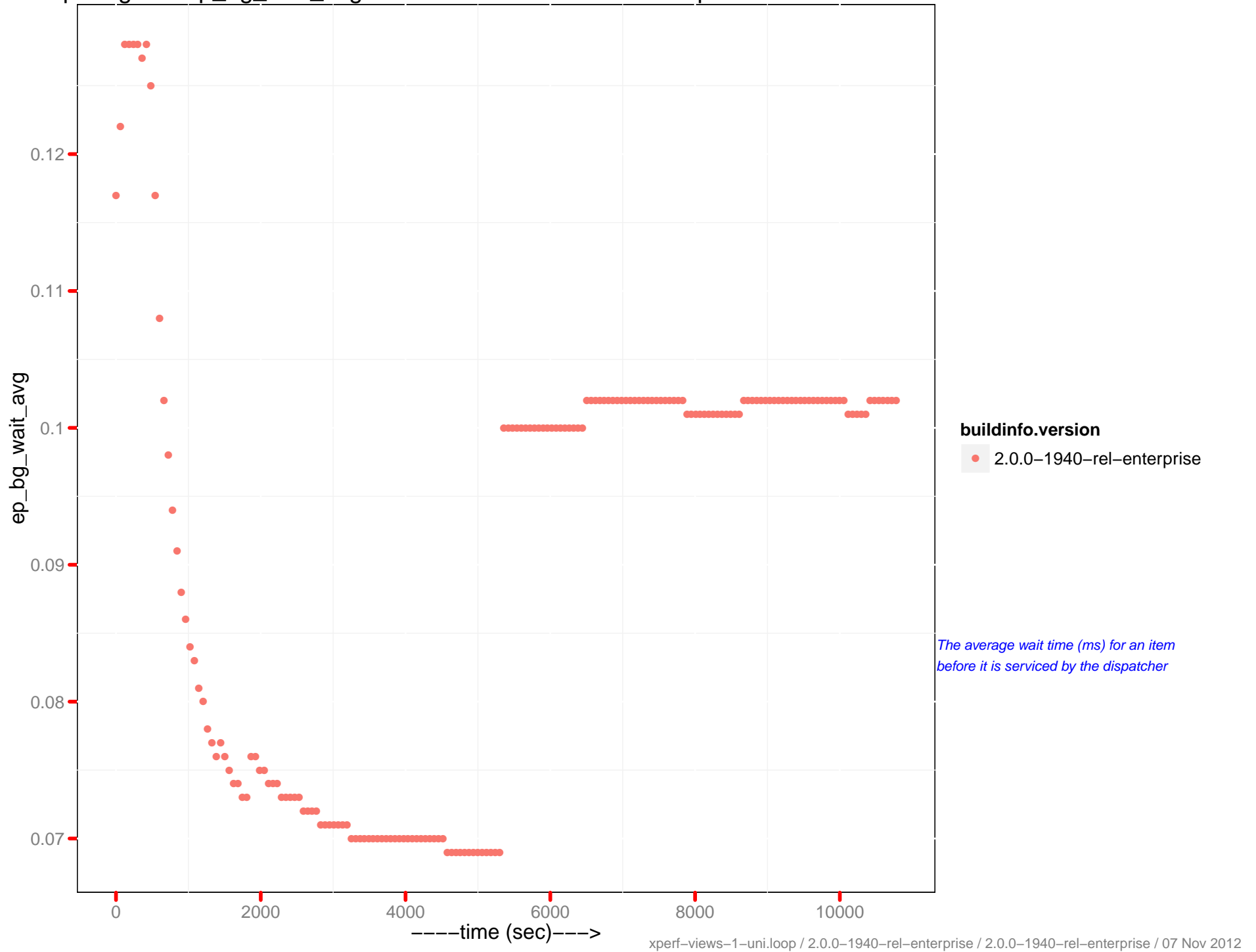
ep-engine : ep_bg_wait_avg - ec2-23-22-246-165.compute-1.amazonaws.com



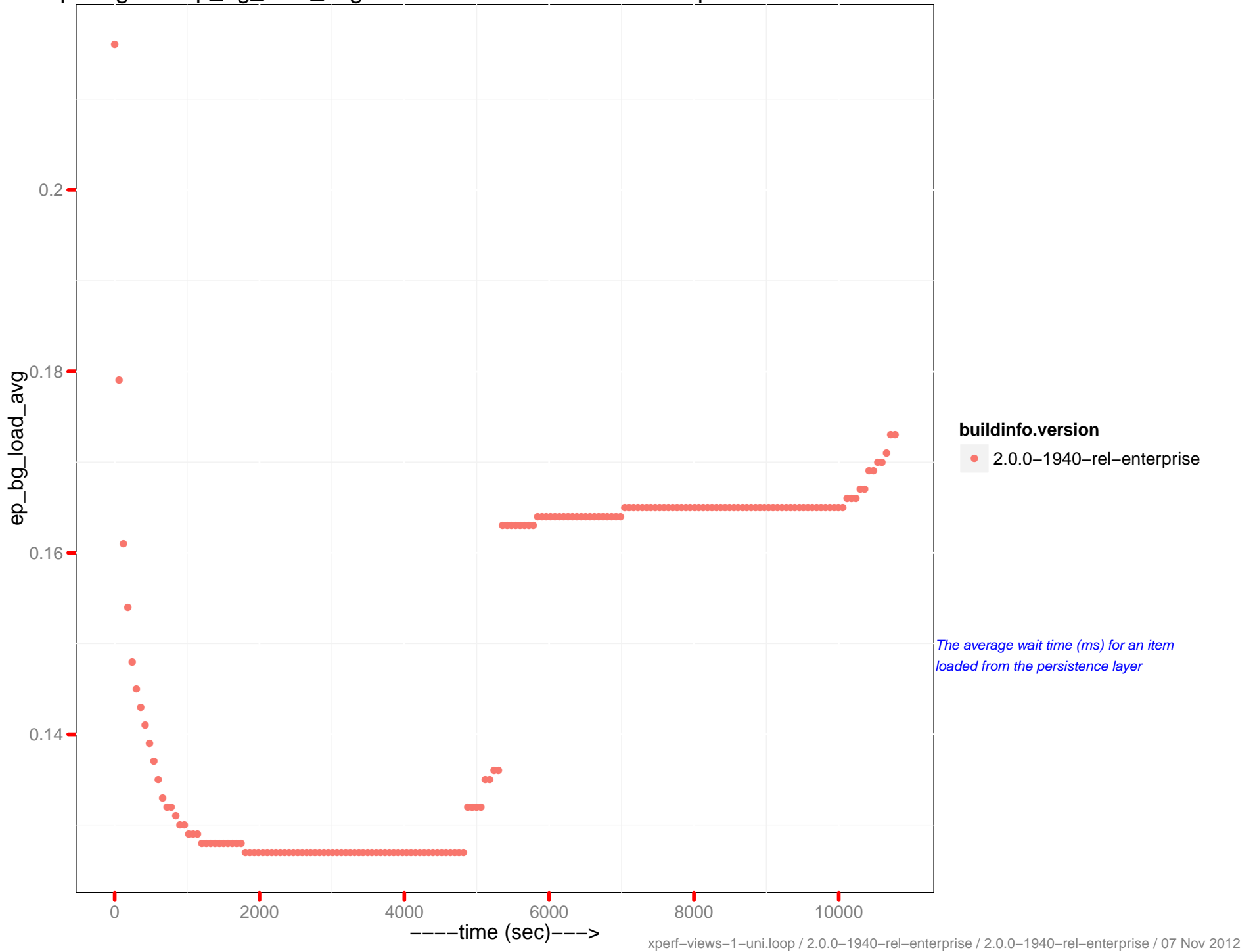
ep-engine : ep_bg_load_avg - ec2-23-22-246-165.compute-1.amazonaws.com



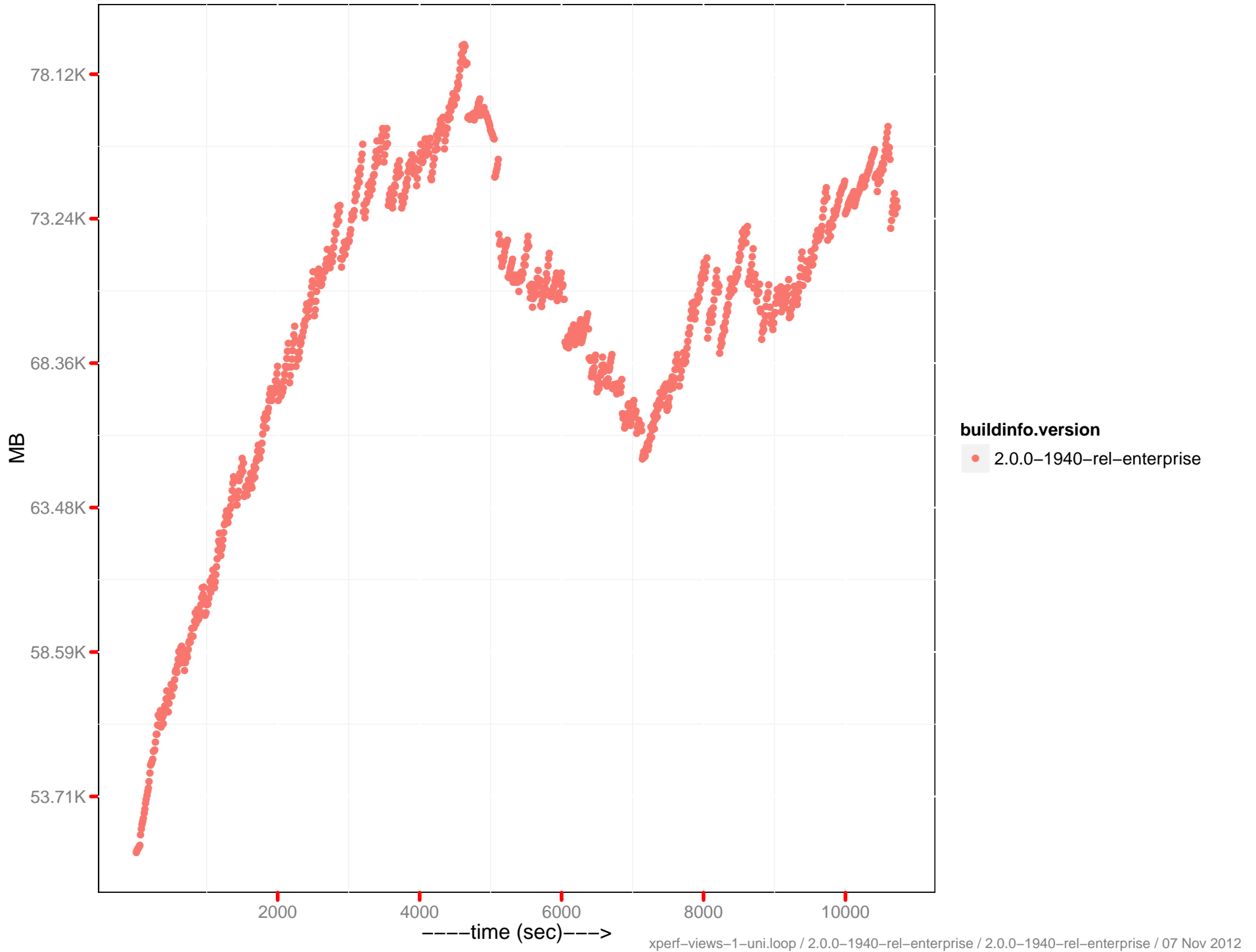
ep-engine : ep_bg_wait_avg - ec2-54-242-190-218.compute-1.amazonaws.com



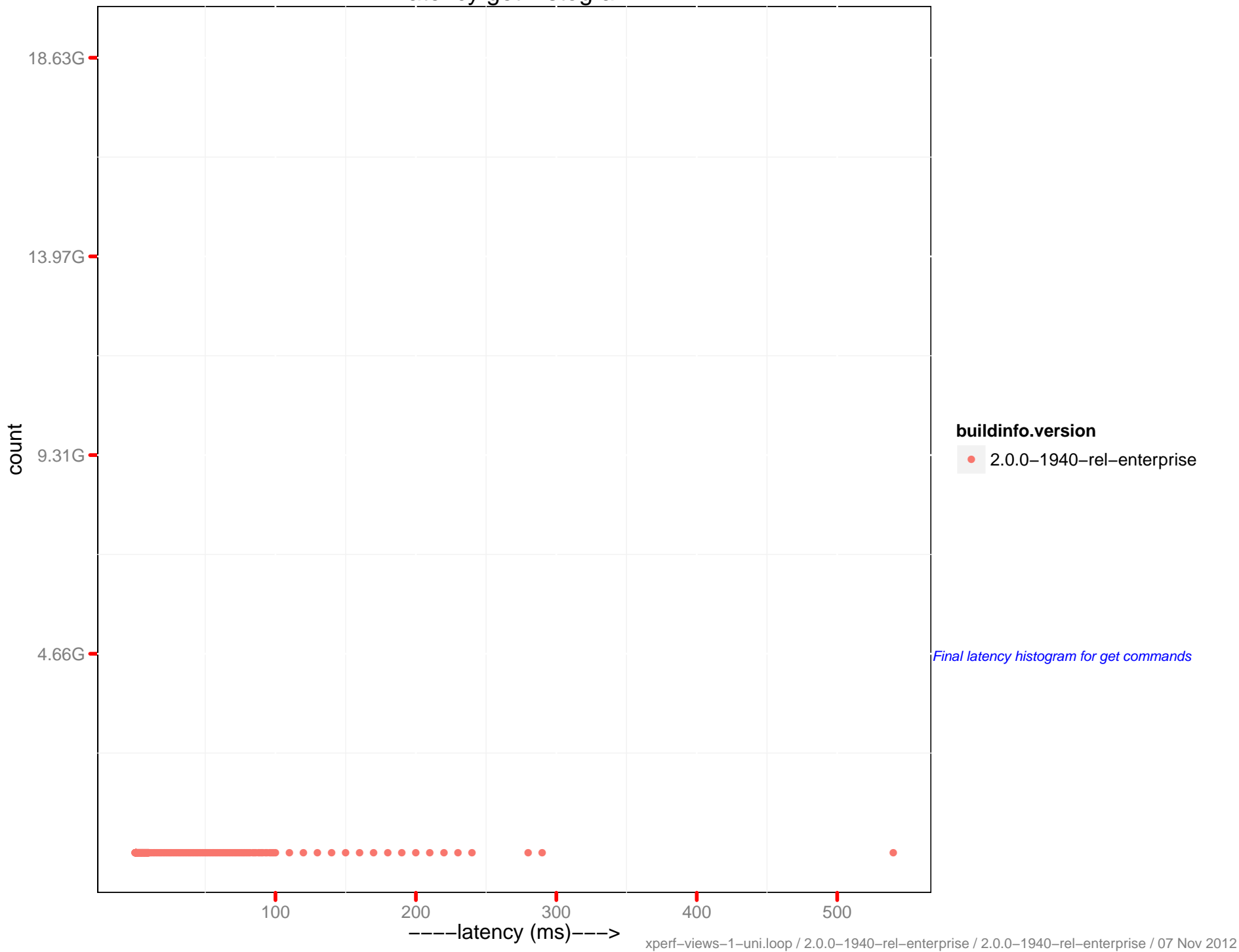
ep-engine : ep_bg_load_avg - ec2-54-242-190-218.compute-1.amazonaws.com



