P99 CONF

How to Measure Latency?

Heinrich Hartmann

Principal Engineer @ zalando

Brought to you by SCYLLA

Motivation and Background

- I have been talking about Statistics and Latency for the last years
  State of the Histogram (SLOConf 2021) / Circllhist (paper) / Latency SLOs Done Right (FOSDEM 2019) / Statistics for Engineers (2014..2019)

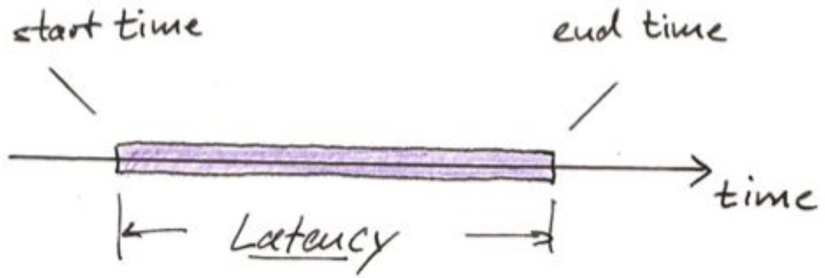
- Inspiration comes from series of talks from ~2013-15
  Gil Tene - How (not) to measure Latency
  Slides (London 2013) / Video (StrangeLoop 2015) / Blog - HighScalability 2015

- On Coordinated Omission - Ivan Prisyazhynyy
  Published two days ago on P99CONF.io

"It's slow" is the hardest problem you will ever debug.

Theo Schlossnagle @postwait

What is Latency?



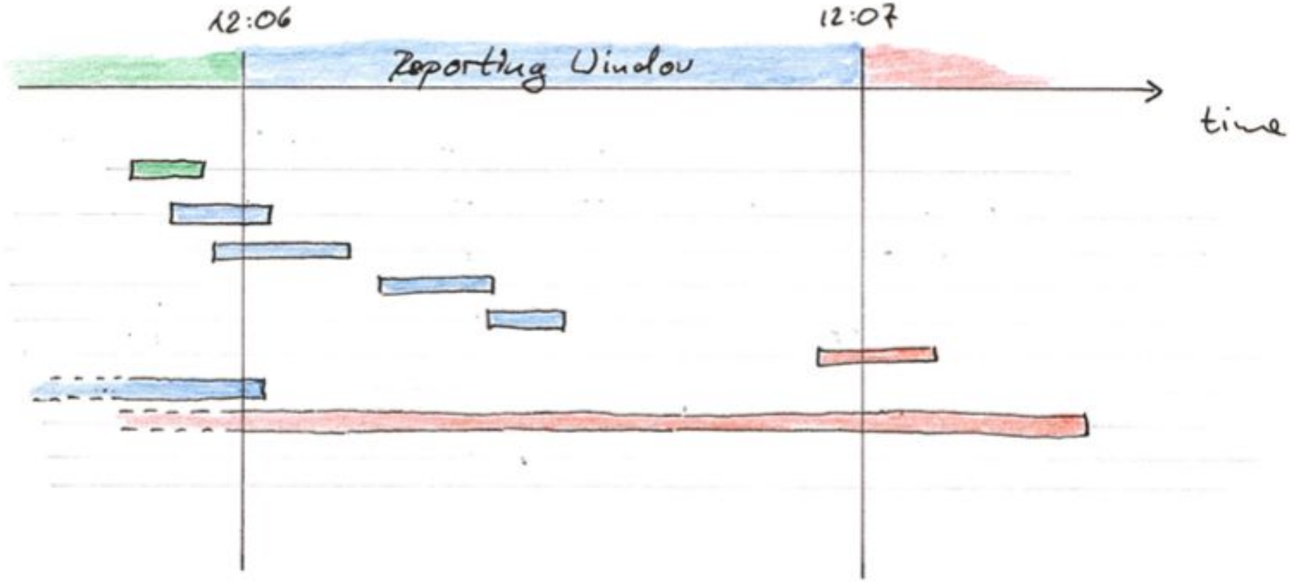
How to Measure Latency?

```
t_start = time.now()

#
# operation you want to measure
#

latency = time.now() - t_start
```

Things to watch out for

- Capture early returns / exceptions
  - Use: try/catch/finally or defer
- Which clock is used?
  - Want: high-resolution, monotonic, system time (e.g. time.monotonic() in Python)
- Measurement Overhead
  - Measuring time takes time (at least 30ns, often >300ns)
  - OK for 0.1ms or more (I/O Latency)
  - Careful for 10us or lower (micro benchmarking)
- Abstracting time measurements in code
  - Write a @timed decorator. Use tracing libraries (@trace)

Measuring Latency over Time



Measuring Latency over Time - Hints

- Common / Pragmatic Approach:
  a. Associated spans to the reporting window where they ended
  b. Don't record latency of ongoing spans

- Caution: No latency measurements =/=> No active requests.

