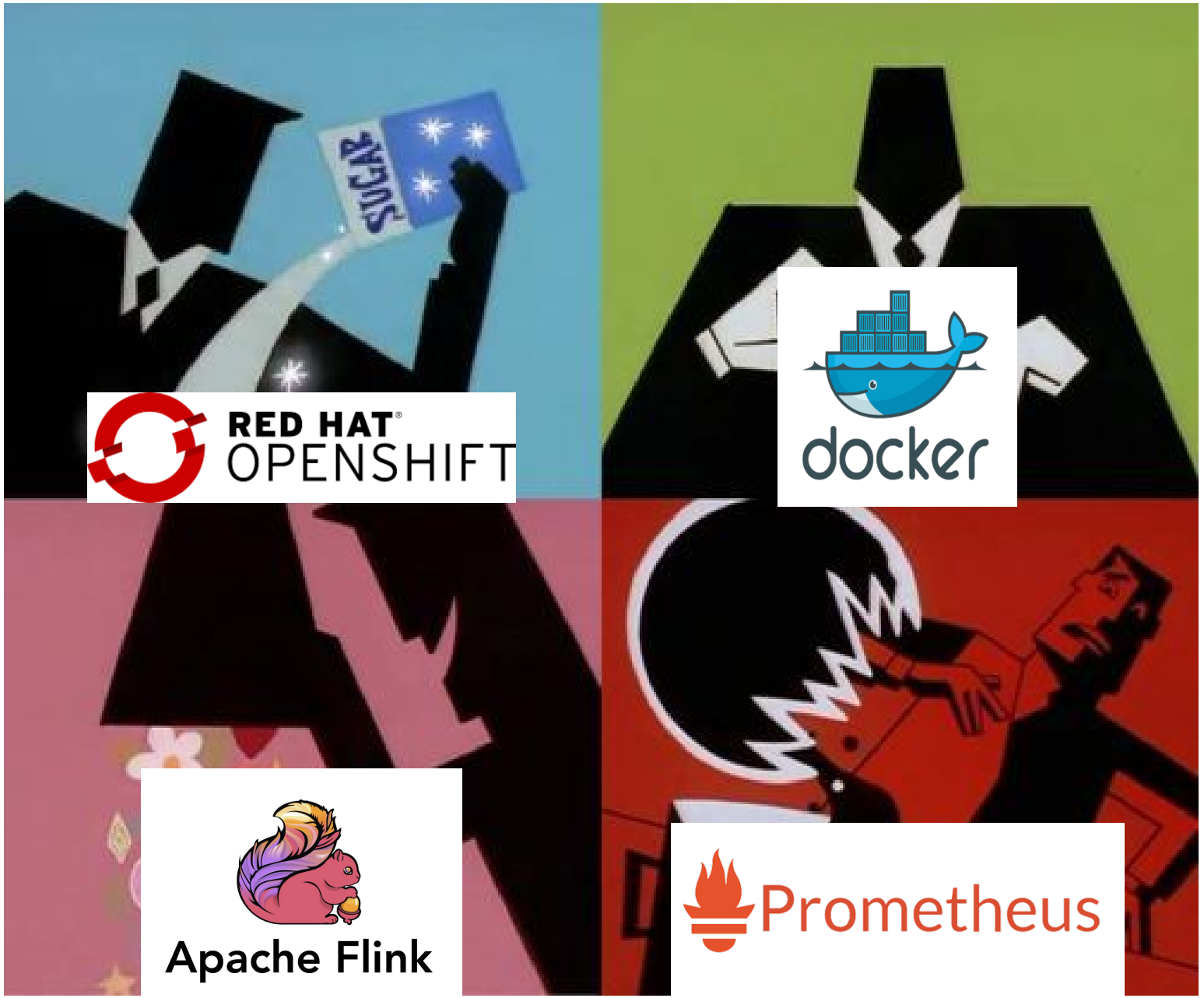
Automated Benchmarking of Container Applications