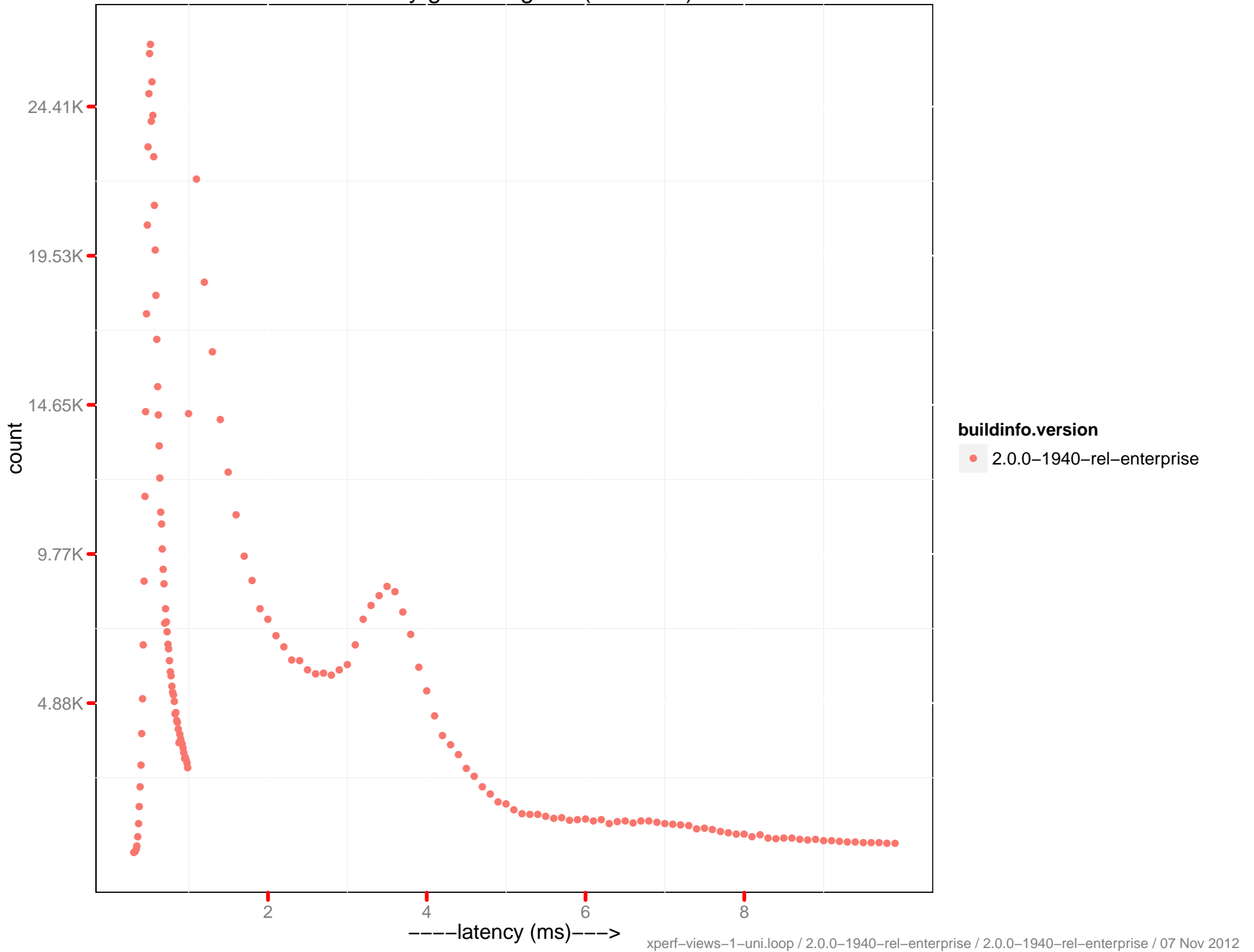
Data disk size



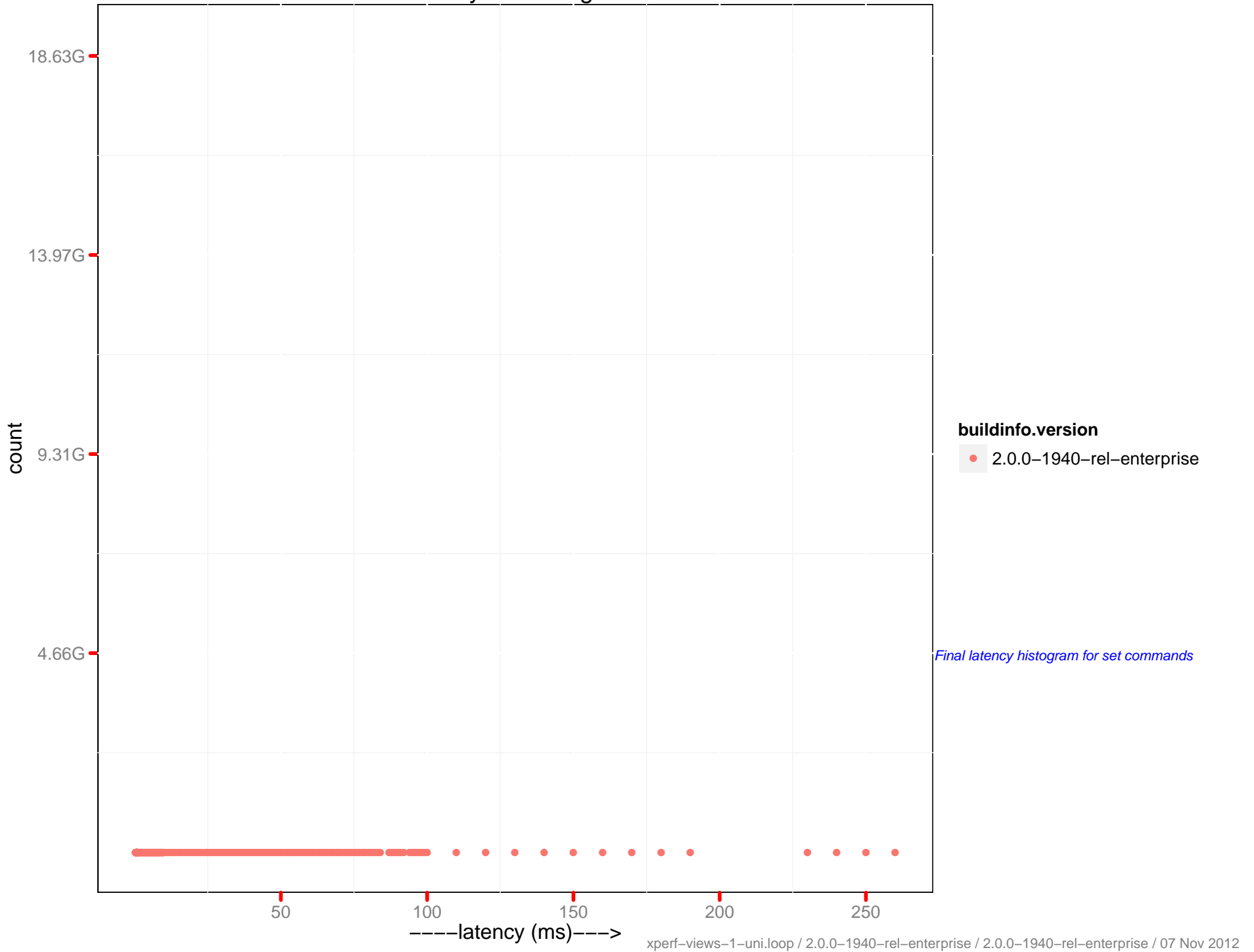
Latency get histogram



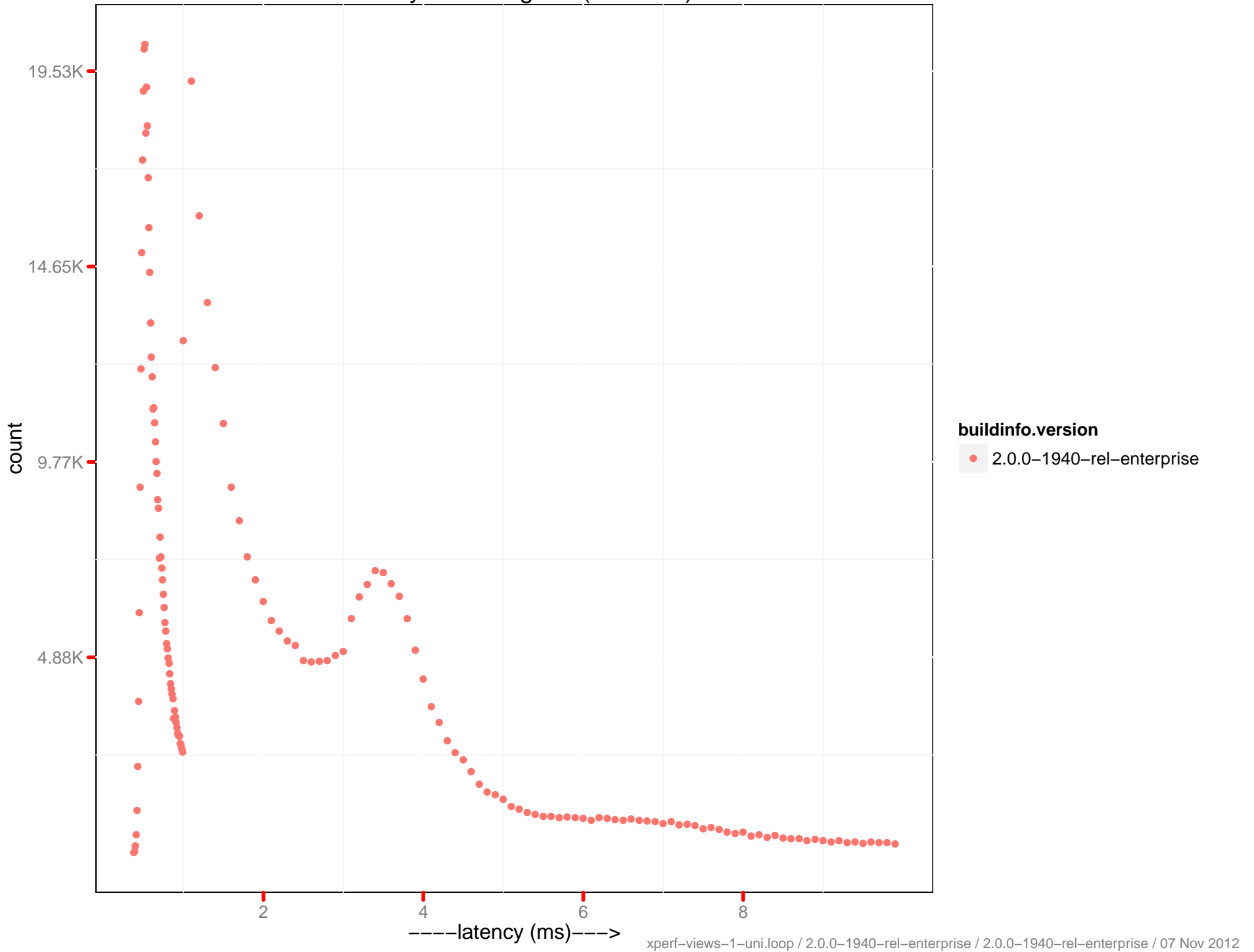
Latency get histogram (0-10 ms)



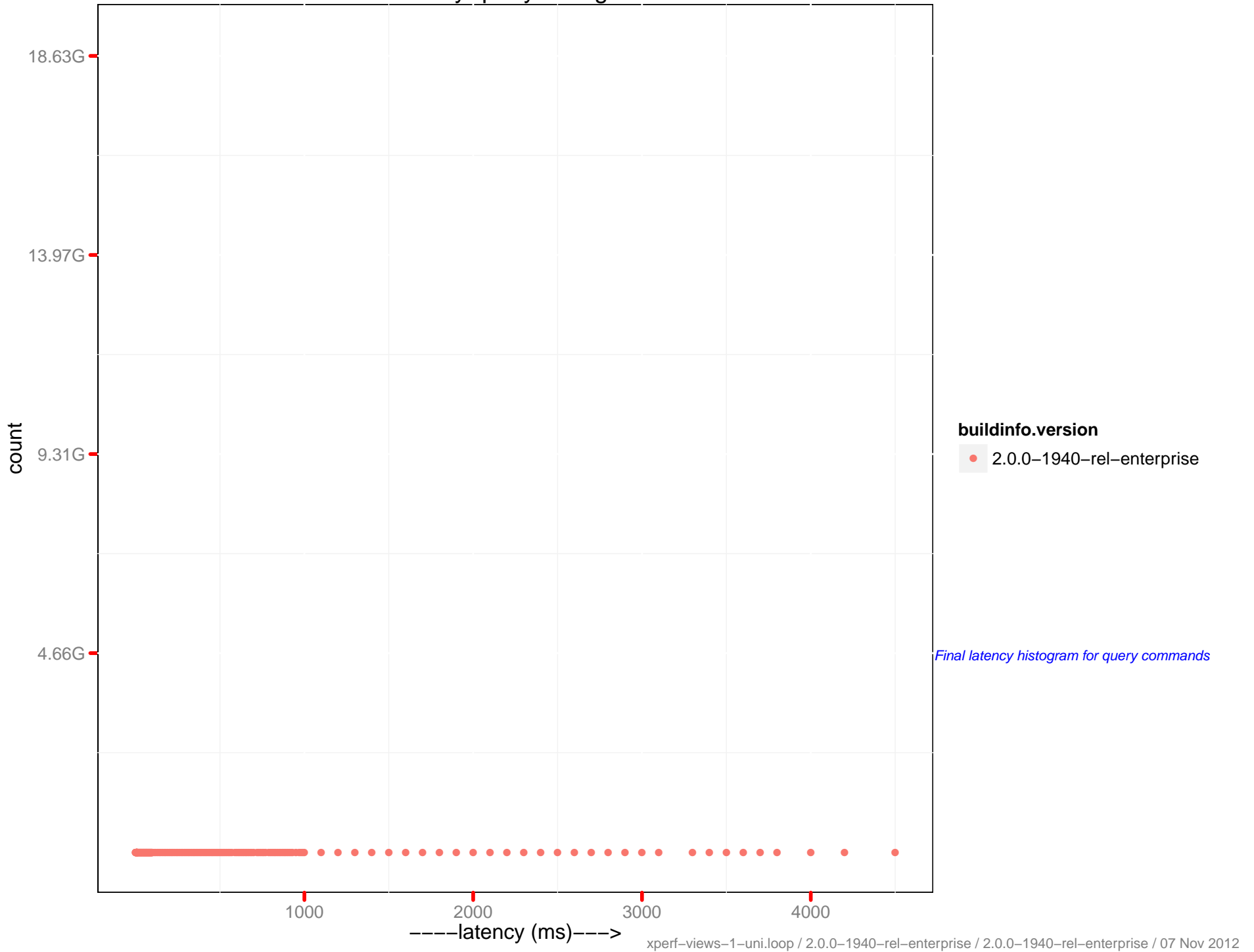
Latency set histogram



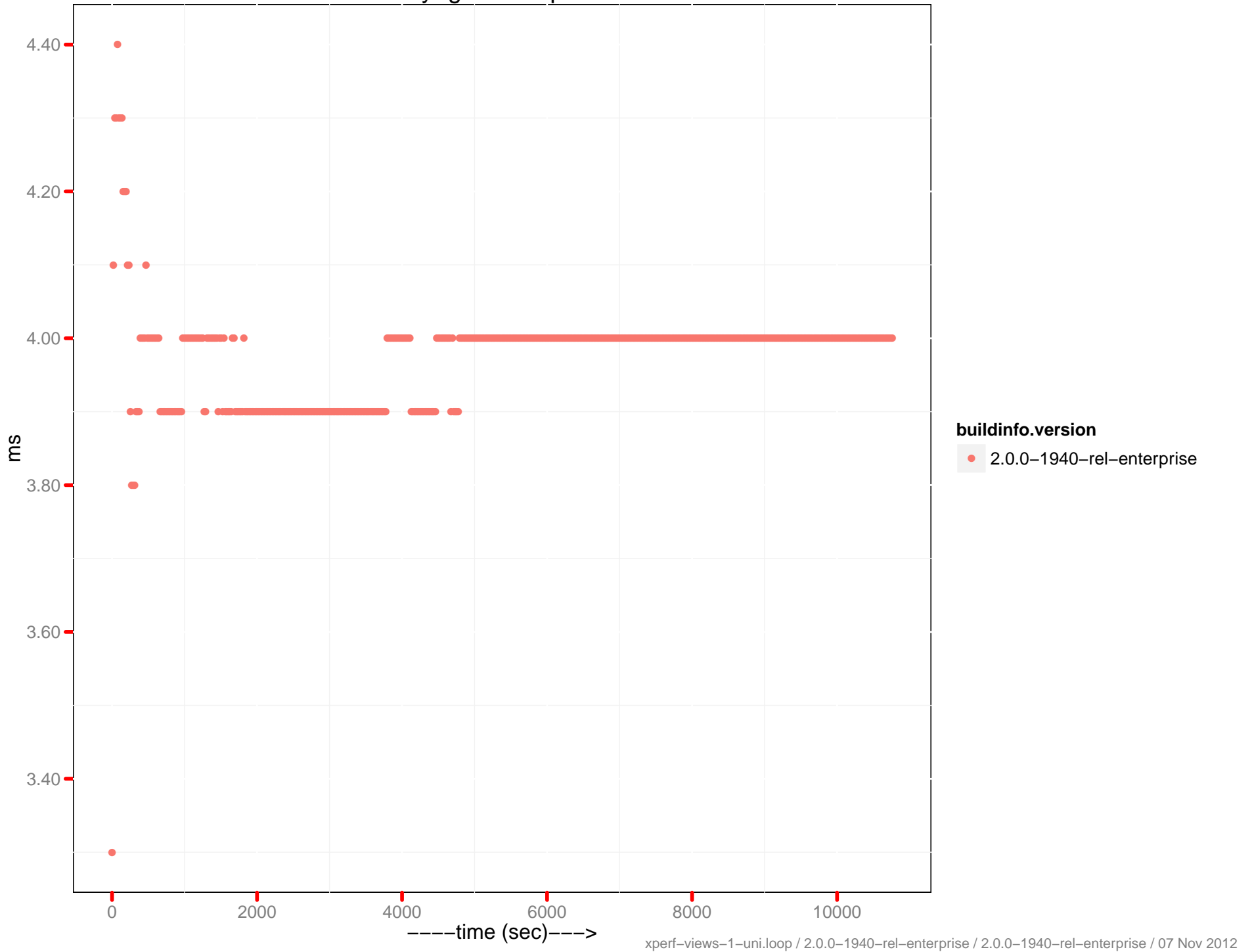
Latency set histogram (0–10 ms)