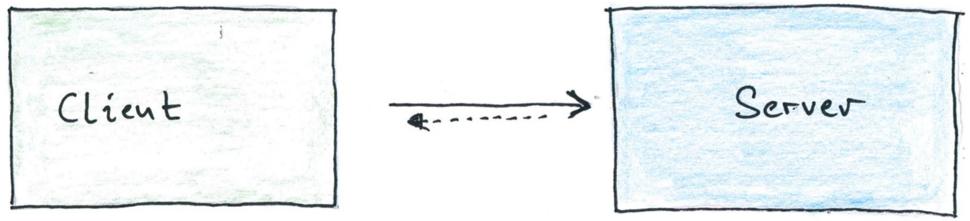
- Keep an eye on concurrency and max latency to catch this

- When benchmarking: Wait until all requests completed

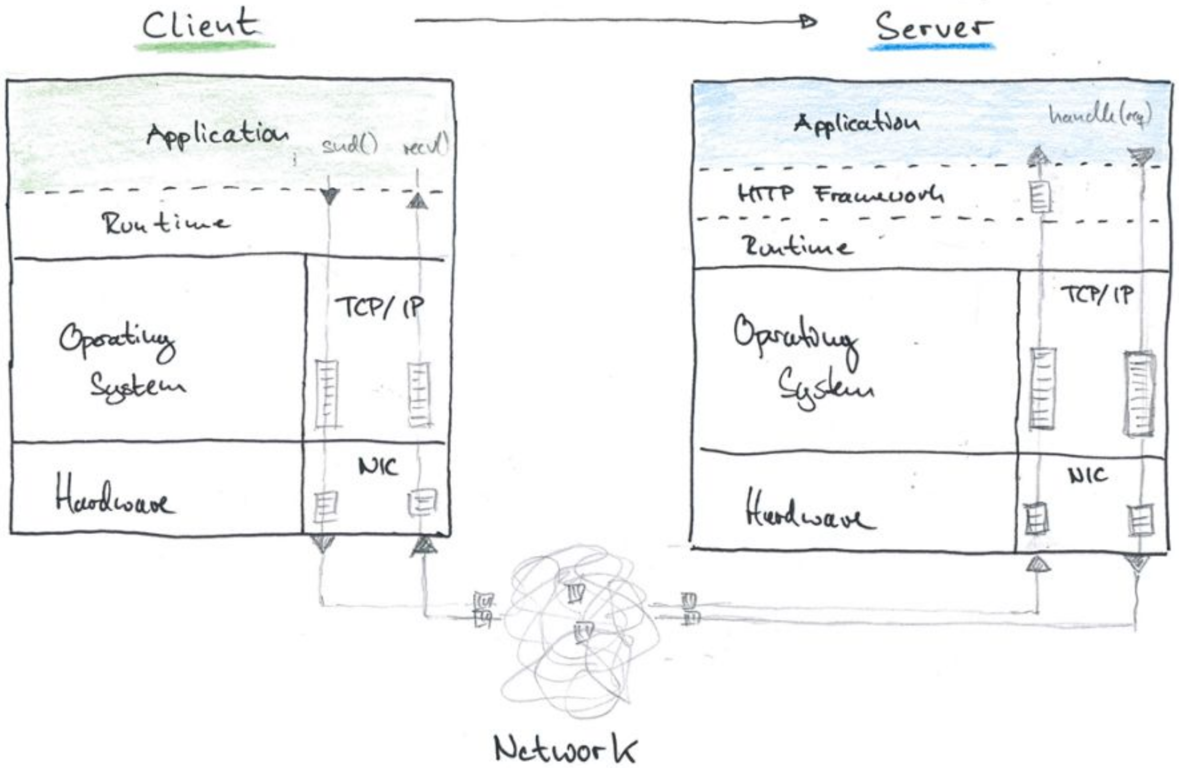
The End

Where to Measure Latency?