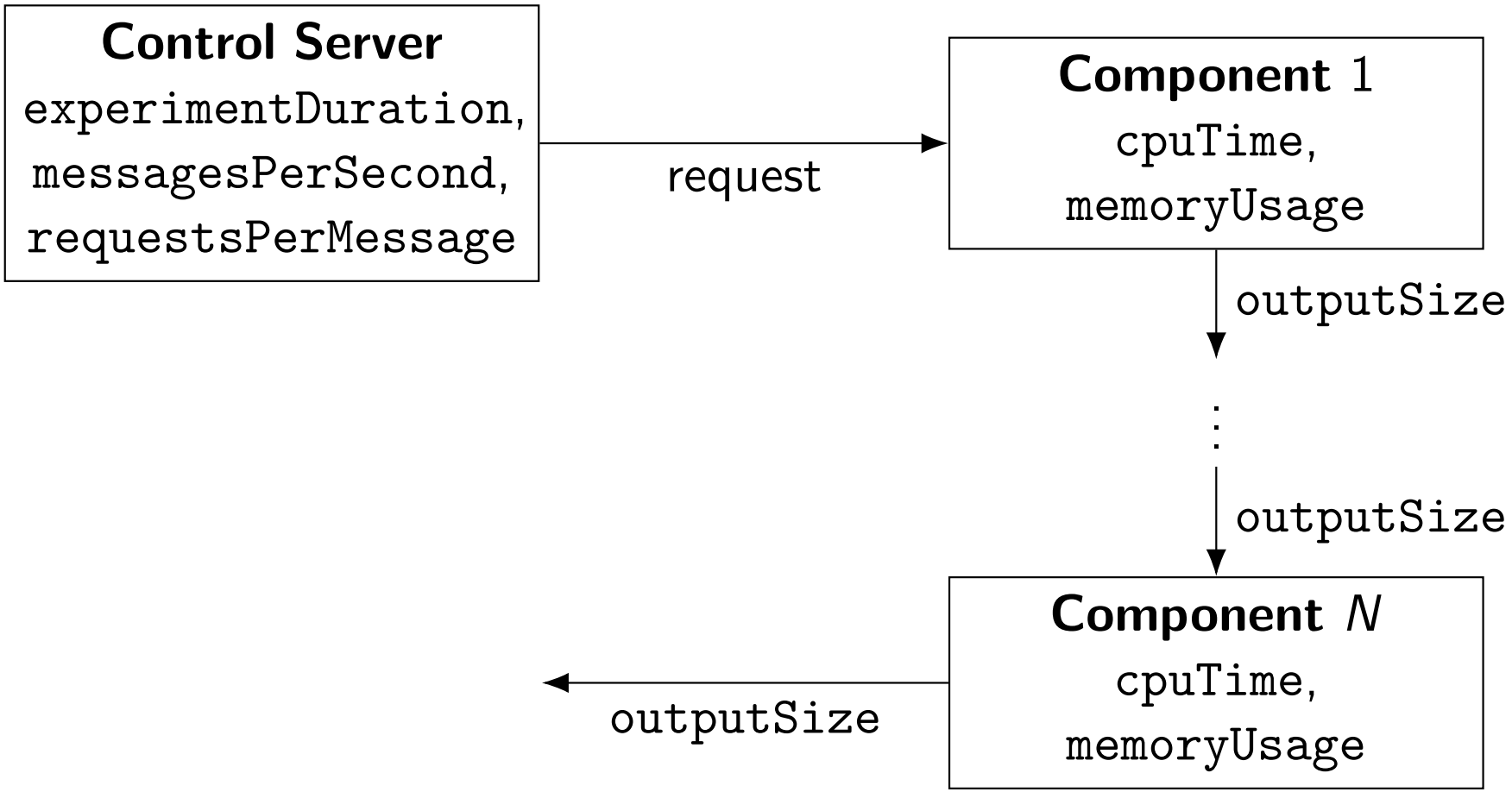
Paulius Dilkas

1st August 2019

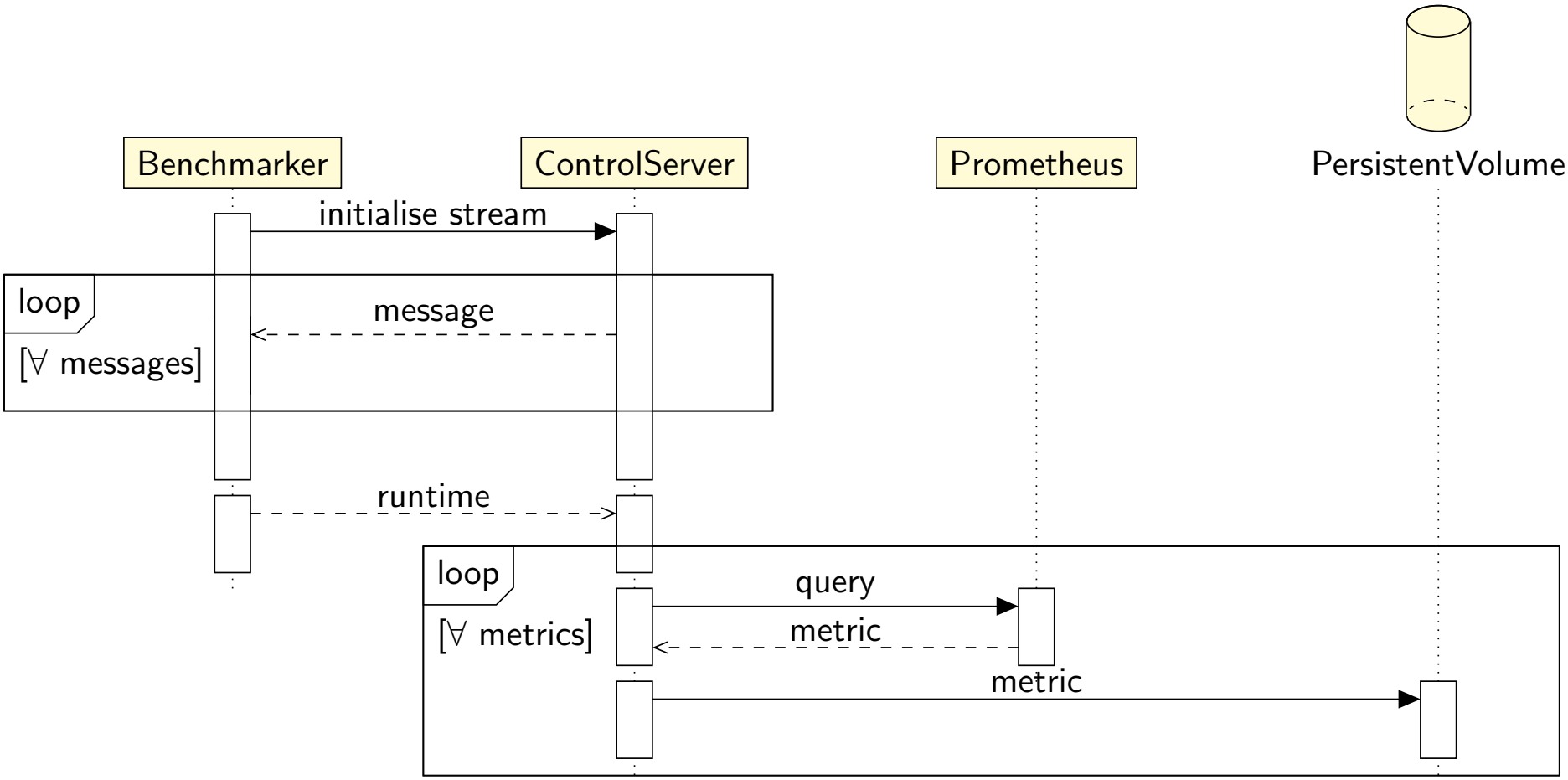
Main Ingredients