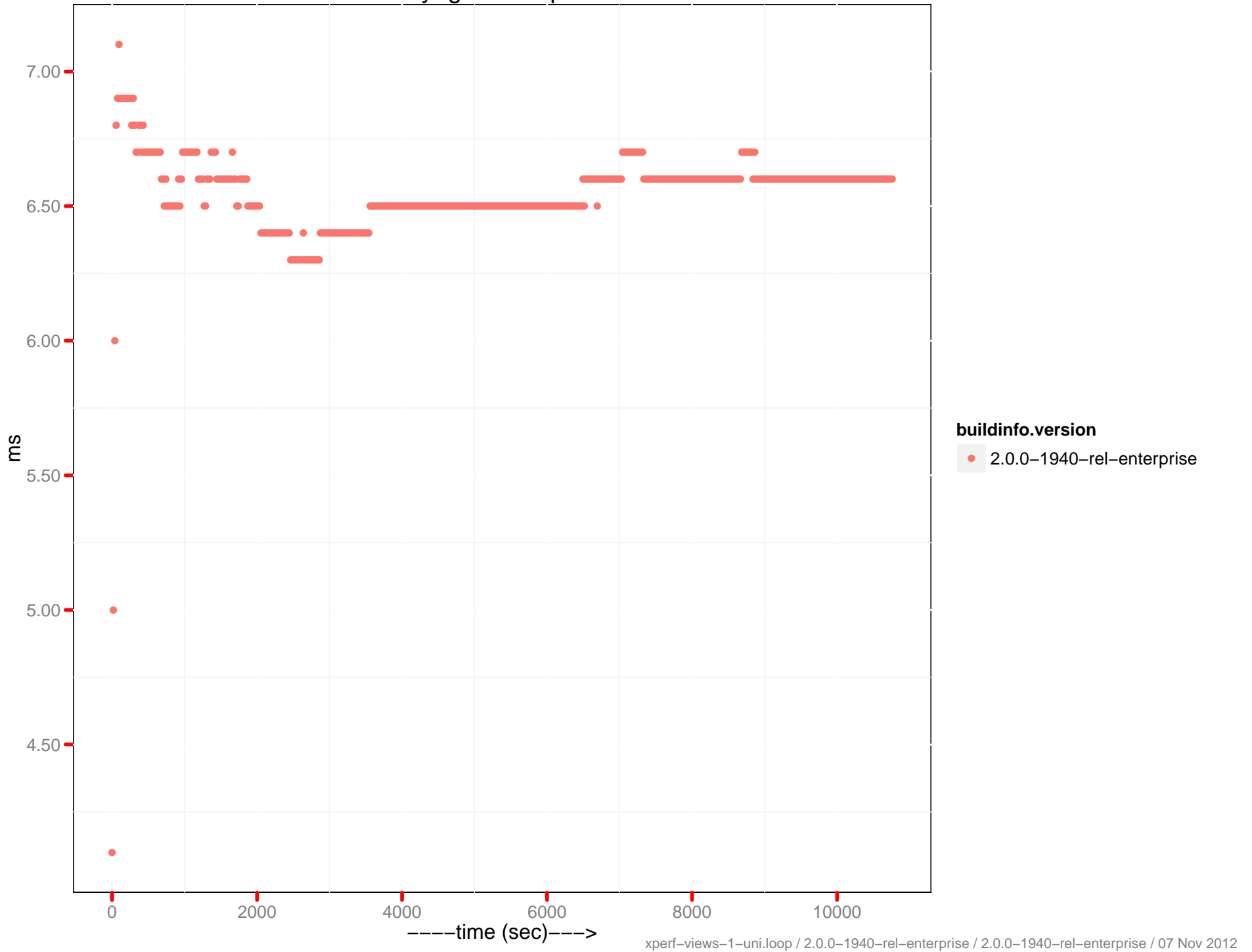
Latency query histogram



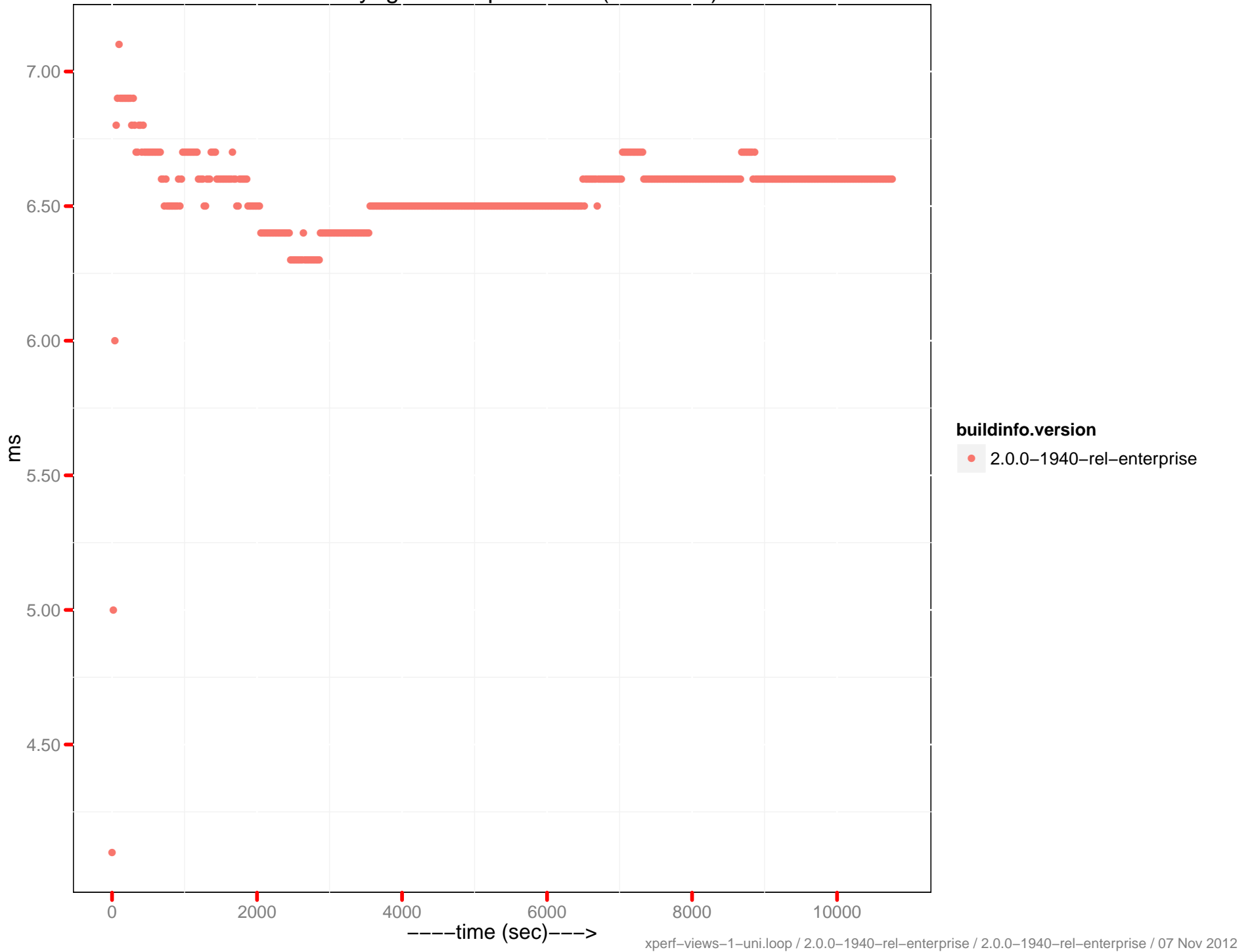
Latency-get 90th percentile



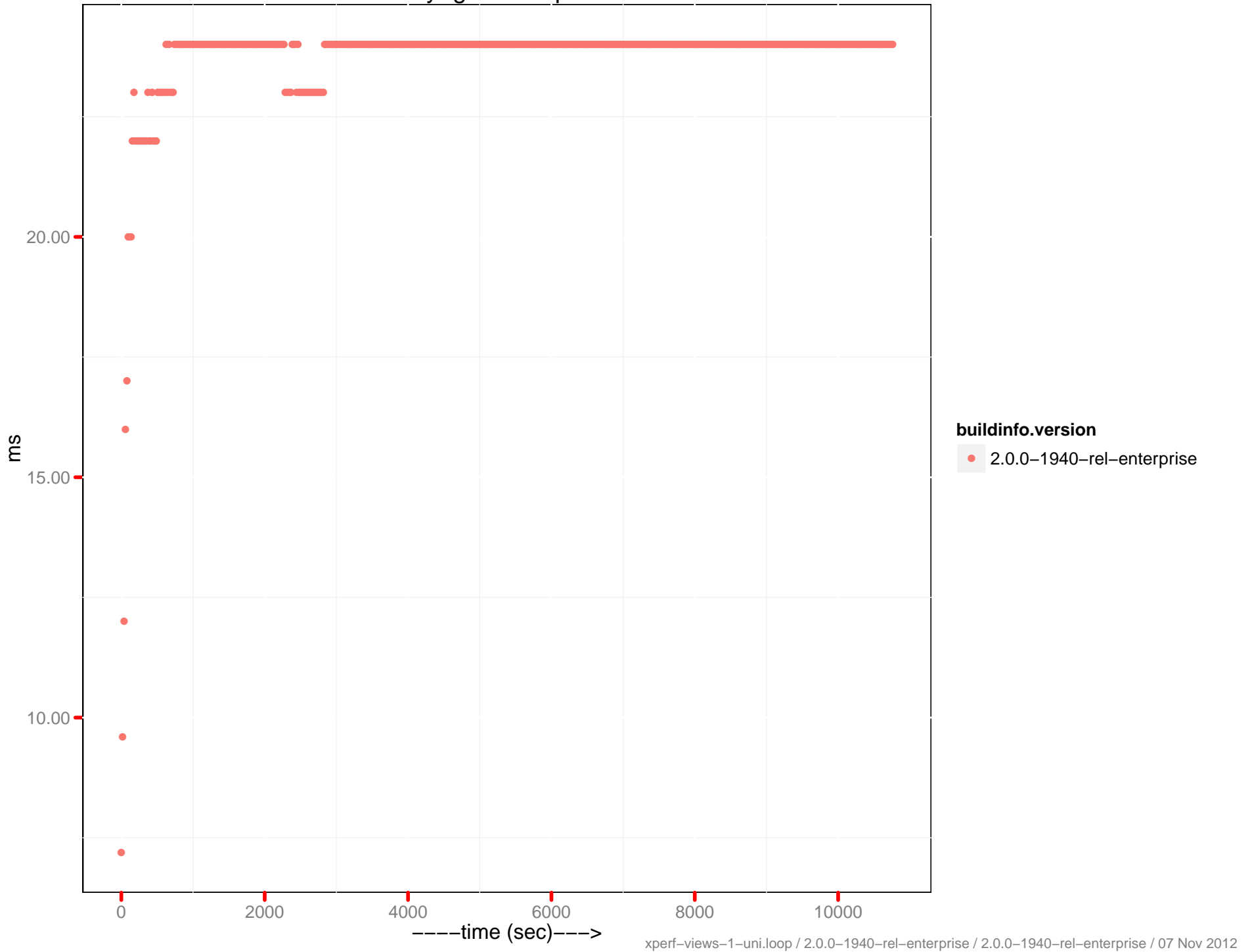
Latency-get 95th percentile



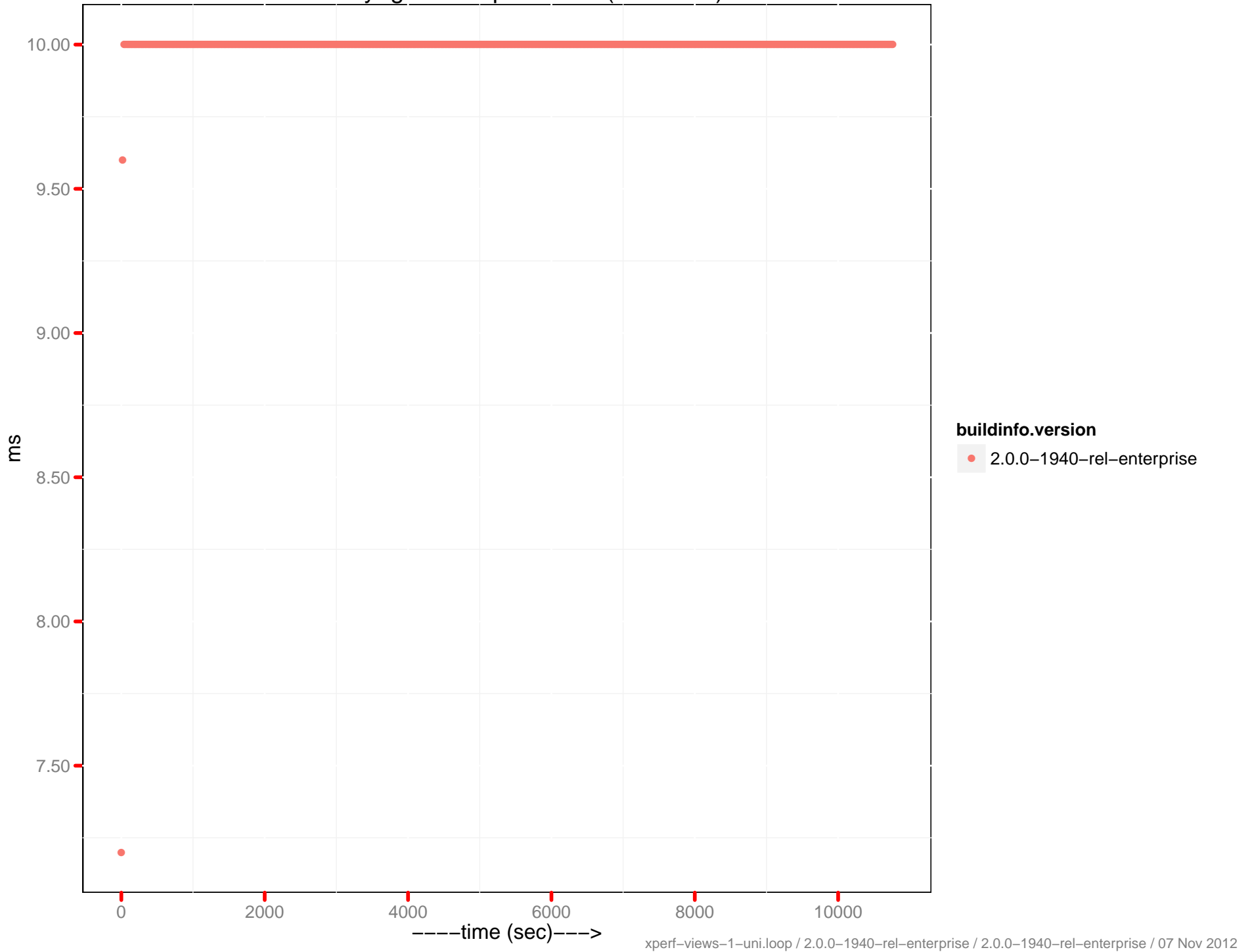
Latency-get 95th percentile (0 - 10ms)