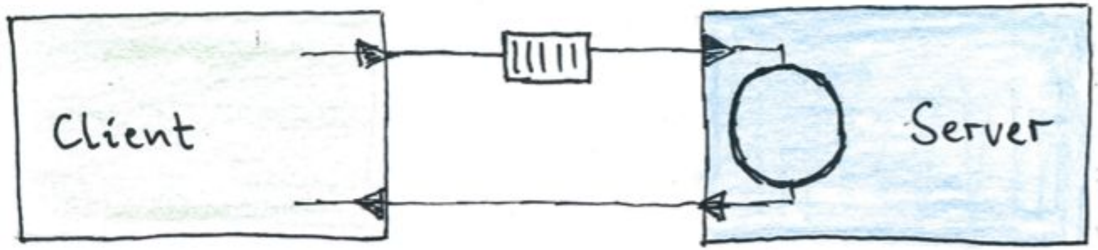
Hidden Queues



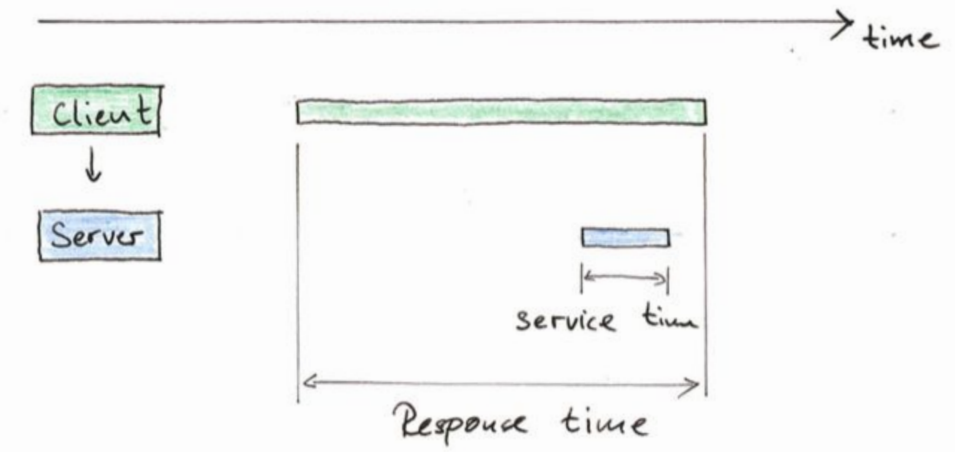
Hidden Queues



A practical Queuing Model



Response Time vs. Service Time



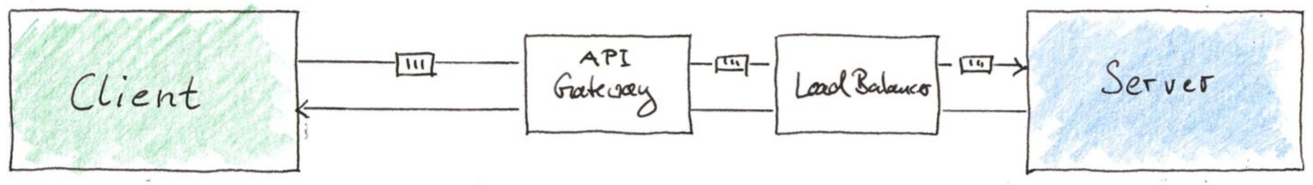
Response Time vs. Service Time

Response Time

Service Time

Image Credits: ŠJů, Wikimedia Commons, CC BY-SA 3.0 , via Wikimedia Commons

Where to Measure Latency?



+ Most meaningful

- Hard to implement

- Only get Service Time

+ Easy to implement

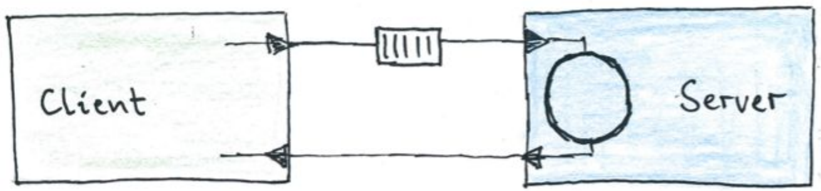
You can't measure Response Time on the Server.

SAD BUT TRUE

Request Time vs. Service Time

An Experiment

Simulation Setup



- 10 workers
- 10ms service time
- 1K rps capacity

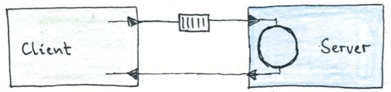
Metrics

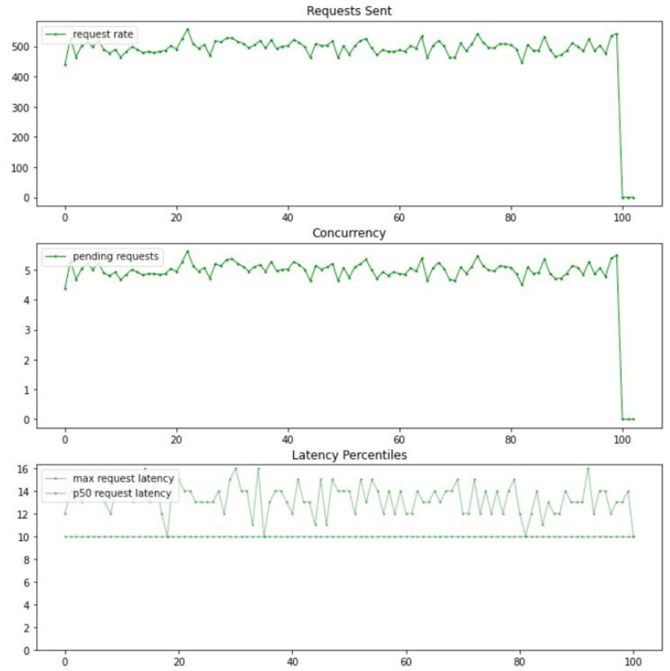
- Request Rate (~ constant)
- Concurrency (Active Requests)
- Response Time

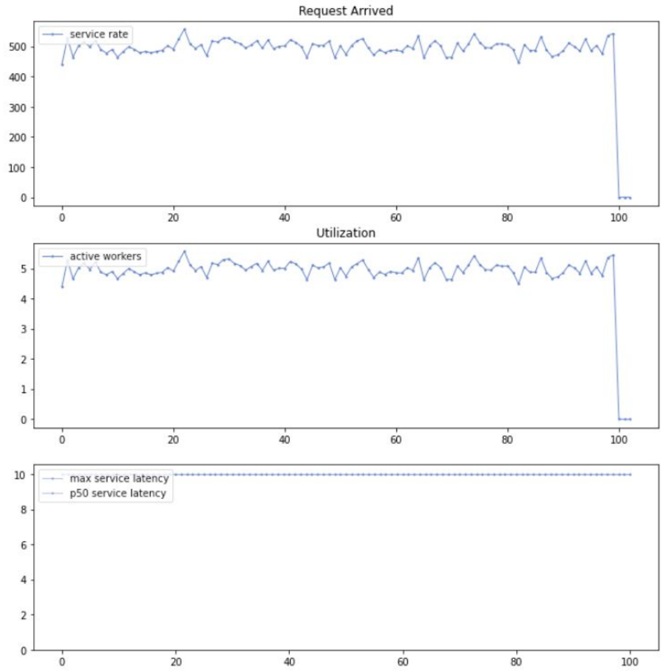
- Arrival Rate
- Concurrency (Active Workers)
- Service Time (constant)