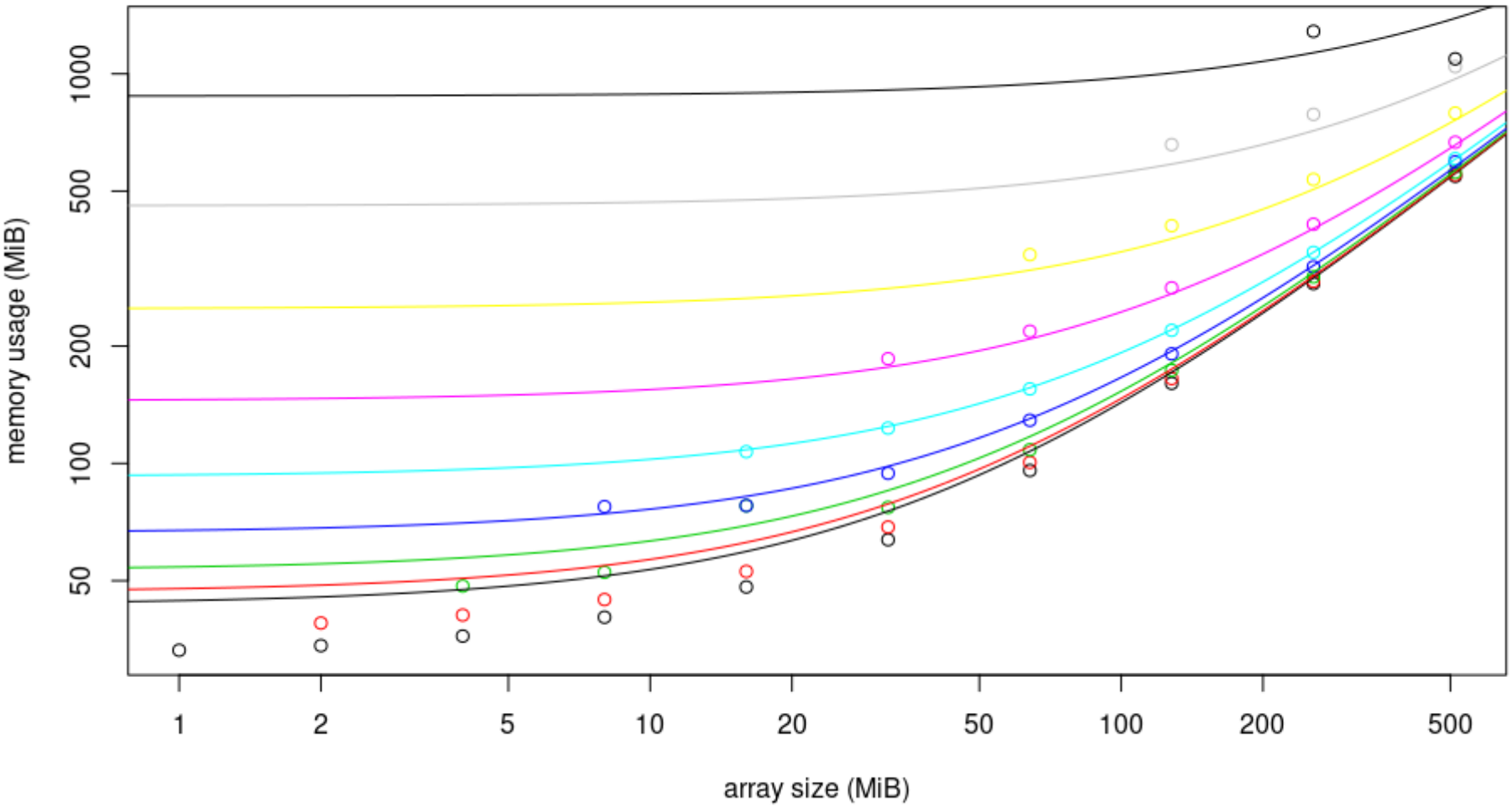
The General Idea



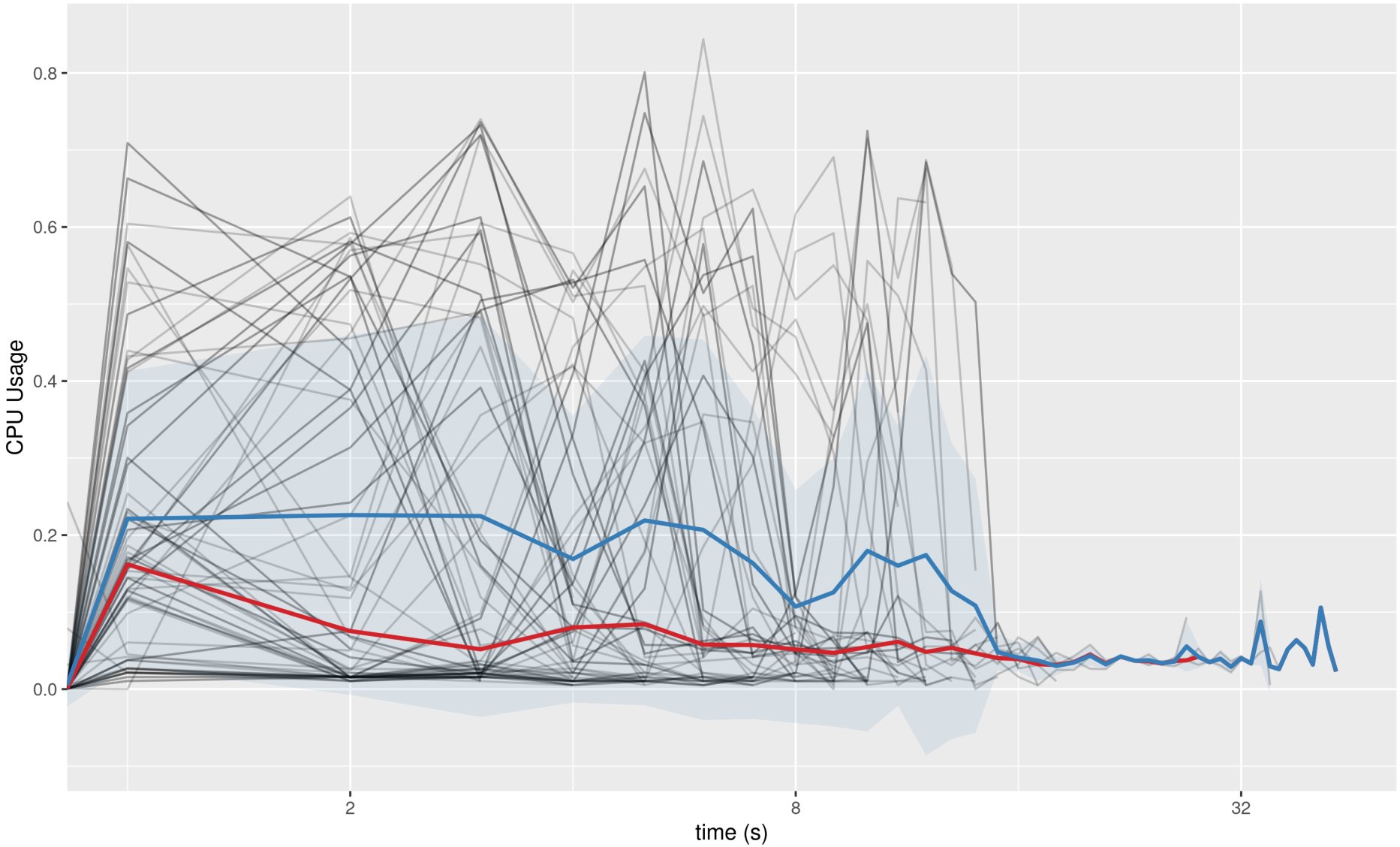
Execution


Benchmarker
ControlServer
Prometheus
PersistentVolume
initialise stream
loop
[∀ messages]
message
runtime
loop
[∀ metrics]
query
metric
metric