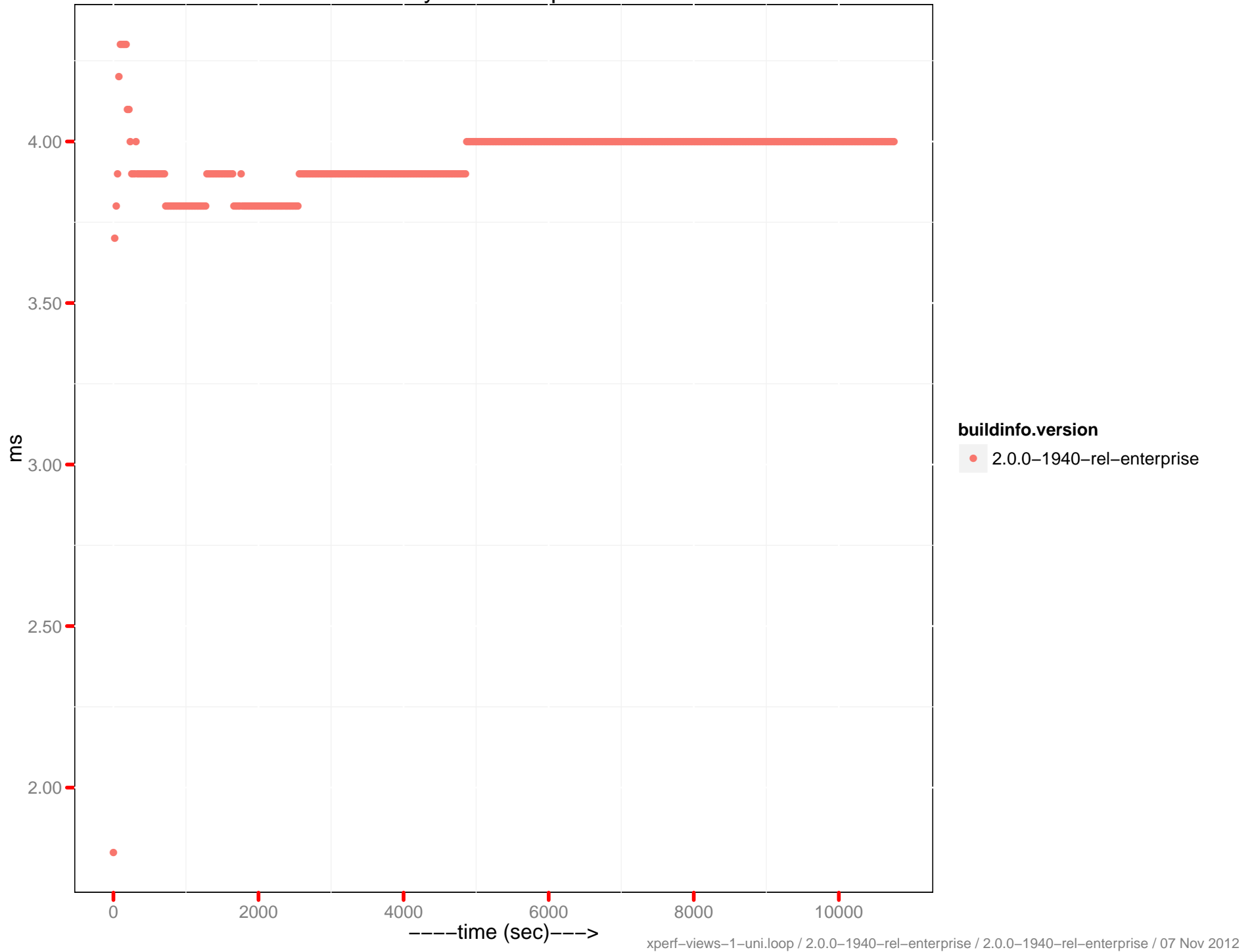
Latency-get 99th percentile



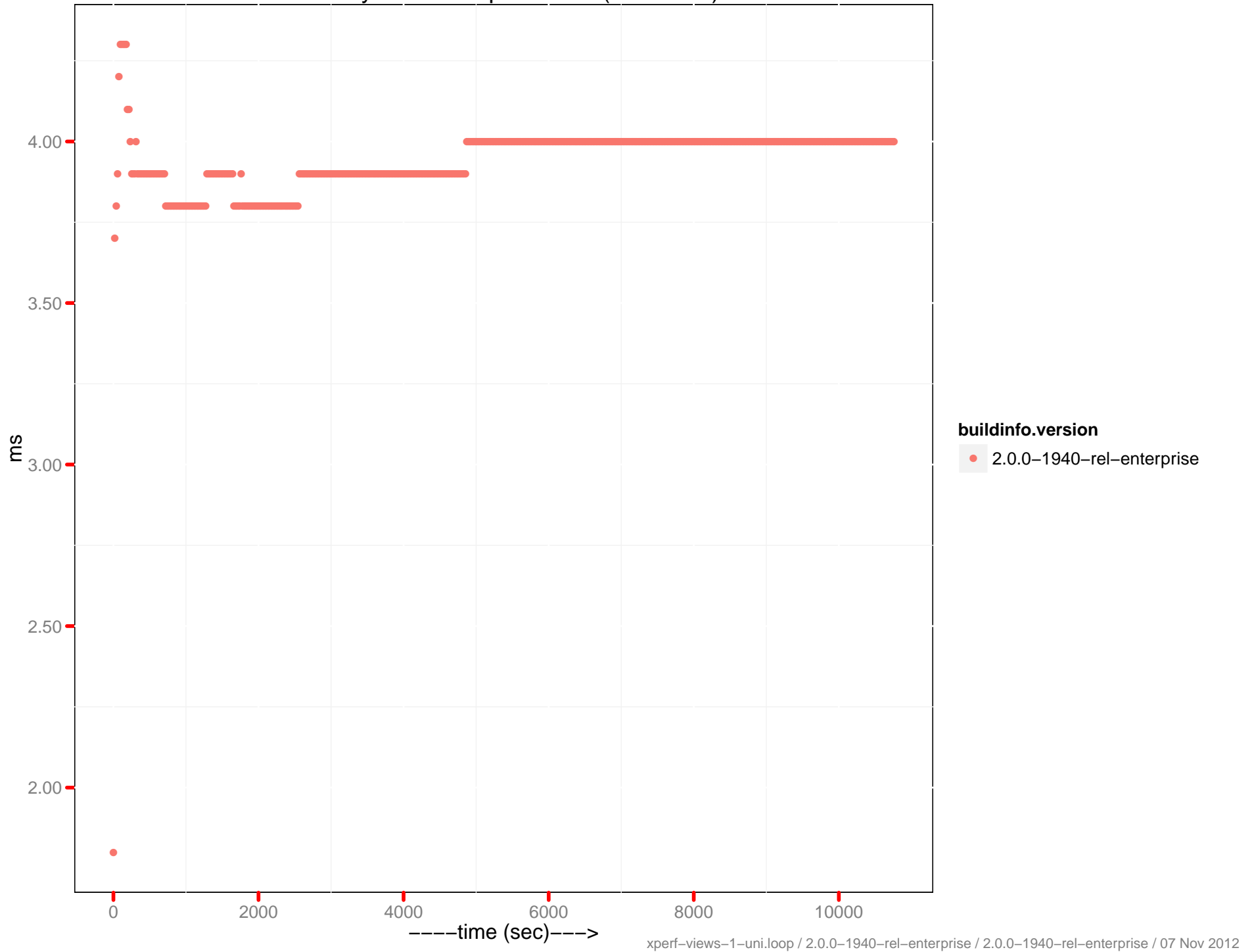
Latency-get 99th percentile (0 - 10ms)



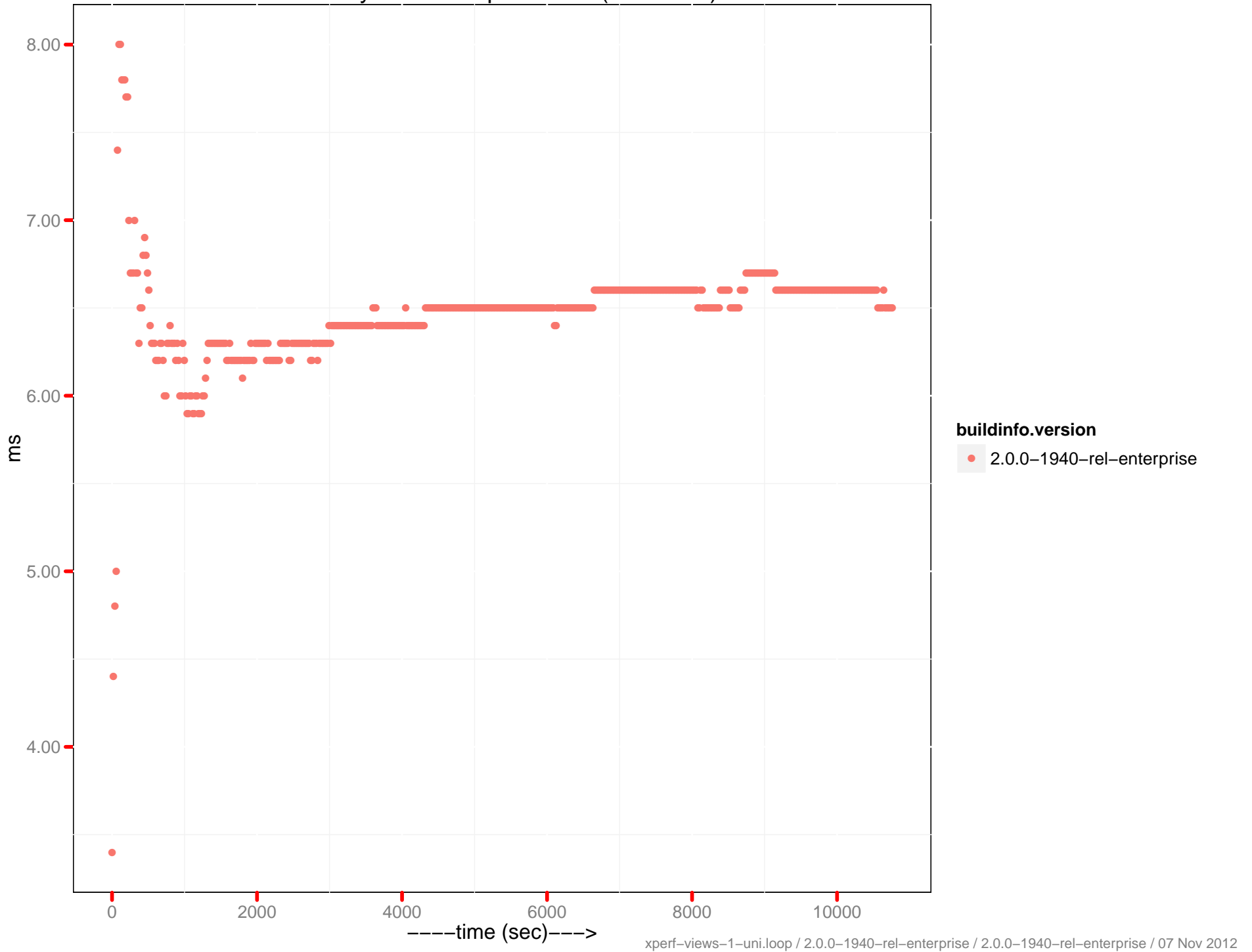
Latency-set 90th percentile



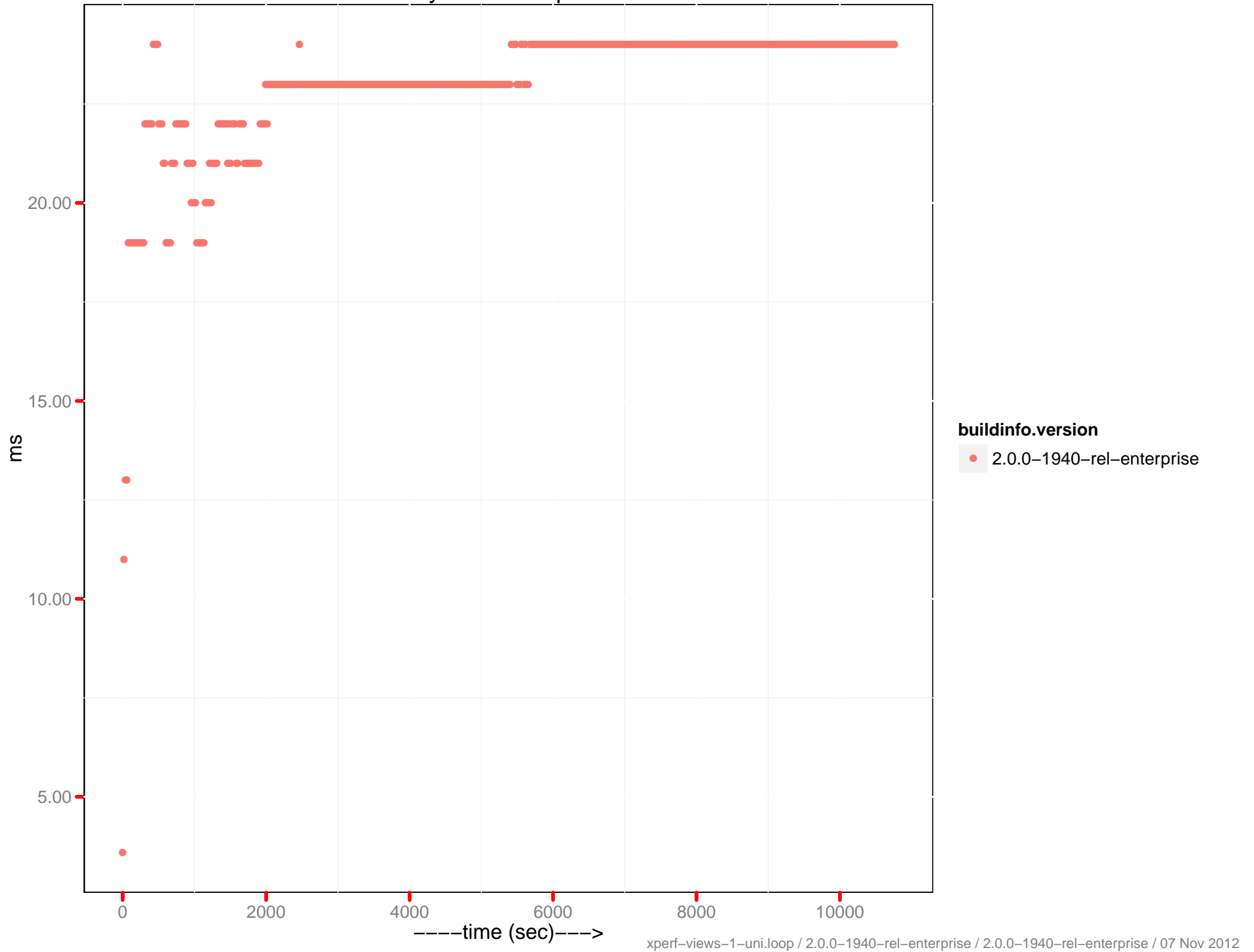
Latency-set 90th percentile (0 - 10ms)



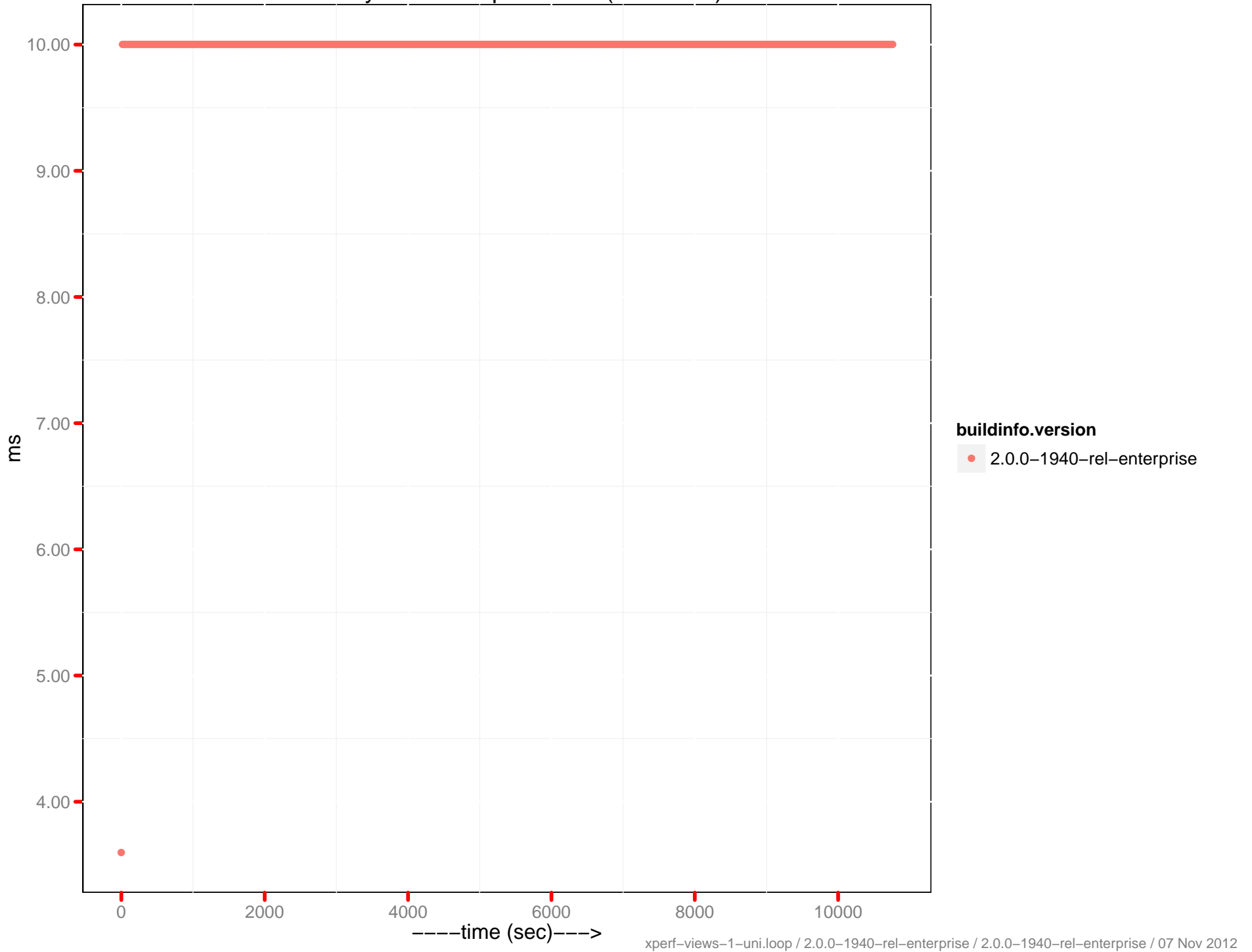
Latency-set 95th percentile (0 - 10ms)