code @ https://github.com/HeinrichHartmann/libq

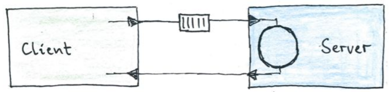
Queuing System at 50% Capacity

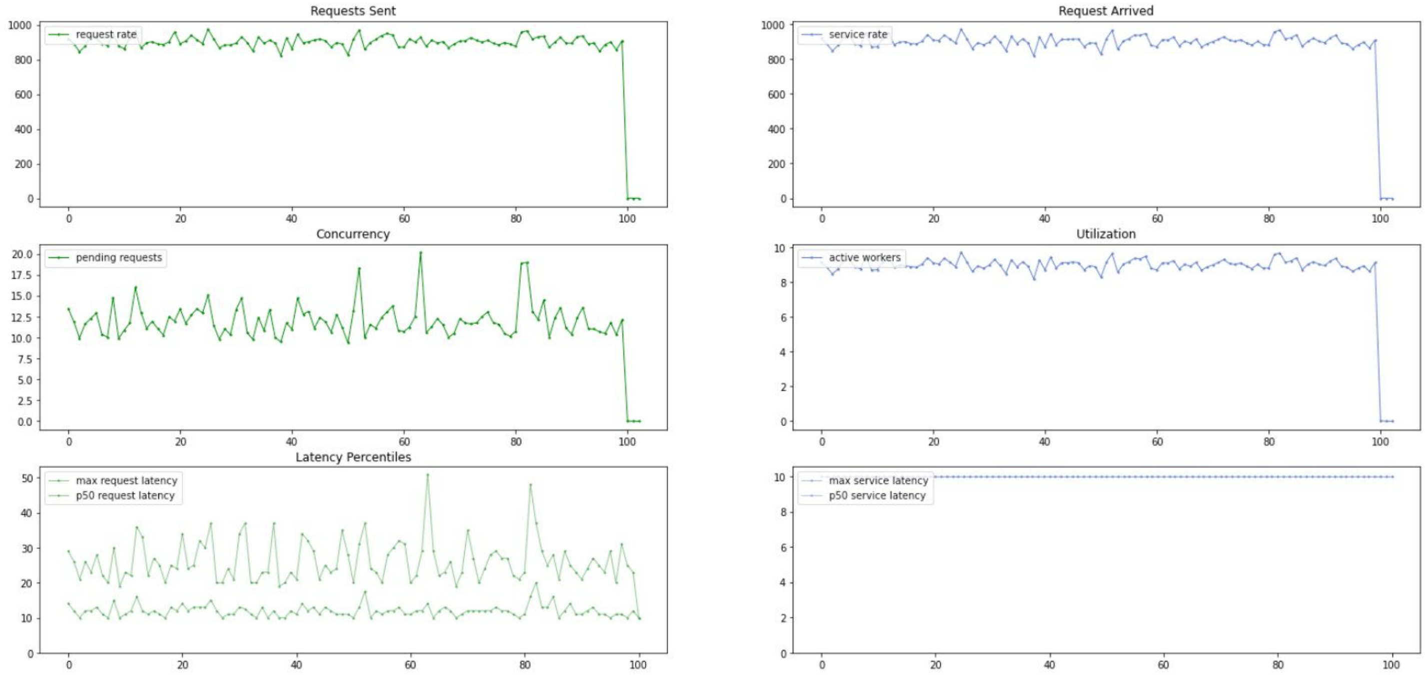




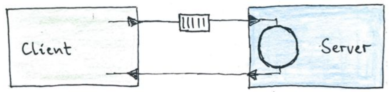


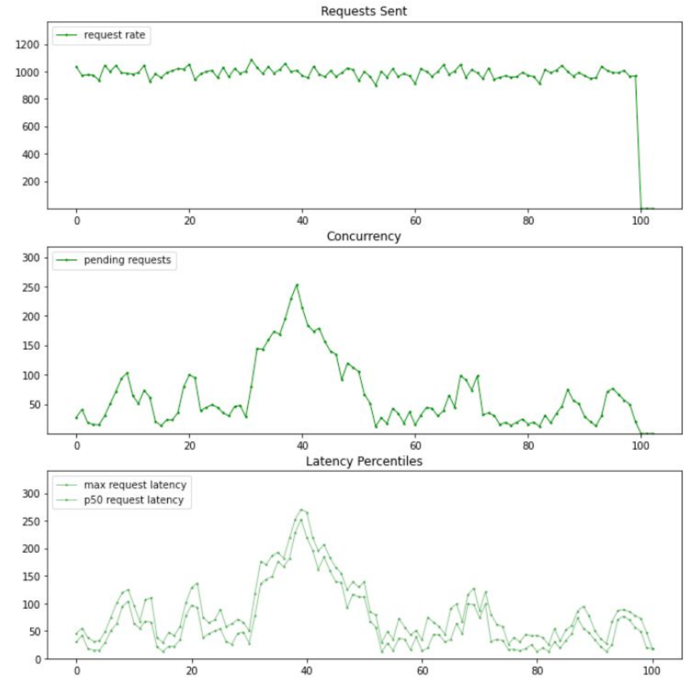
Queuing System at 90% Capacity