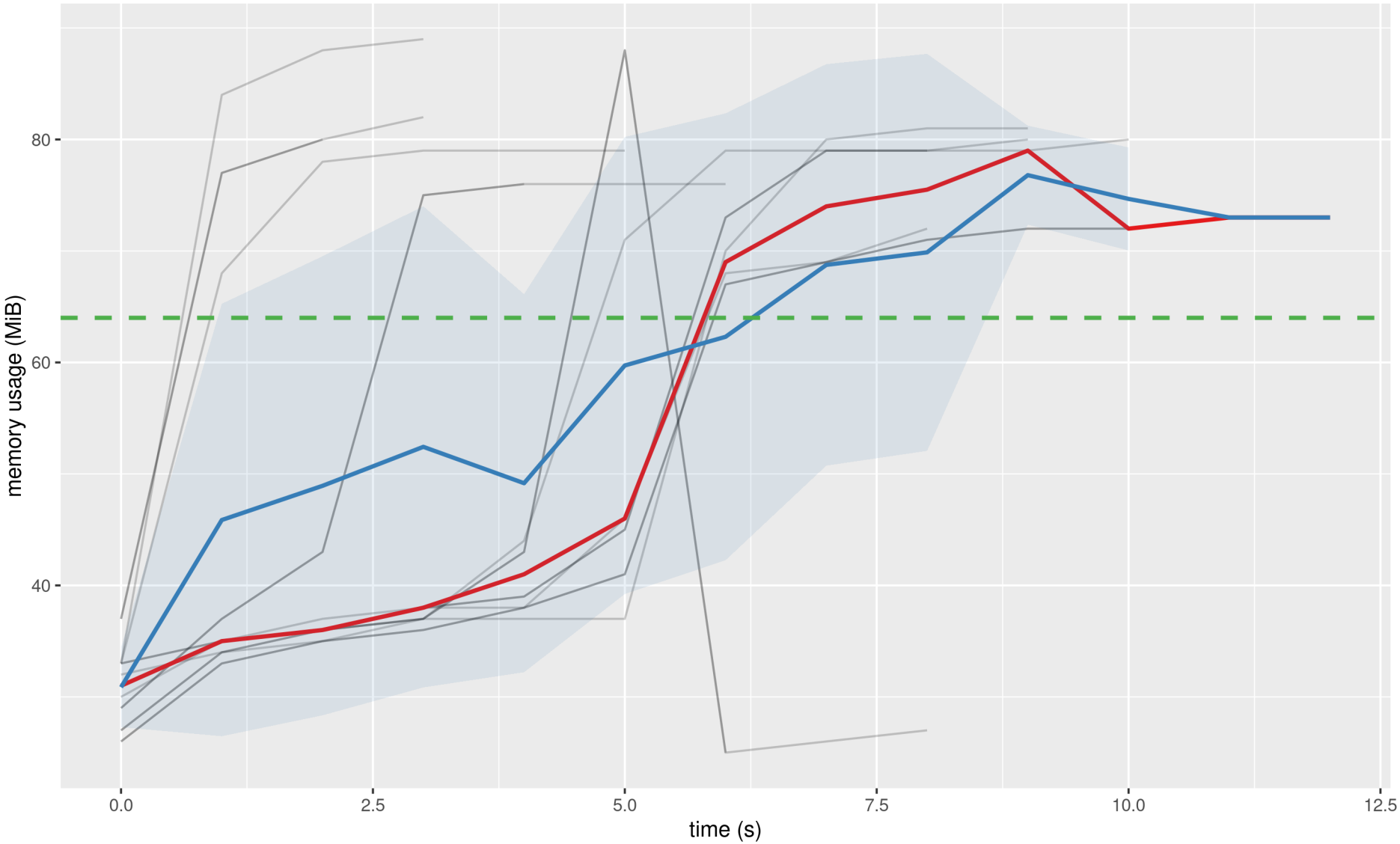
Calibration