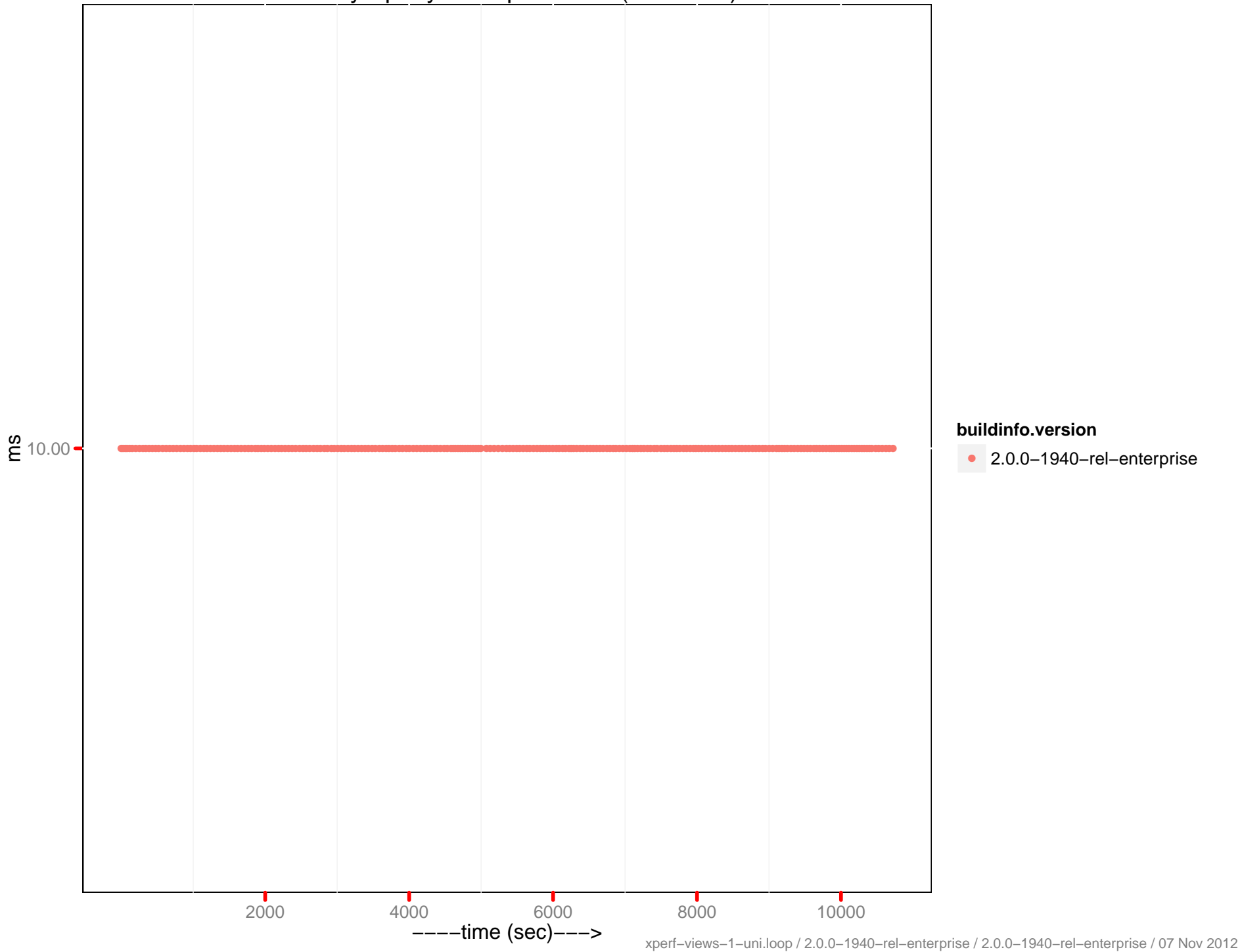
Latency-set 99th percentile



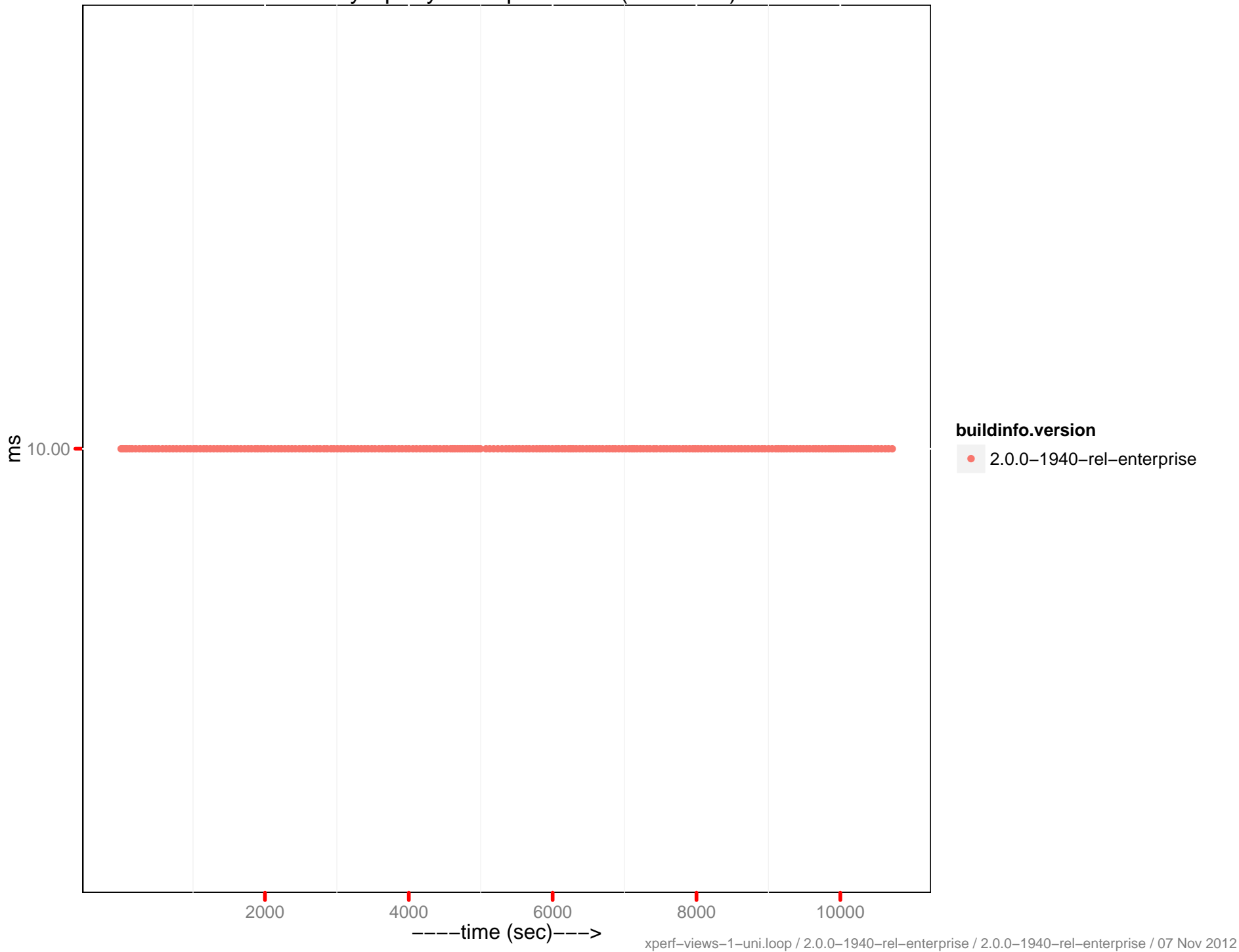
Latency-set 99th percentile (0 – 10ms)



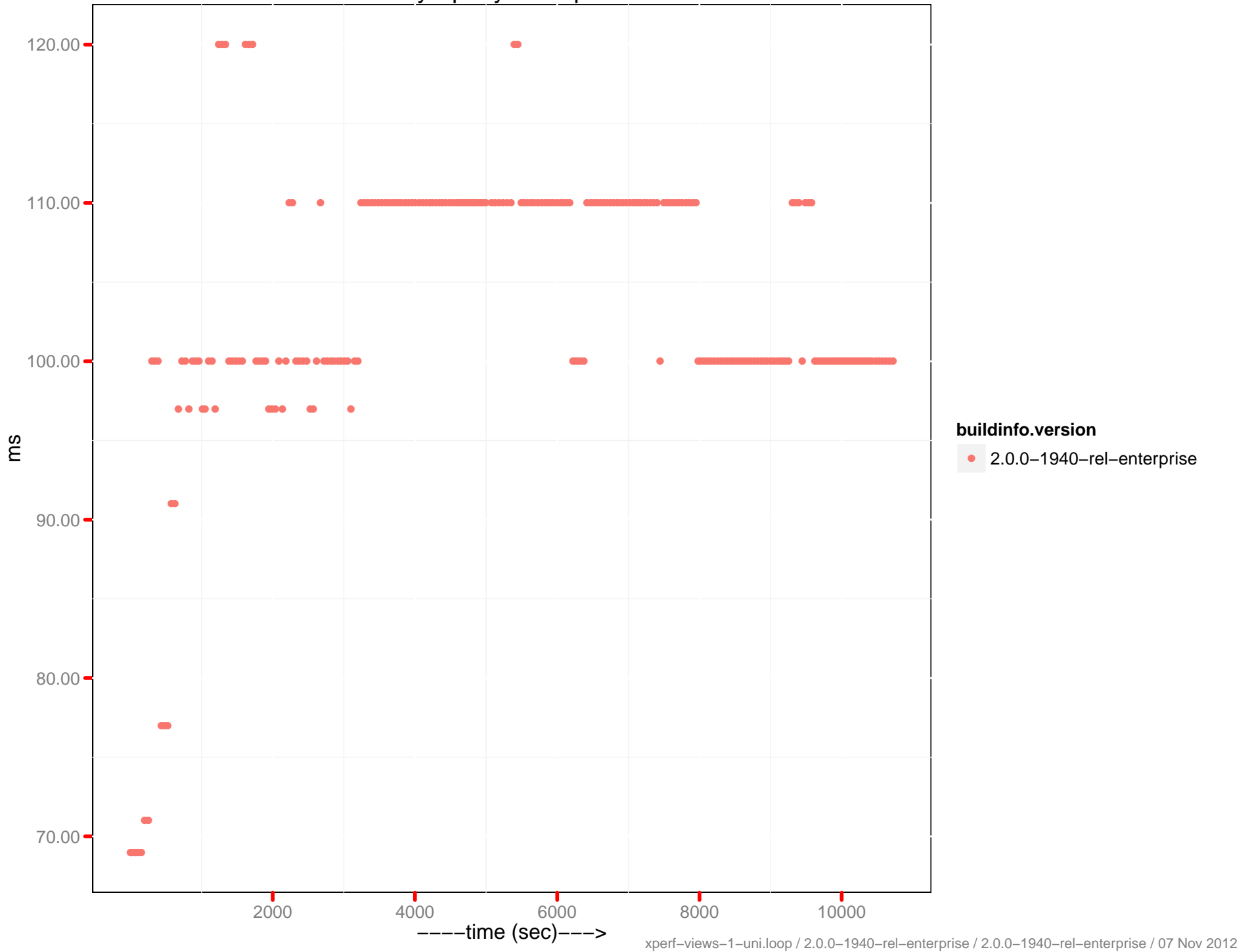
Latency-query 80th percentile (0 - 10ms)



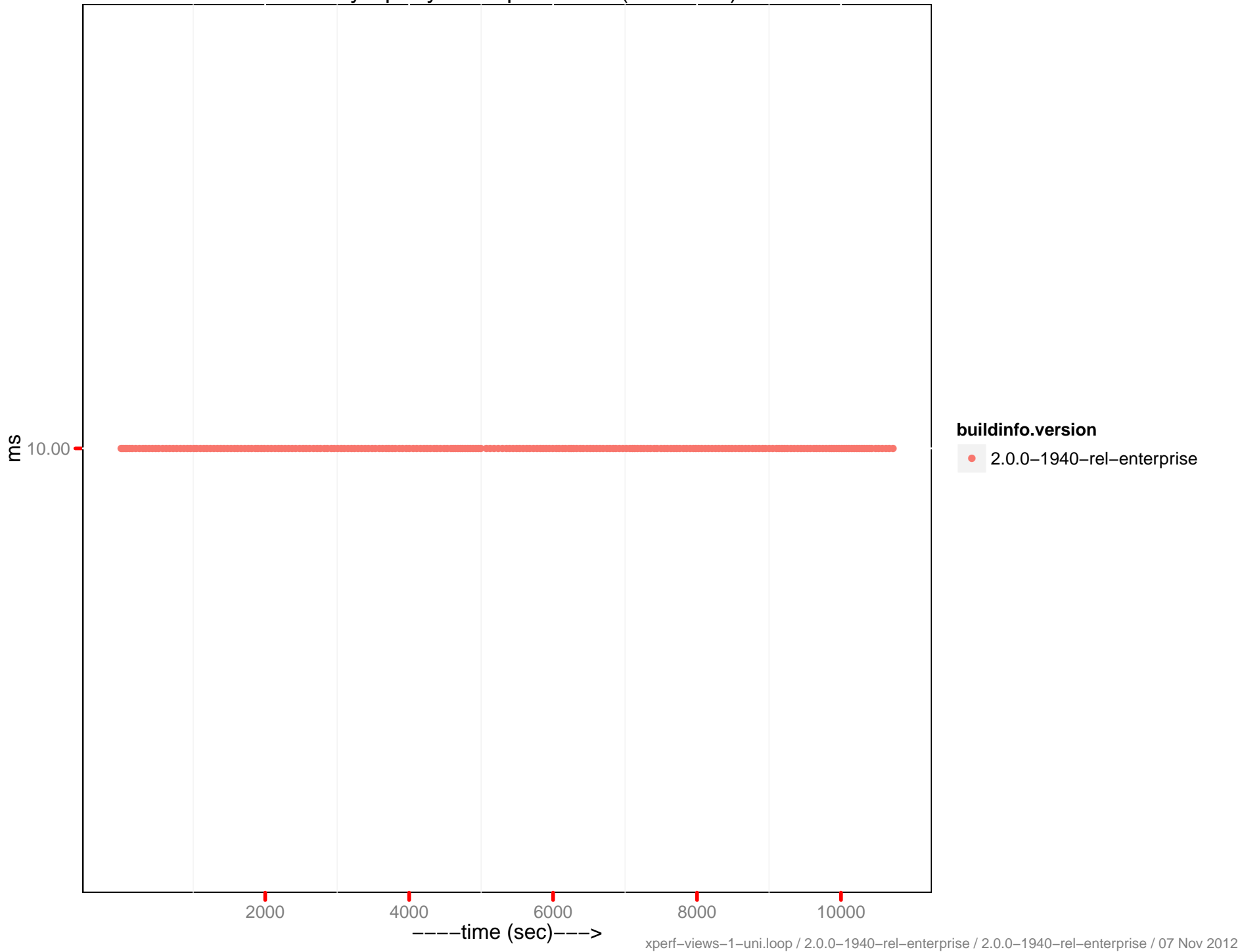
Latency-query 90th percentile (0 - 10ms)