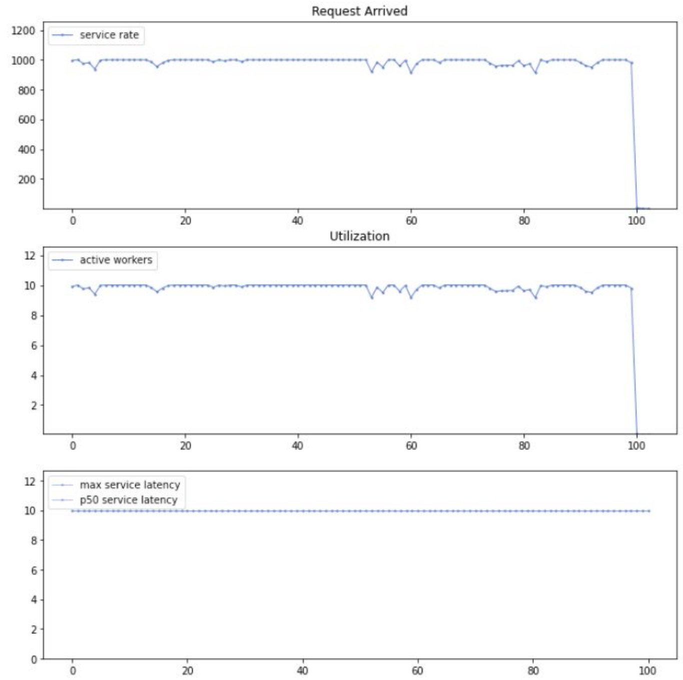




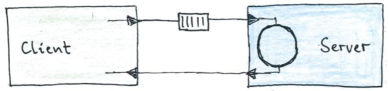
Queuing System at 99% Capacity

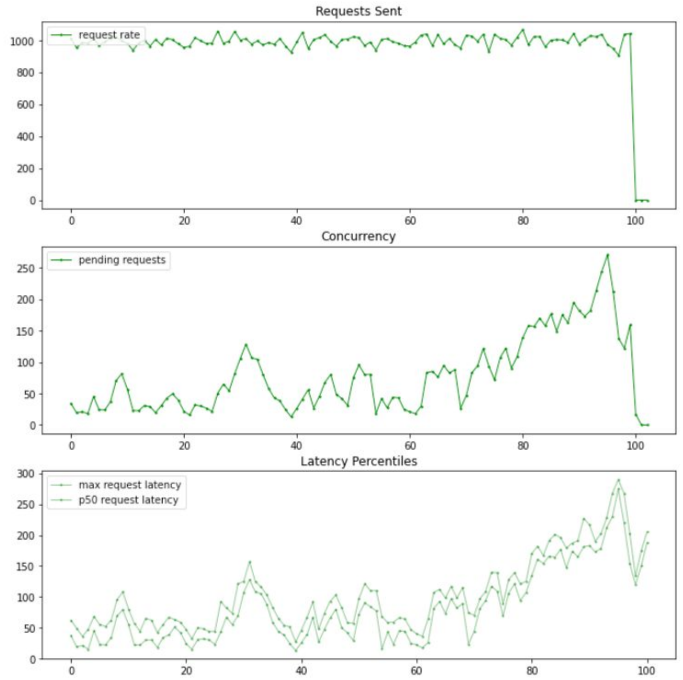


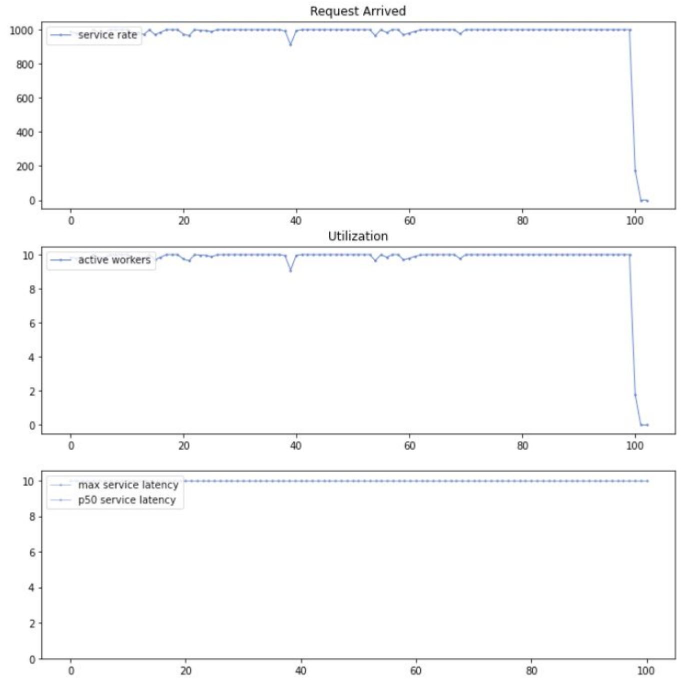




Queuing System at 100% Capacity