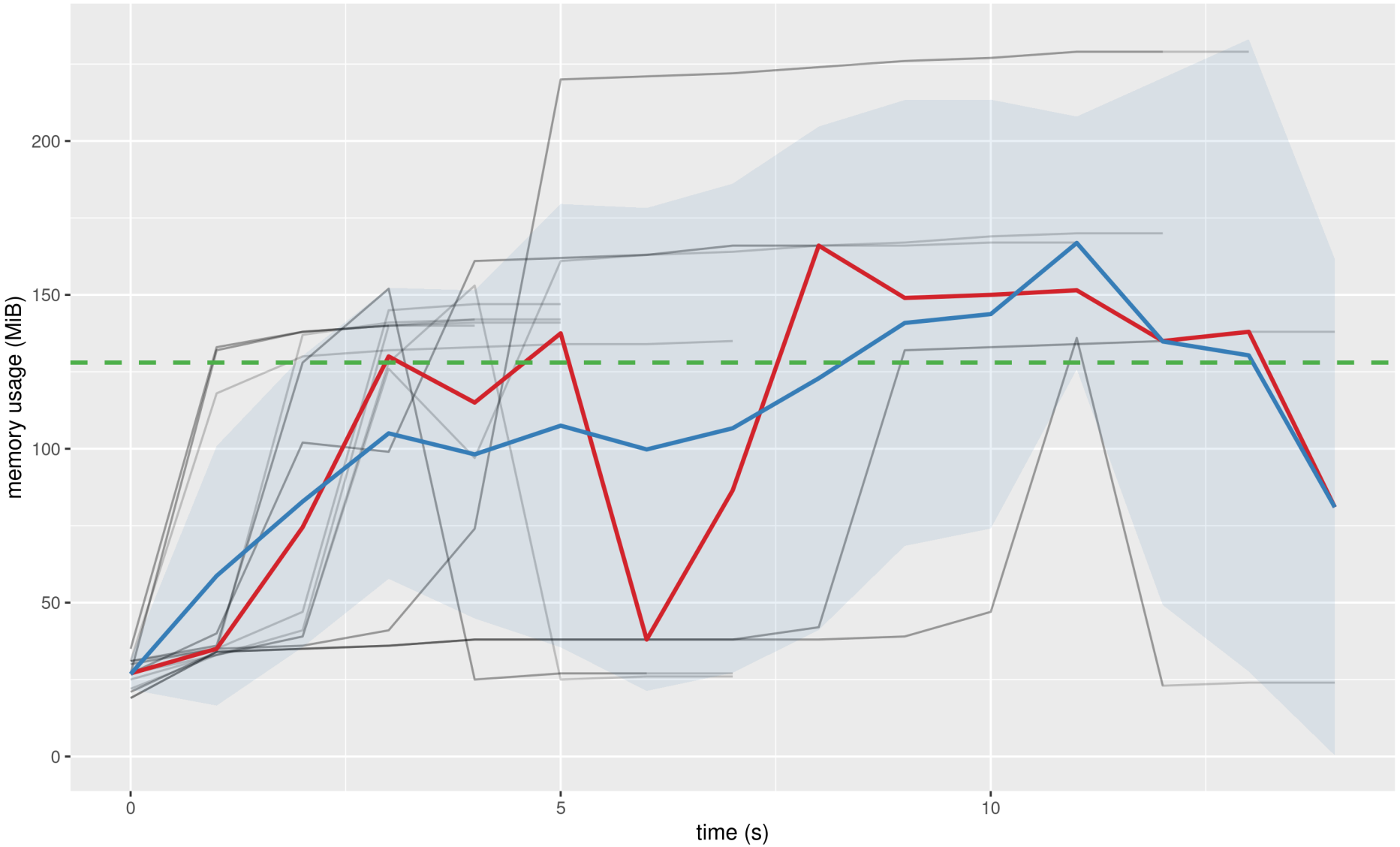
Evaluation: CPU