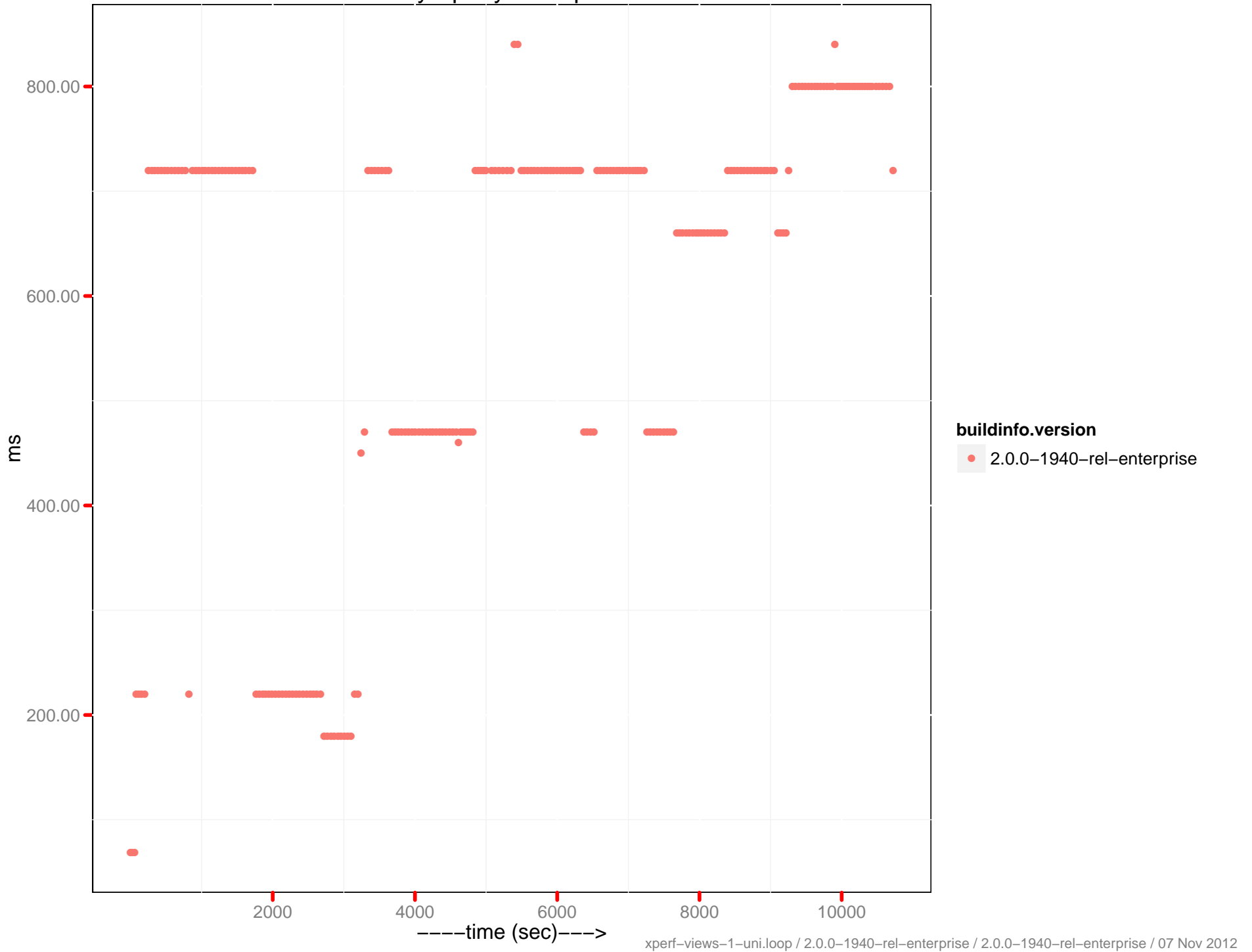
Latency-query 95th percentile



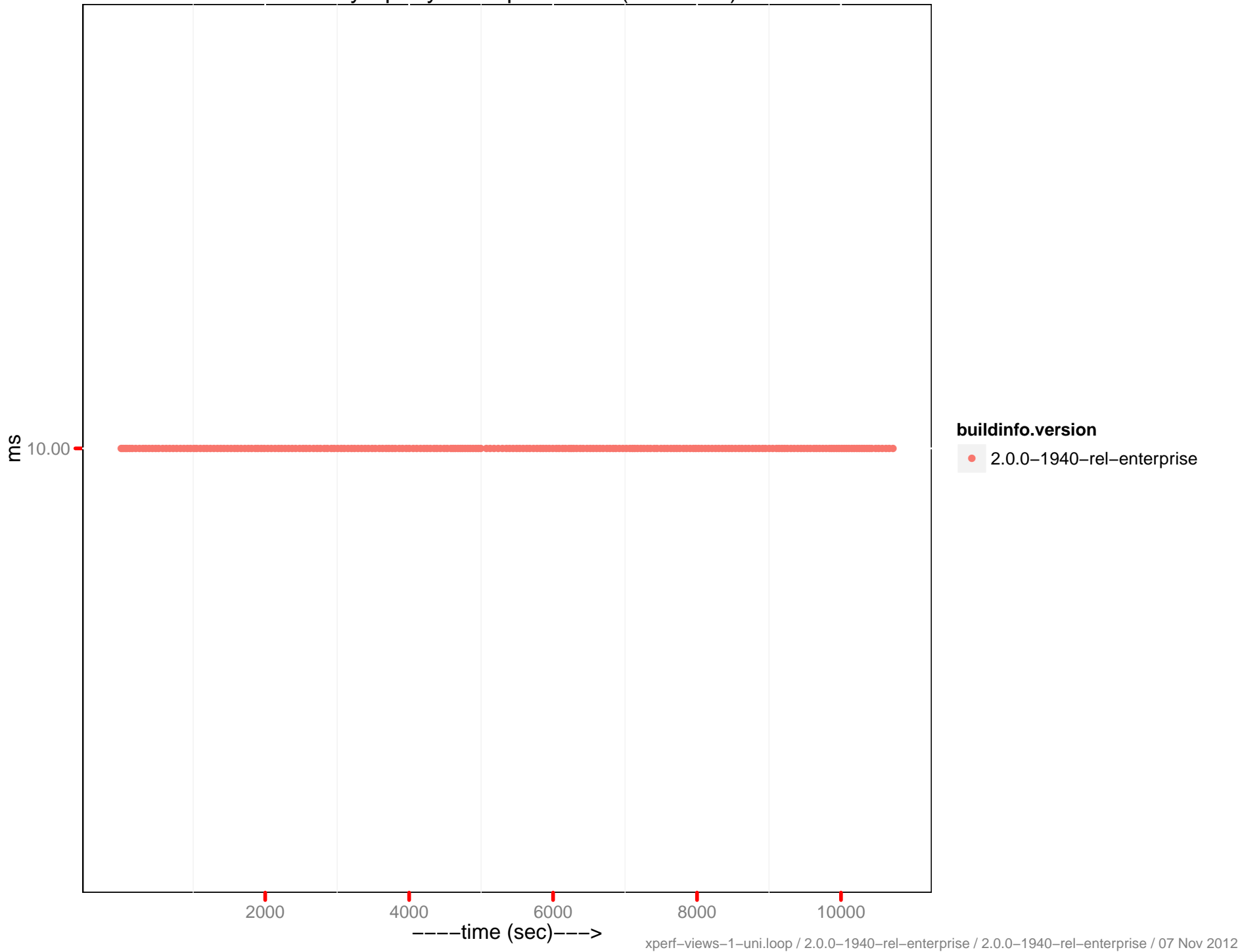
Latency-query 95th percentile (0 - 10ms)



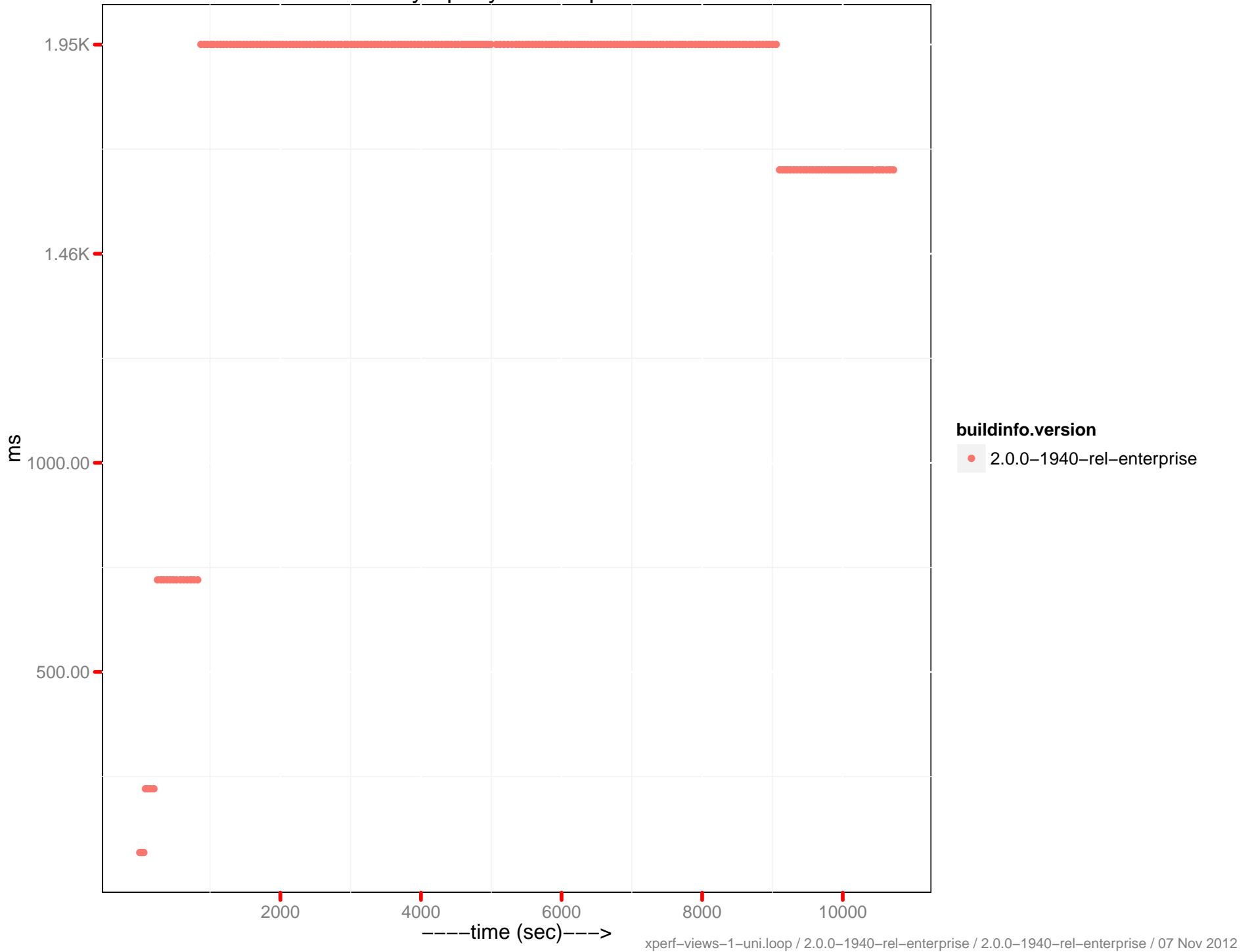
Latency-query 99th percentile



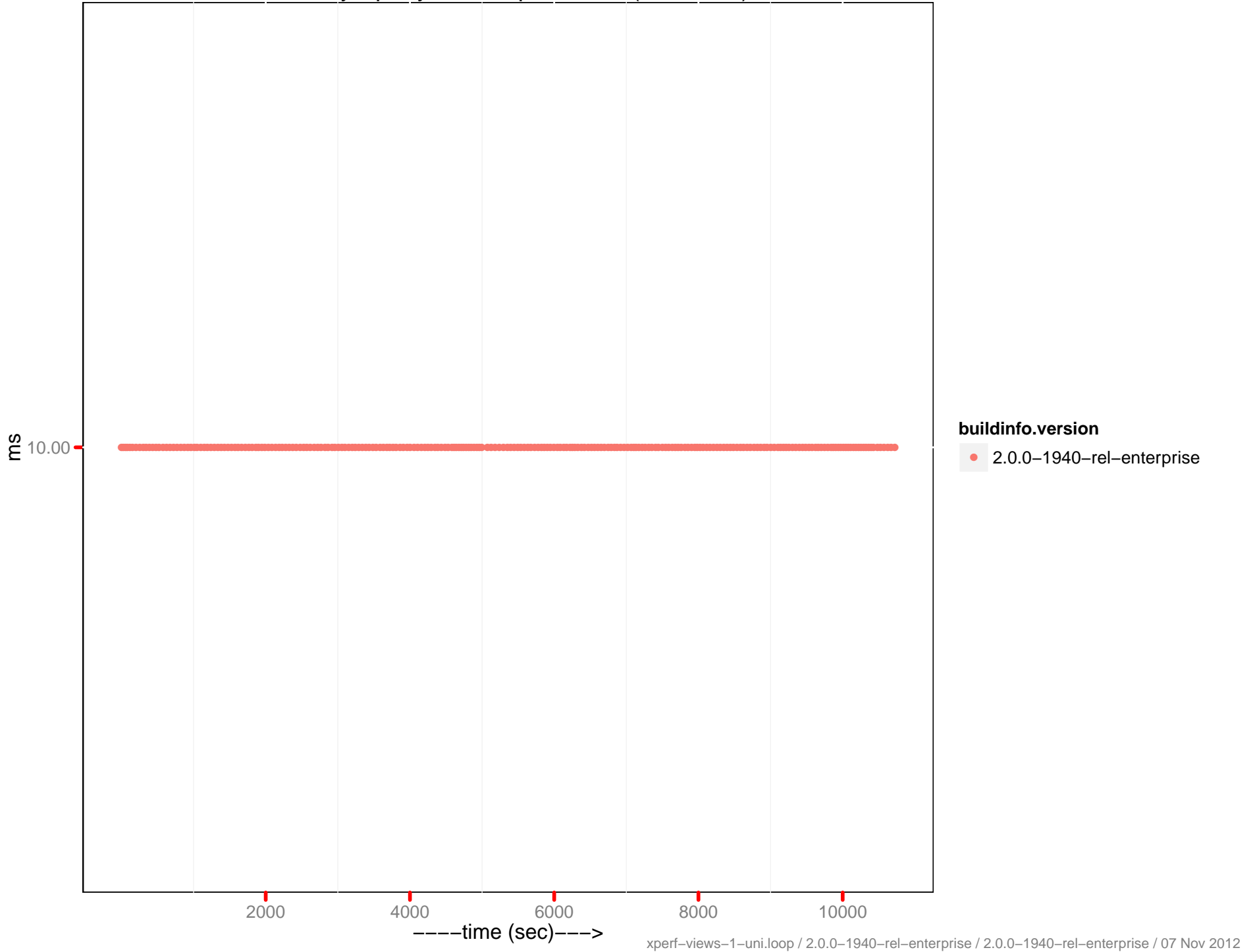
Latency-query 99th percentile (0 - 10ms)