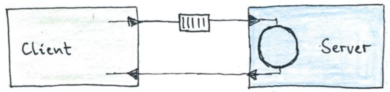


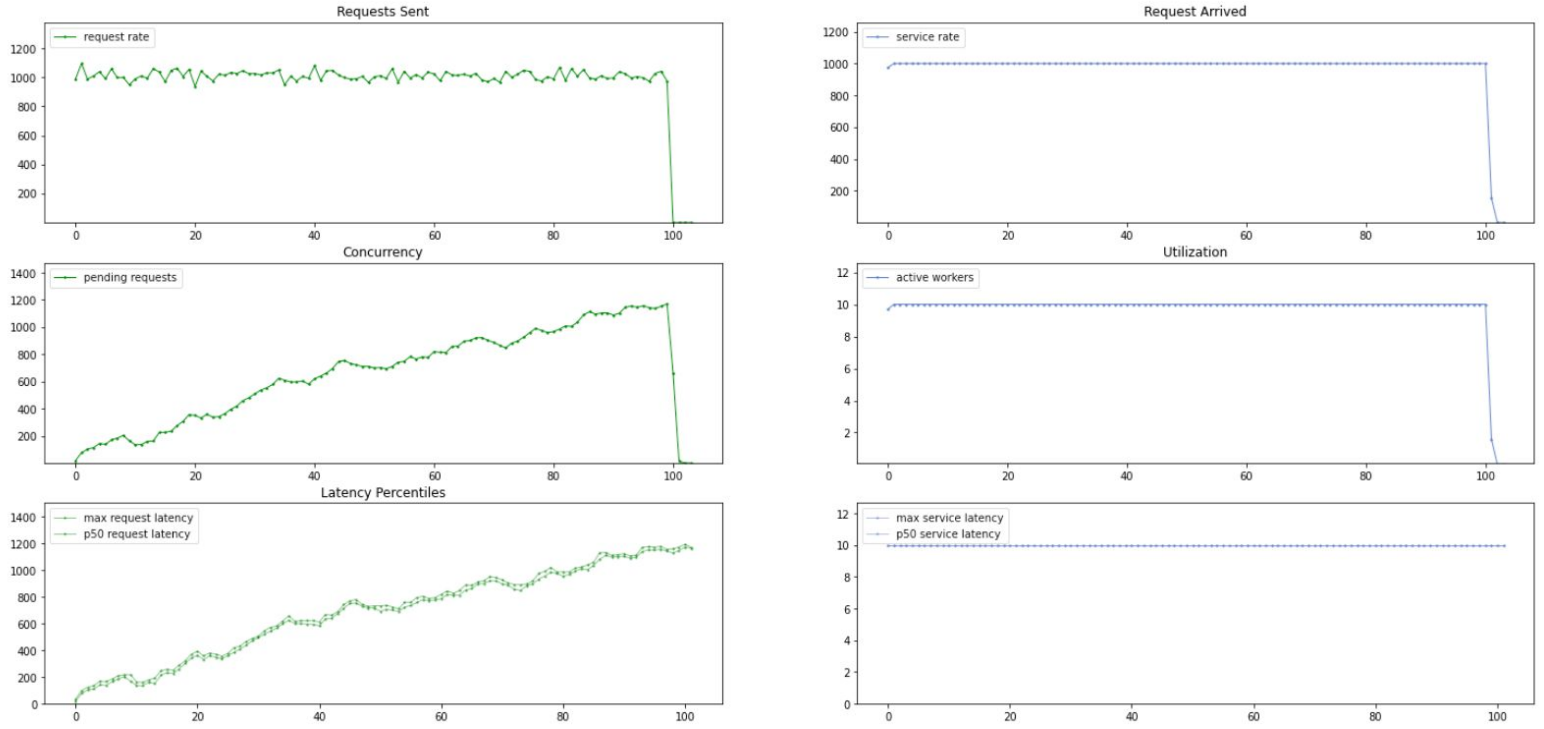




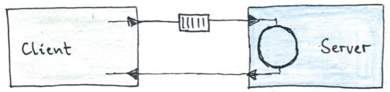
@HeinrichHartmann

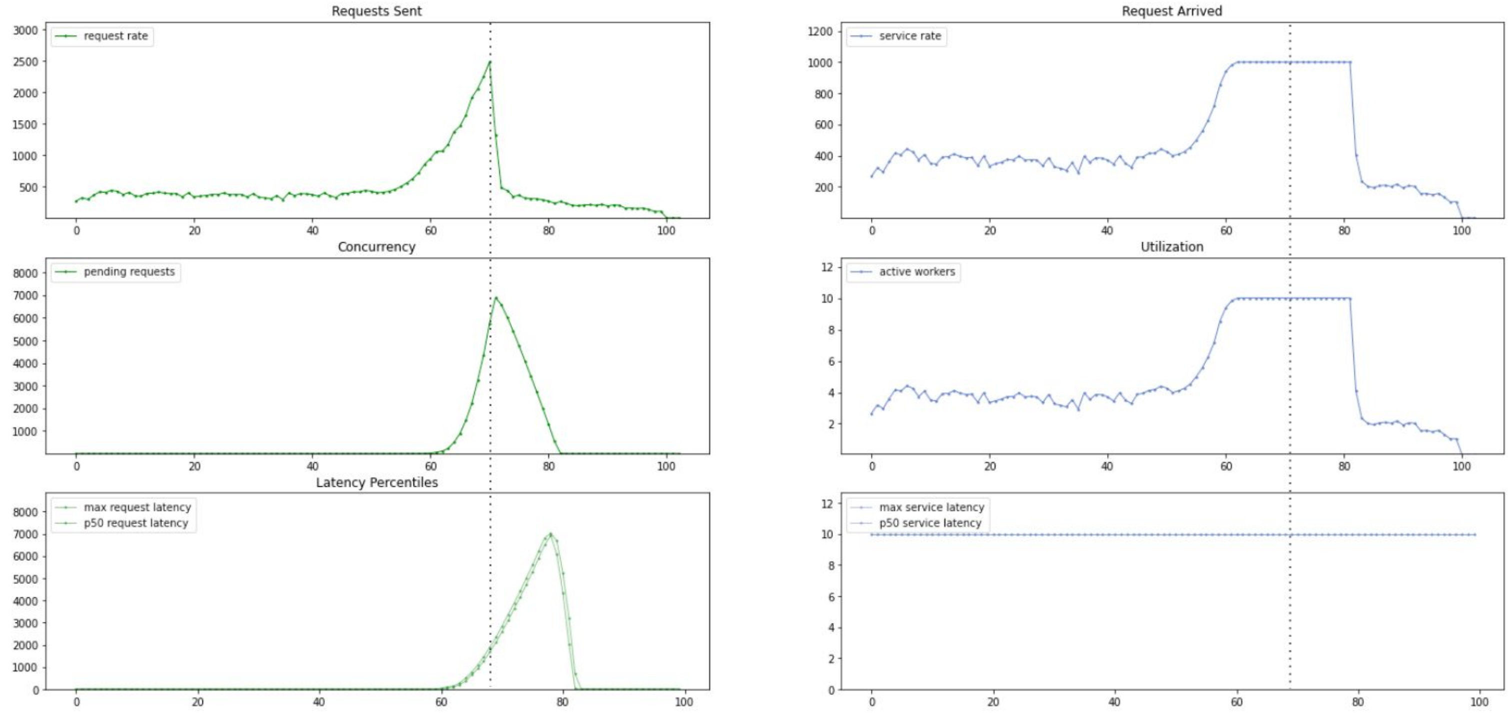
Queuing System at 101% Capacity



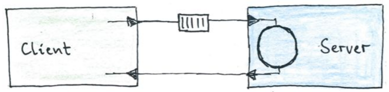