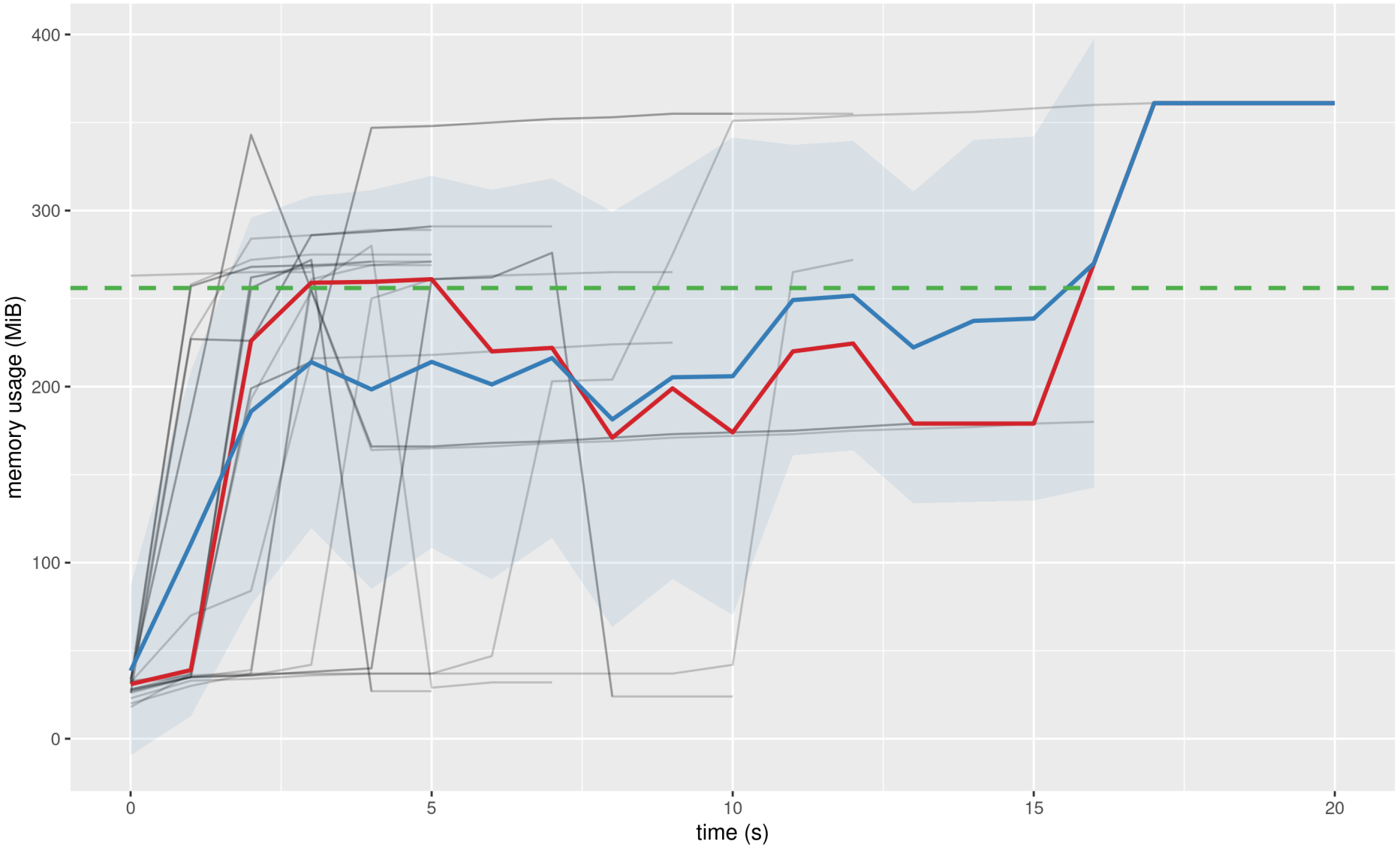
Evaluation: Memory (64 MiB)