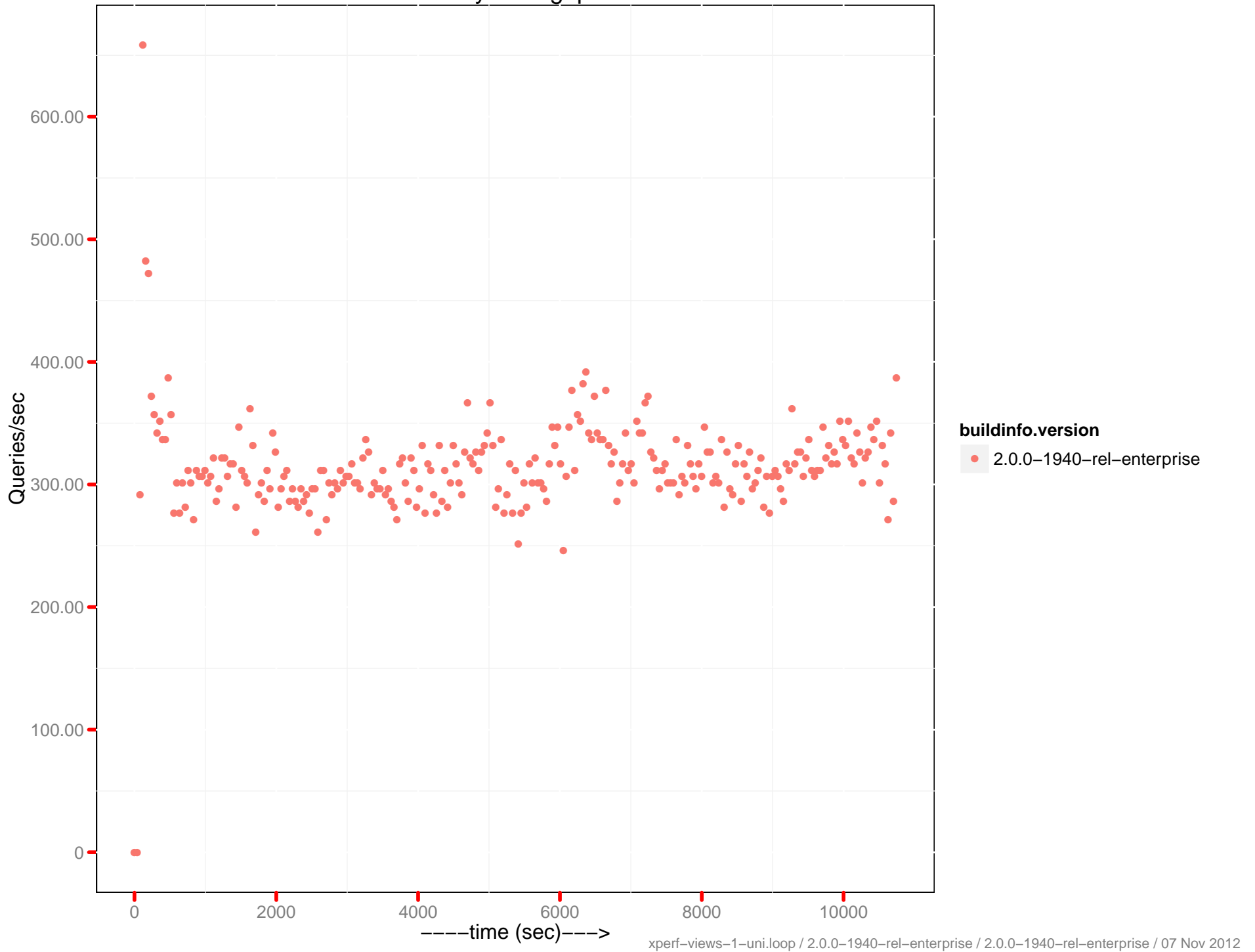
Latency-query 99.9th percentile



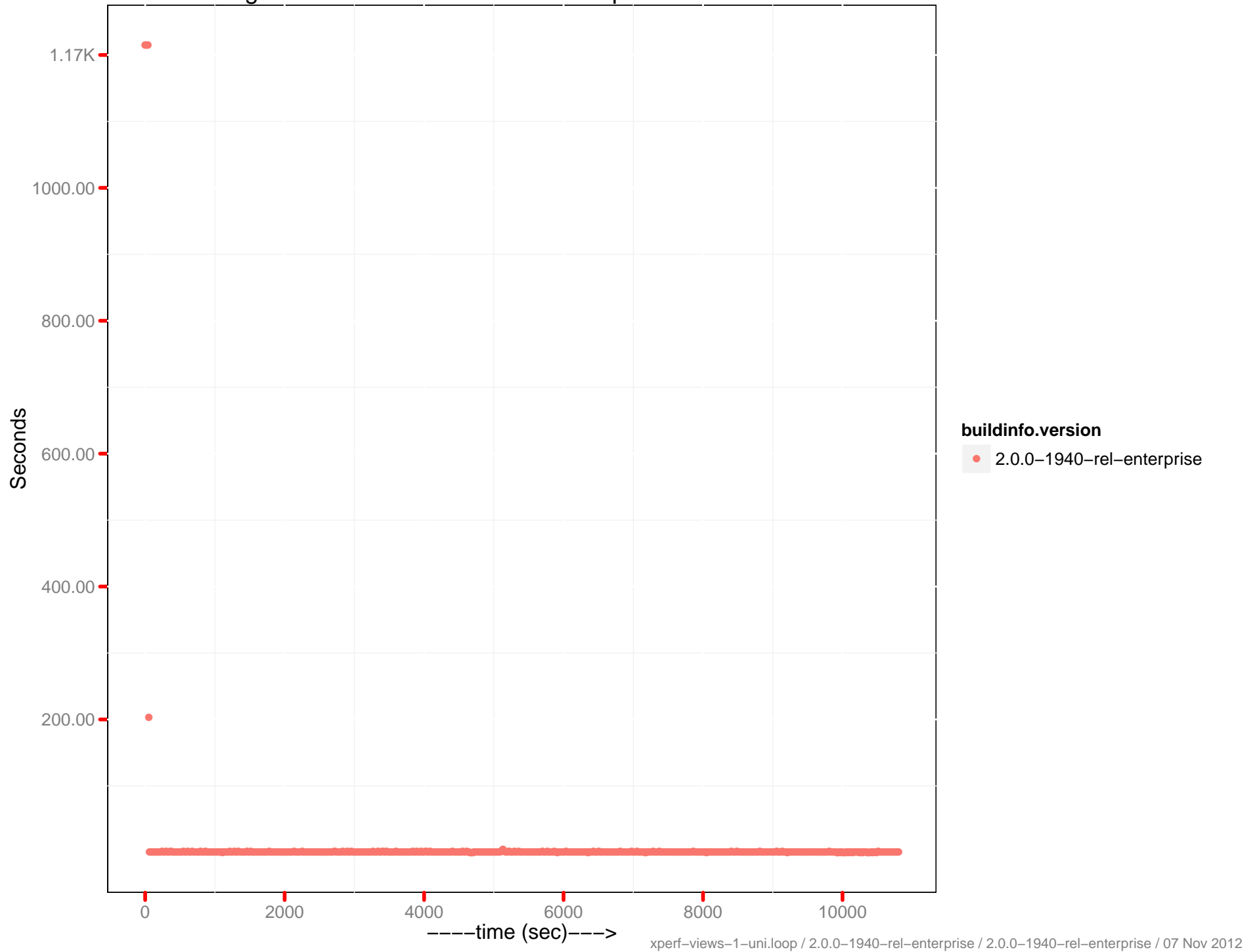
Latency-query 99.9th percentile (0 - 10ms)



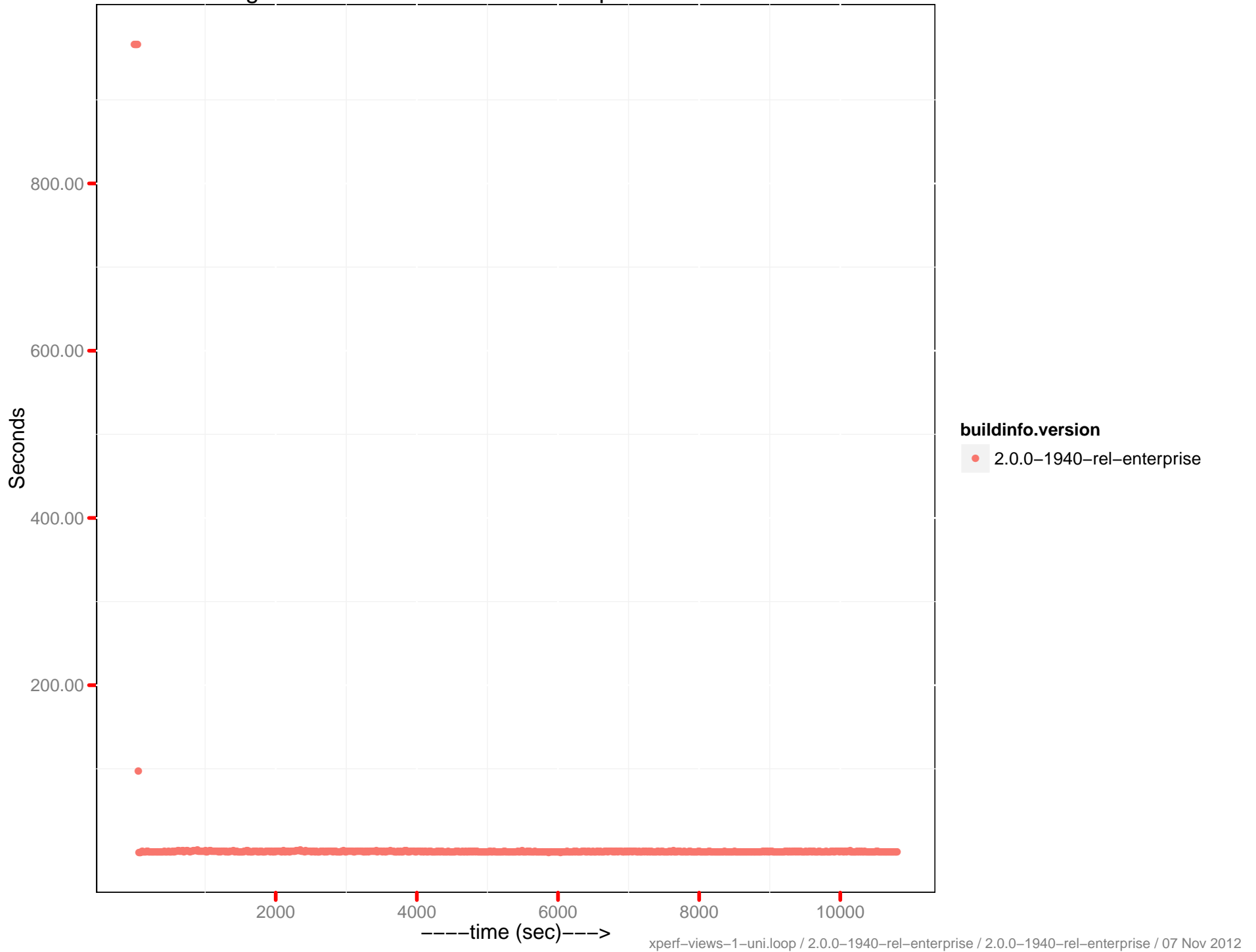
Query throughput



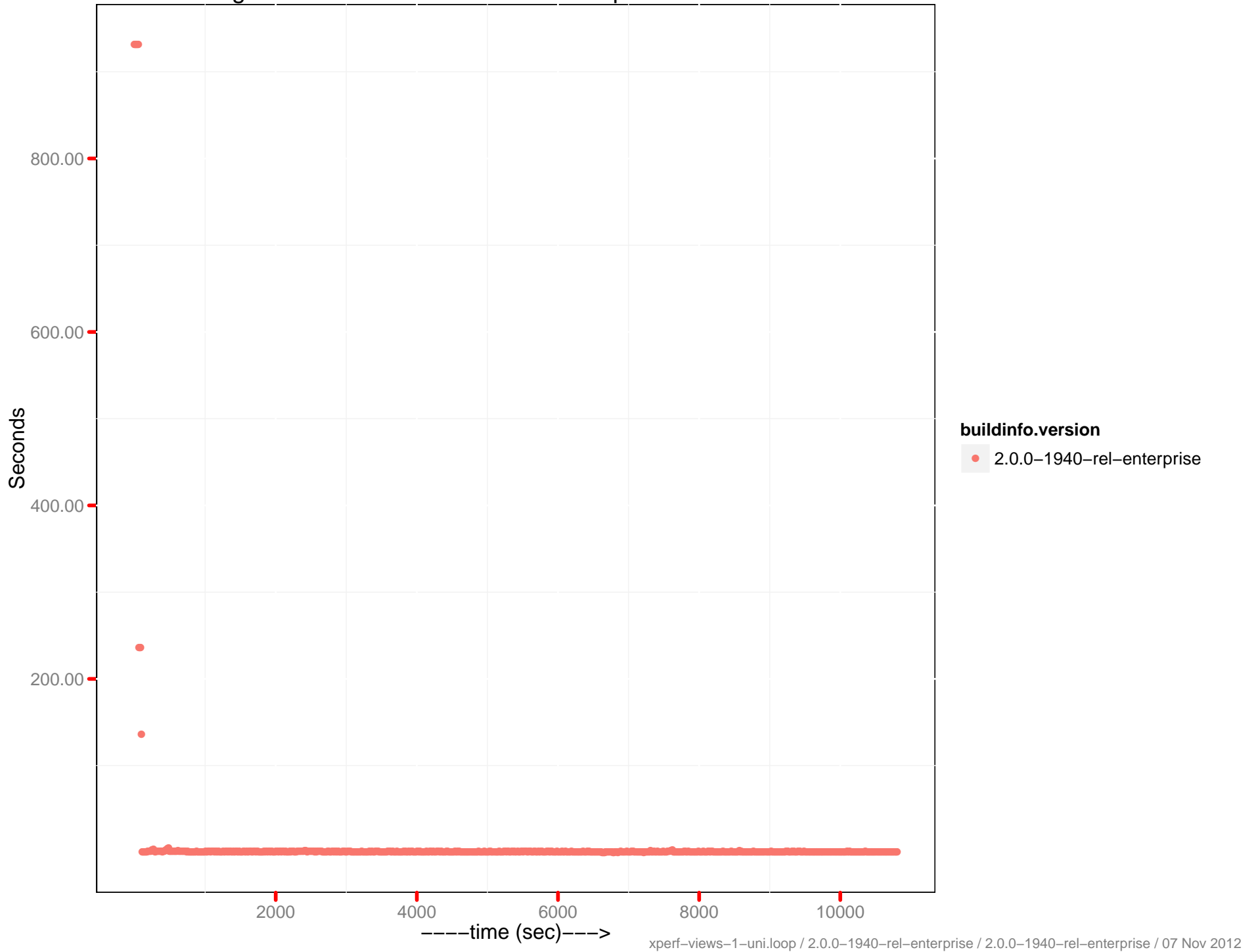
Indexing time – ec2-107-20-6-150.compute-1.amazonaws.com



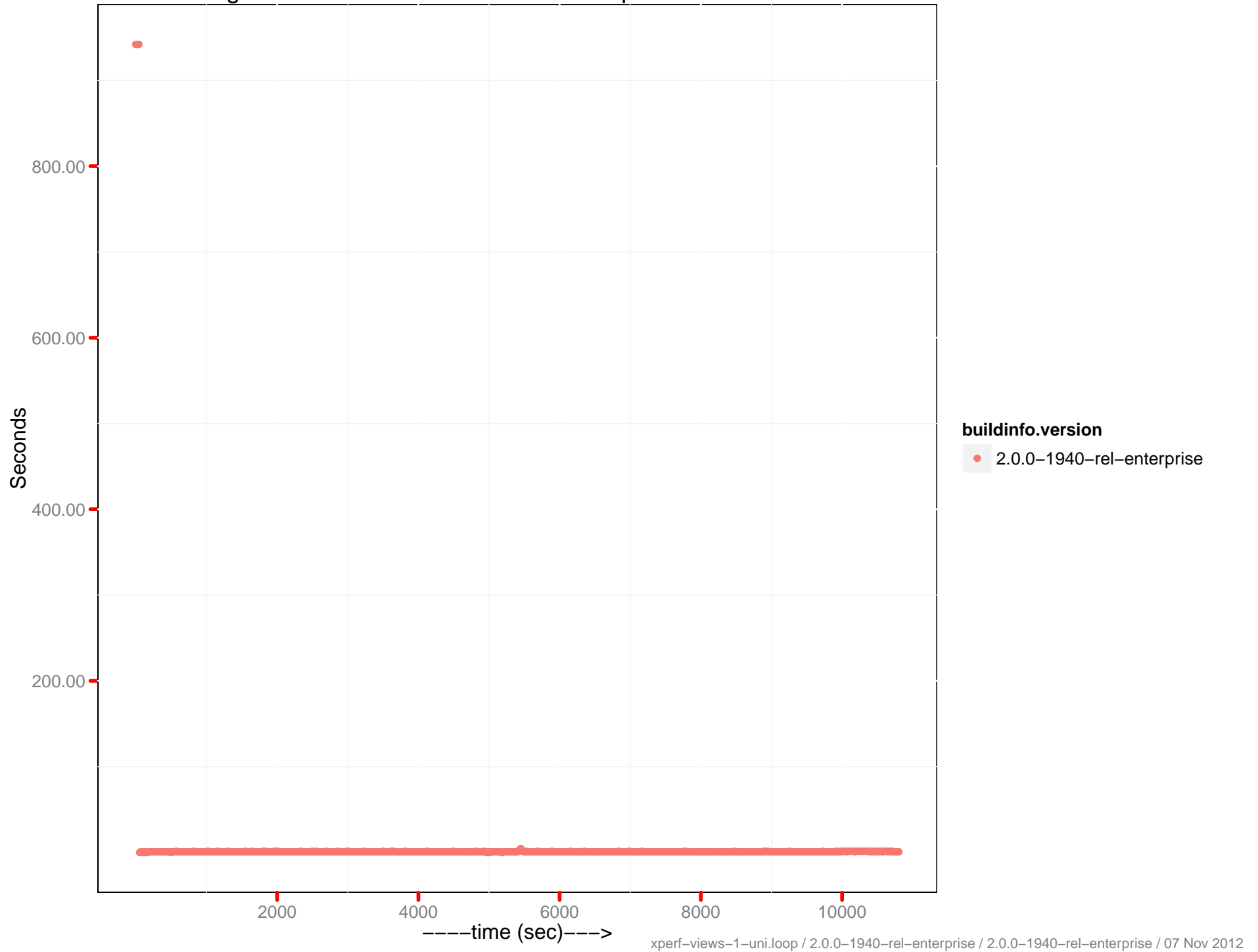
Indexing time – ec2-23-20-61-95.compute-1.amazonaws.com



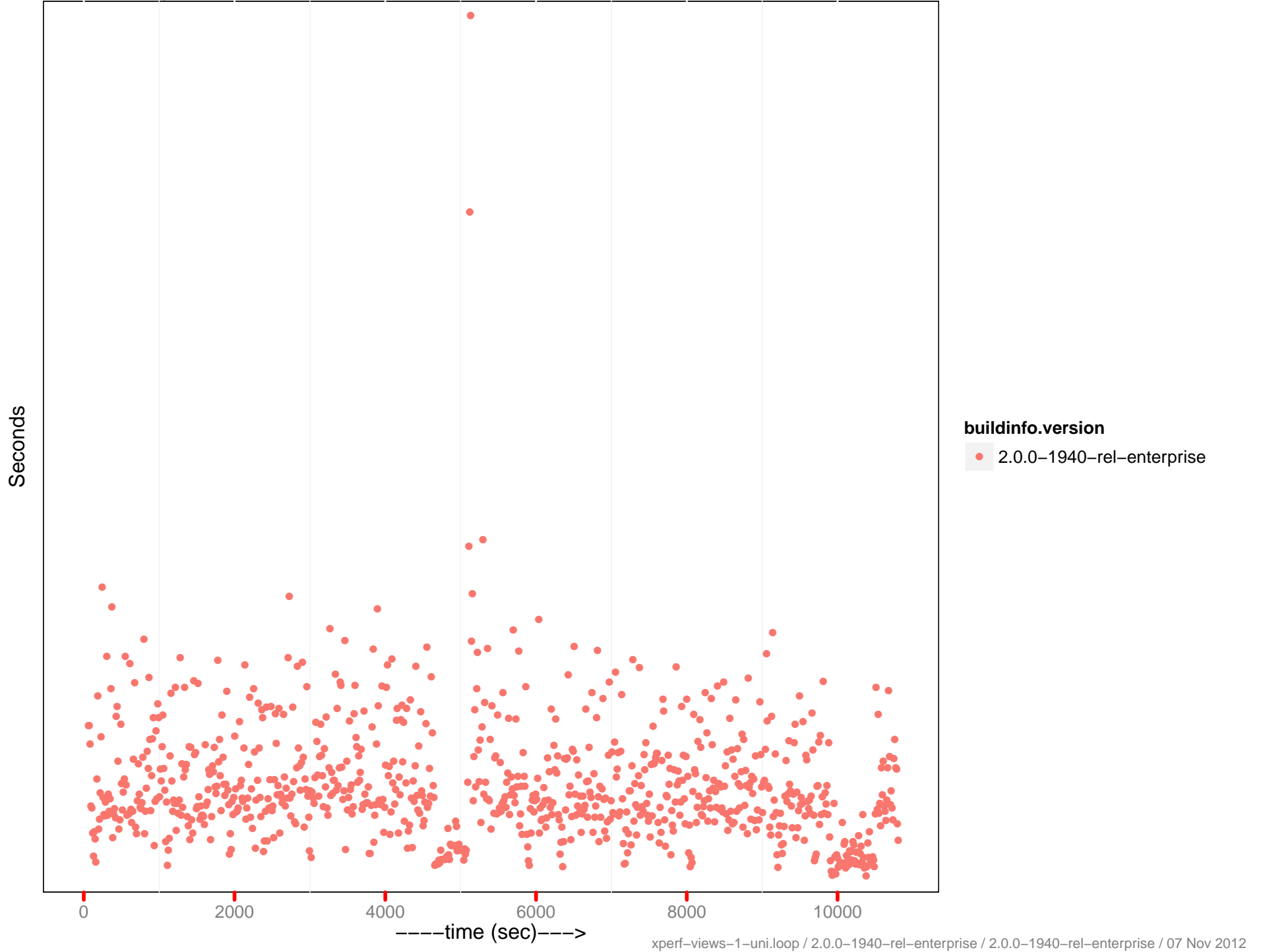
Indexing time – ec2-23-22-246-165.compute-1.amazonaws.com