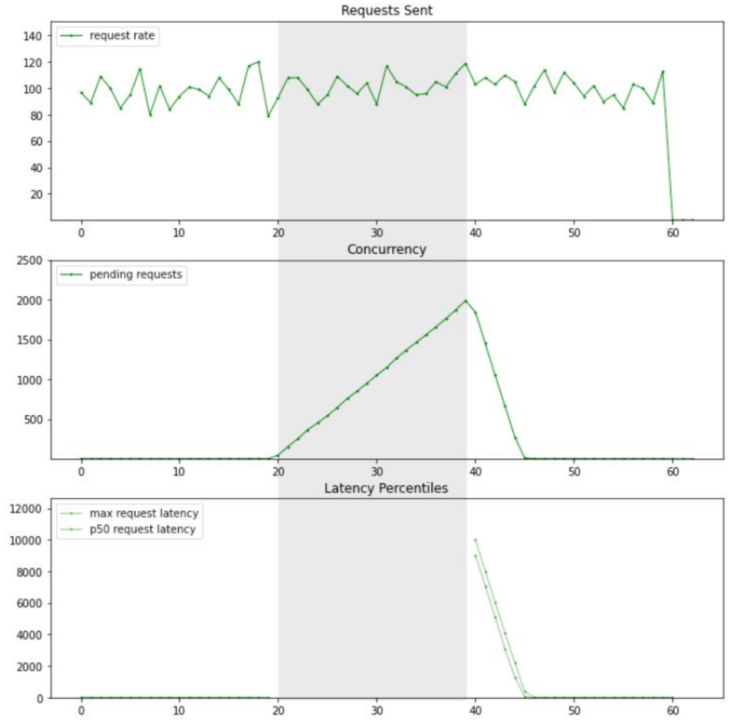
A Hockey Stick

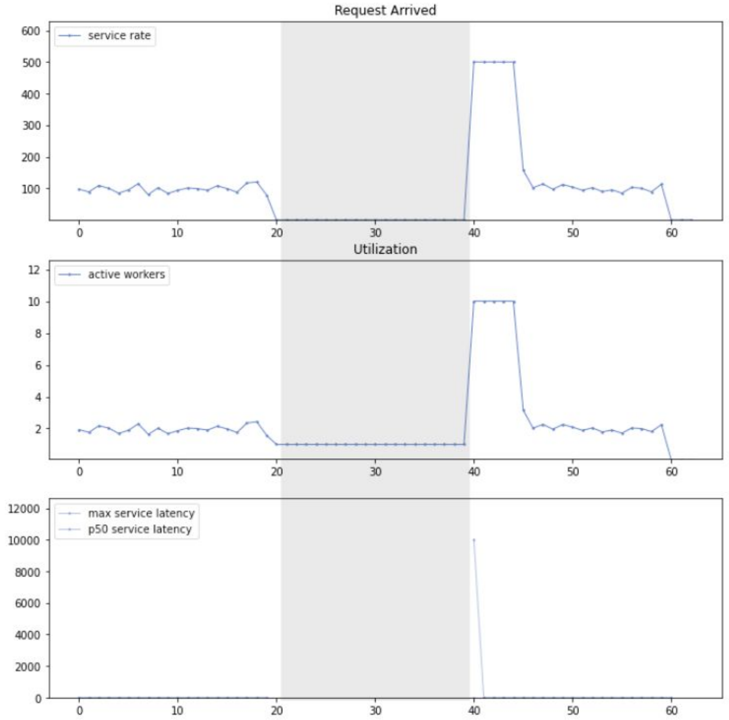




A Stalled System







Coordinated Omission in Load Testing

Def. Coordination between Load Generator (Client) and Server that leads to confusing Service Time with Response times.