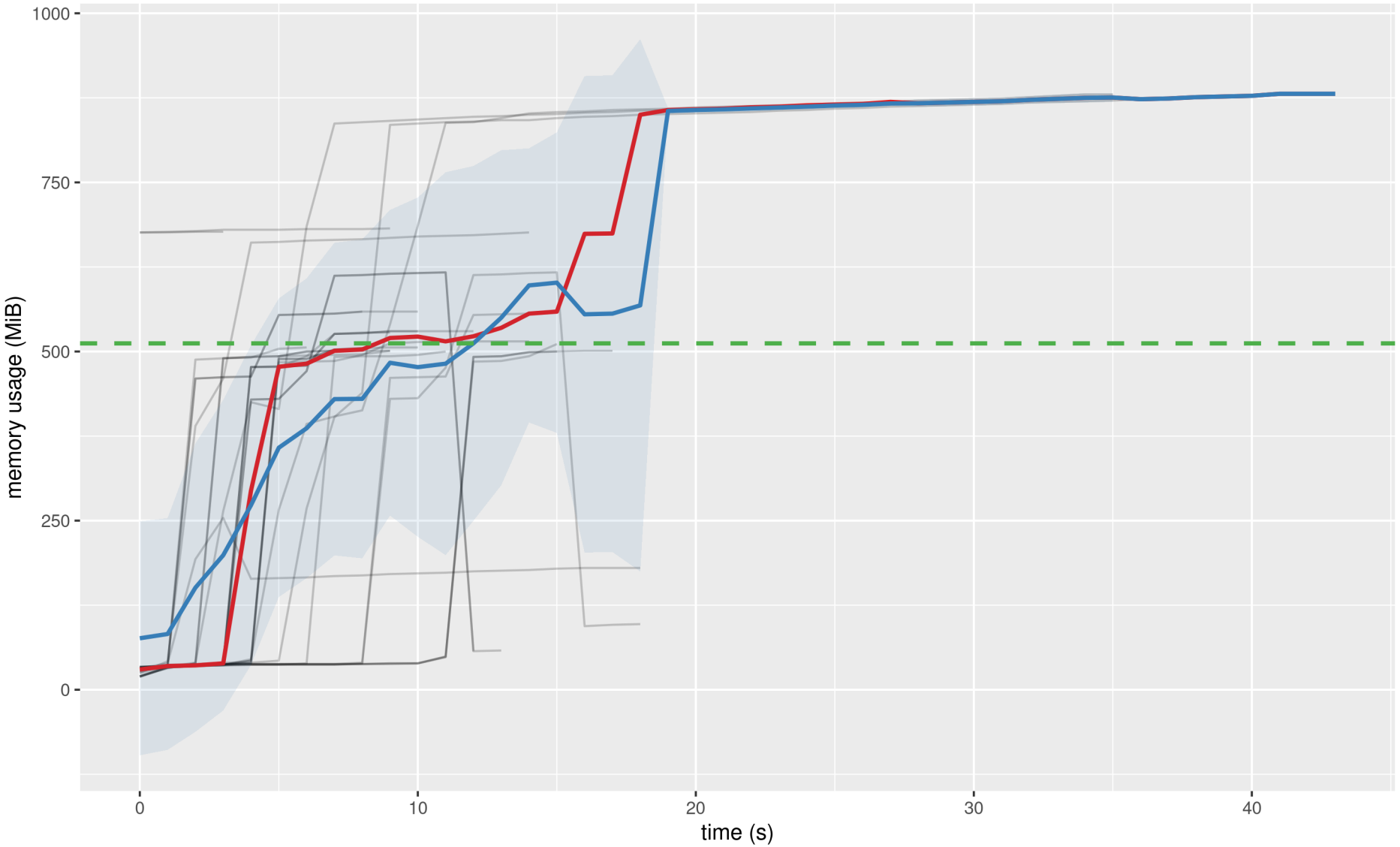
Evaluation: Memory (128 MiB)



Evaluation: Memory (256 MiB)



Evaluation: Memory (512 MiB)



Future Work

- Input/output simulation
- Complex usage patterns
- Automatically answering the question:
  - does this experiment show that the application could benefit from more resources?
- Complex component topologies
- More performance metrics (e.g. end-to-end latency)
  - and example applications

Future Work

- Input/output simulation
- Complex usage patterns
- Automatically answering the question:
  - does this experiment show that the application could benefit from more resources?
- Complex component topologies
- More performance metrics (e.g. end-to-end latency)
  - and example applications

Thank You!