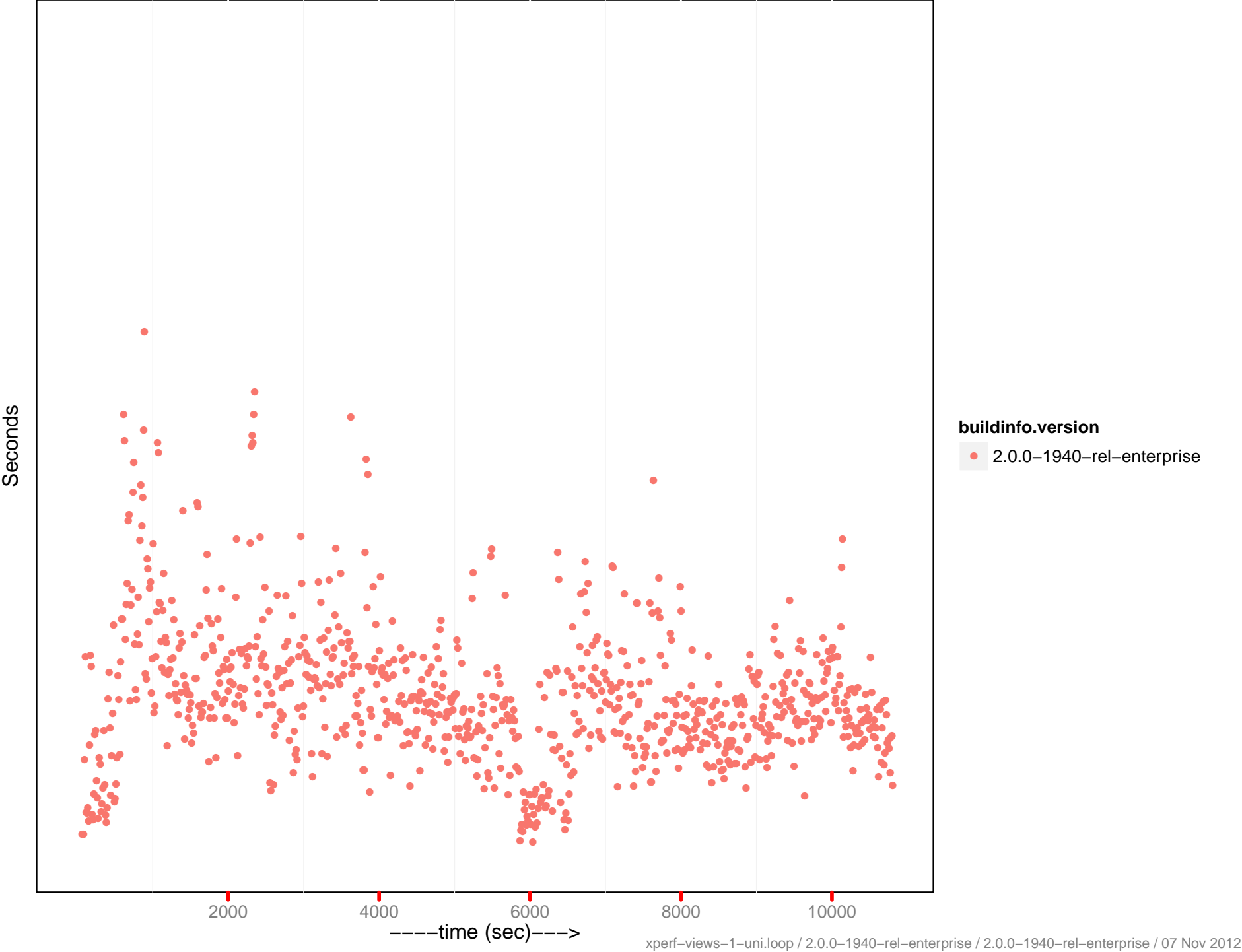
Indexing time – ec2-54-242-190-218.compute-1.amazonaws.com



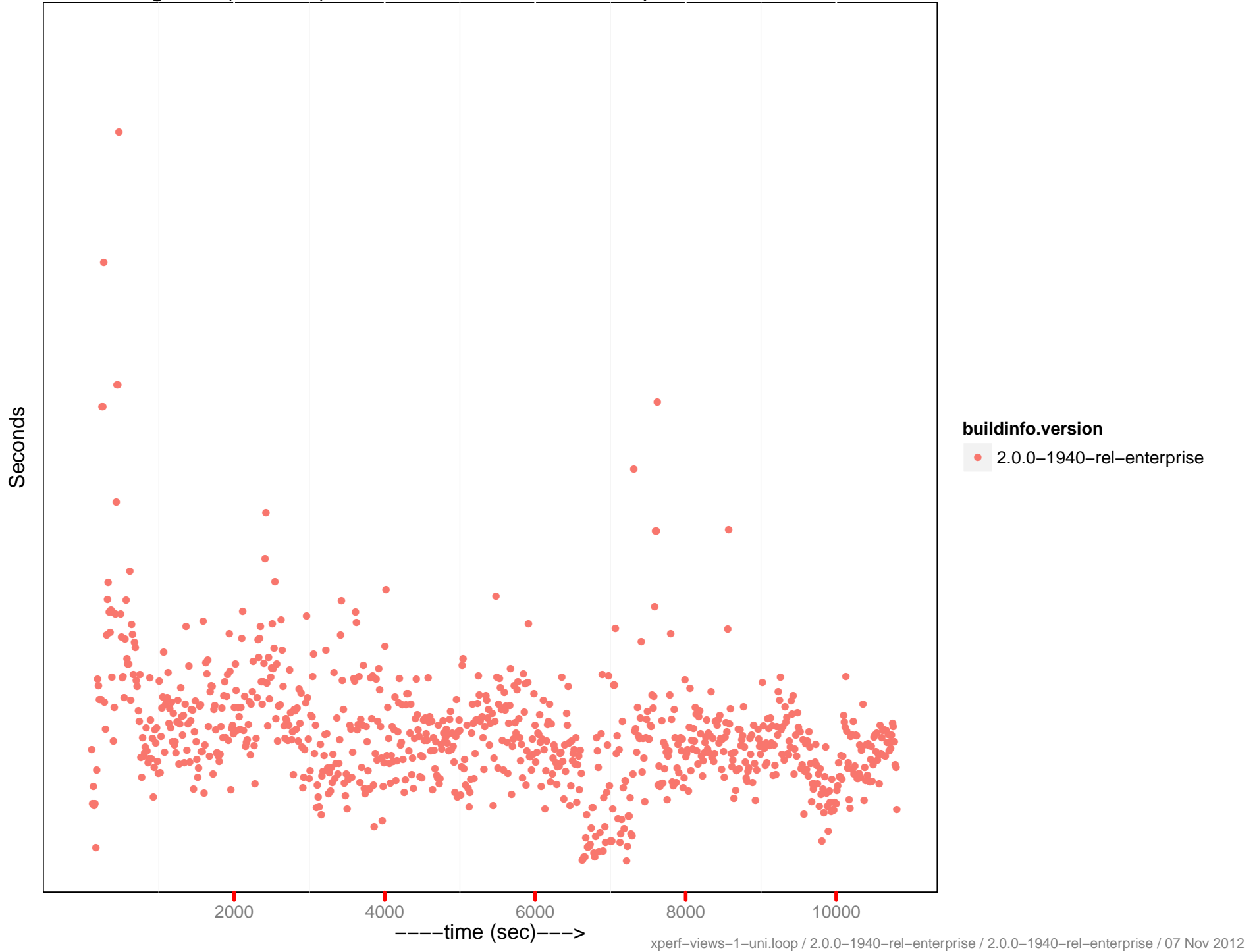
Indexing time (0-5 sec) – ec2-107-20-6-150.compute-1.amazonaws.com



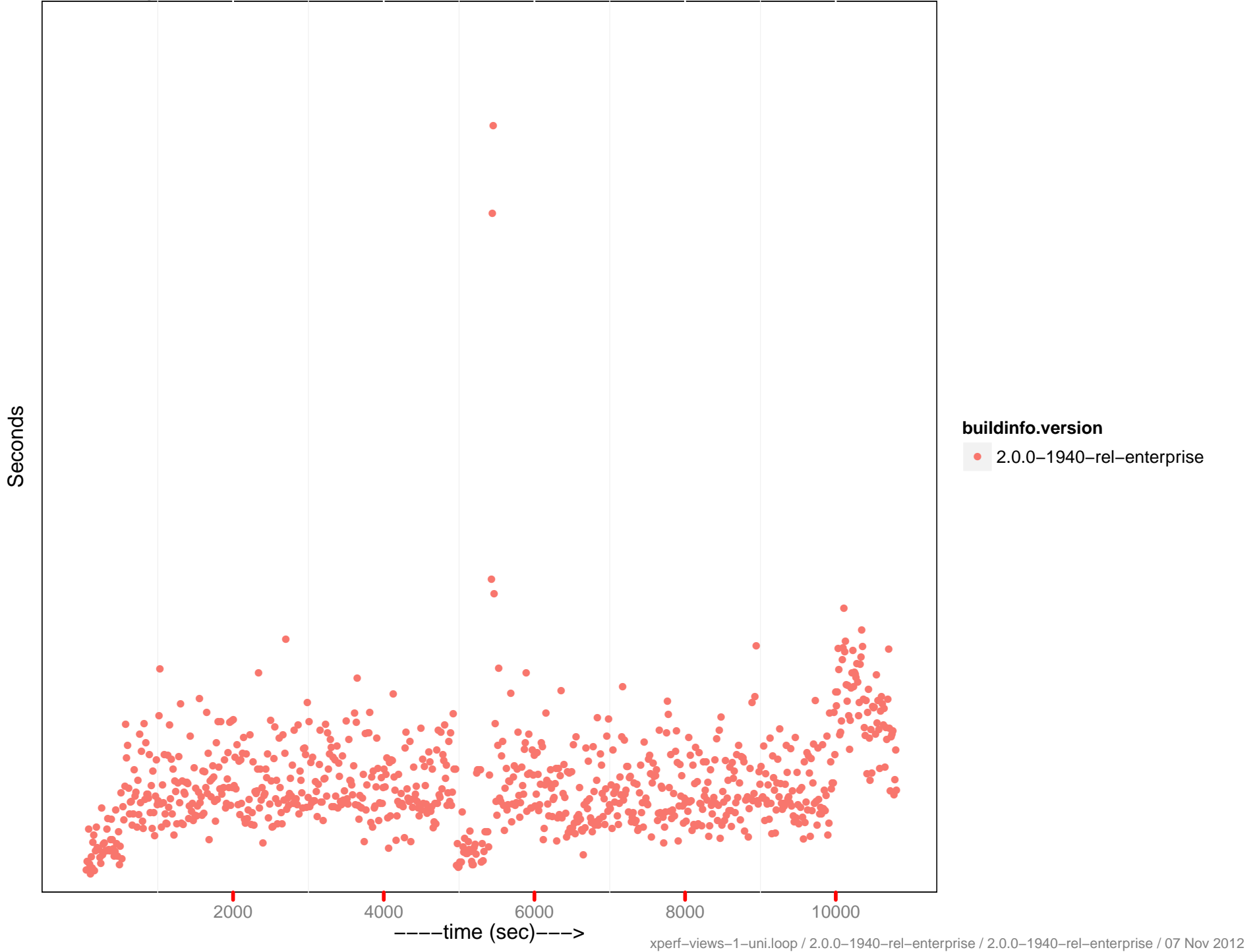
Indexing time (0-5 sec) - ec2-23-20-61-95.compute-1.amazonaws.com



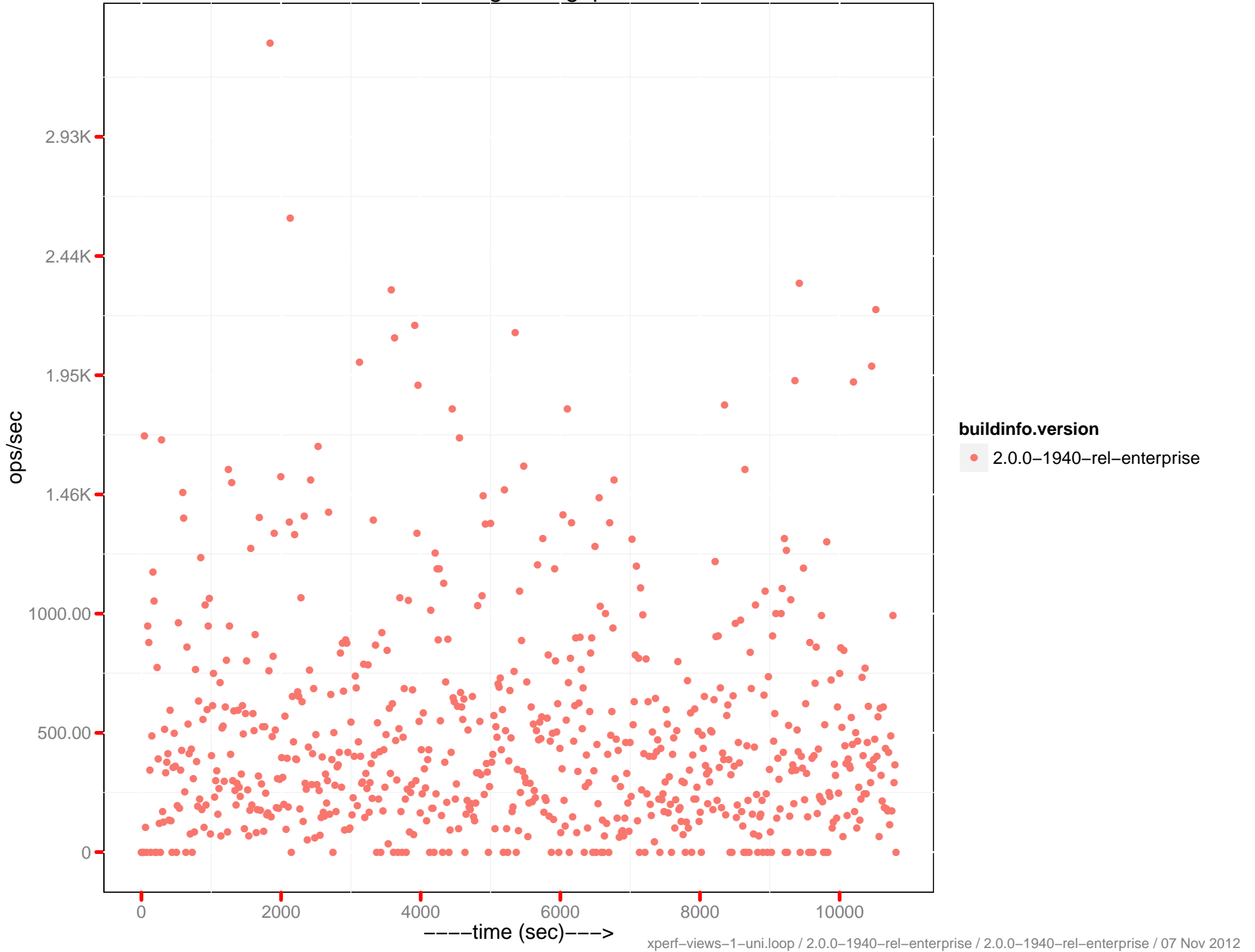
Indexing time (0-5 sec) - ec2-23-22-246-165.compute-1.amazonaws.com



Indexing time (0-5 sec) - ec2-54-242-190-218.compute-1.amazonaws.com



Indexing throughput




```
xperf-views-1-uni.conf
# XPERF test with views:
# 8K ops/sec (background, cluster-wide)
# 50% reads, 50% write (30% updates, 10% deletes, 10% inserts)
# 16 clients per cluster
# 10M dataset
# 3 ddocs with [2-2-4] views
# unidirectional
# 1 bucket
# stop after 3 hours

performance.ipperf.XVPerfTests.test_vperf_3d_unidir

params:

# general
batch=50
kind=json
mem_quota=16000

# xdcr
xdcr_num_buckets=1

# load phase
items=1000000
hot_init_items=2000000
wait_for_xdc_replication=1

# access phase
ratio_sets=0.5
ratio_misses=0.025
ratio_creates=0.20
ratio_deletes=0.25
ratio_hot=0.2
ratio_hot_gets=0.95
ratio_hot_sets=0.95
ratio_expirations=0.0
bg_max_ops_per_sec=500
fg_max_ops=8000000000
total_clients=16
time=10800

# control (defaults: pytest/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
```