Examples

- Client backs off when server is falling behind
- Client is stalled when Server is stalled

This is surprisingly common (cf. *Gil's talk*, Ivan's blog)

How to Avoid Coordinated Omission

- Push your systems to/over the breaking point
  - Do you see unbounded latency increases? (good)
  - Do you see a smooth increase in latency as you increase load? (good)

- Halt (SIGSTOP, CTRL-Z) your server during load test:
  - How do your latency metrics react?
  - Do you see lot's of slow requests? (good)
  - Do you see only a few slow requests? (bad)

How to Analyze Latency Data?

Requirements for Latency Analysis

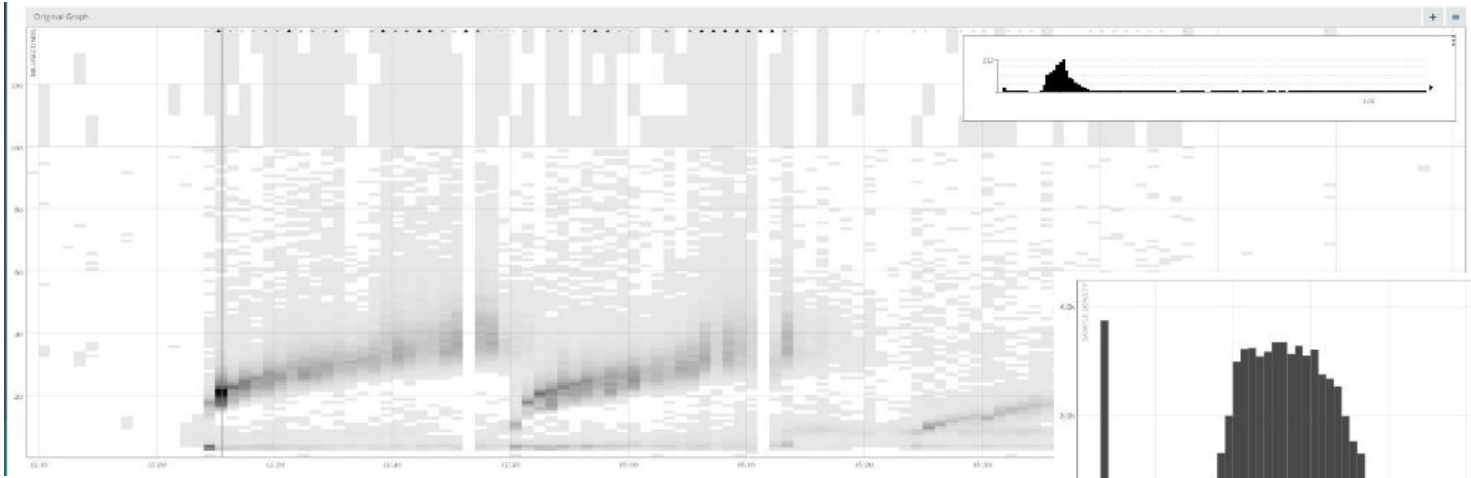
- Calculate Accurate Percentiles (arbitrary / fixed)
- Calculate "less-than" ratios (arbitrary / fixed)
- Mergability of Datasets (across time and levels)

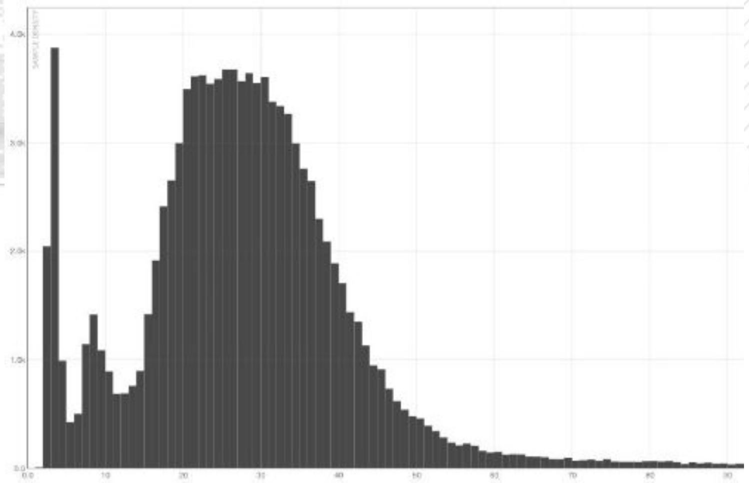
Available Options

Technology	Percentiles	LT-Ratios	Mergability	Comment
Raw Data (traces, logs)	Arbitrary	Arbitrary	Yes	Expensive to store.
Sampled Raw Data	Arbitrary	Arbitrary?	Yes	Accuracy highly depends on method and sampling rate
Percentile Metrics	Fixed	No	No	No Aggregation Possible.
"LessThan" - Metrics	No (low accuracy)	Fixed	Yes	Example: Prometheus Histograms. OK if thresholds are known.
Histograms Metrics	Yes	Yes	Yes	Best Practice.

More Details: Latency SLOs done right @ FOSDEM 2019.

Best Practice: Histogram (Metrics)





More Details:
- Latency SLOs done right @ FOSDEM 2019
- State of The Histogram @ SLOConf 2021 [slides] [video]

P99 CONF

Thank you!

Further Reading

- HeinrichHartmann.com/latency
- @HeinrichHartmann