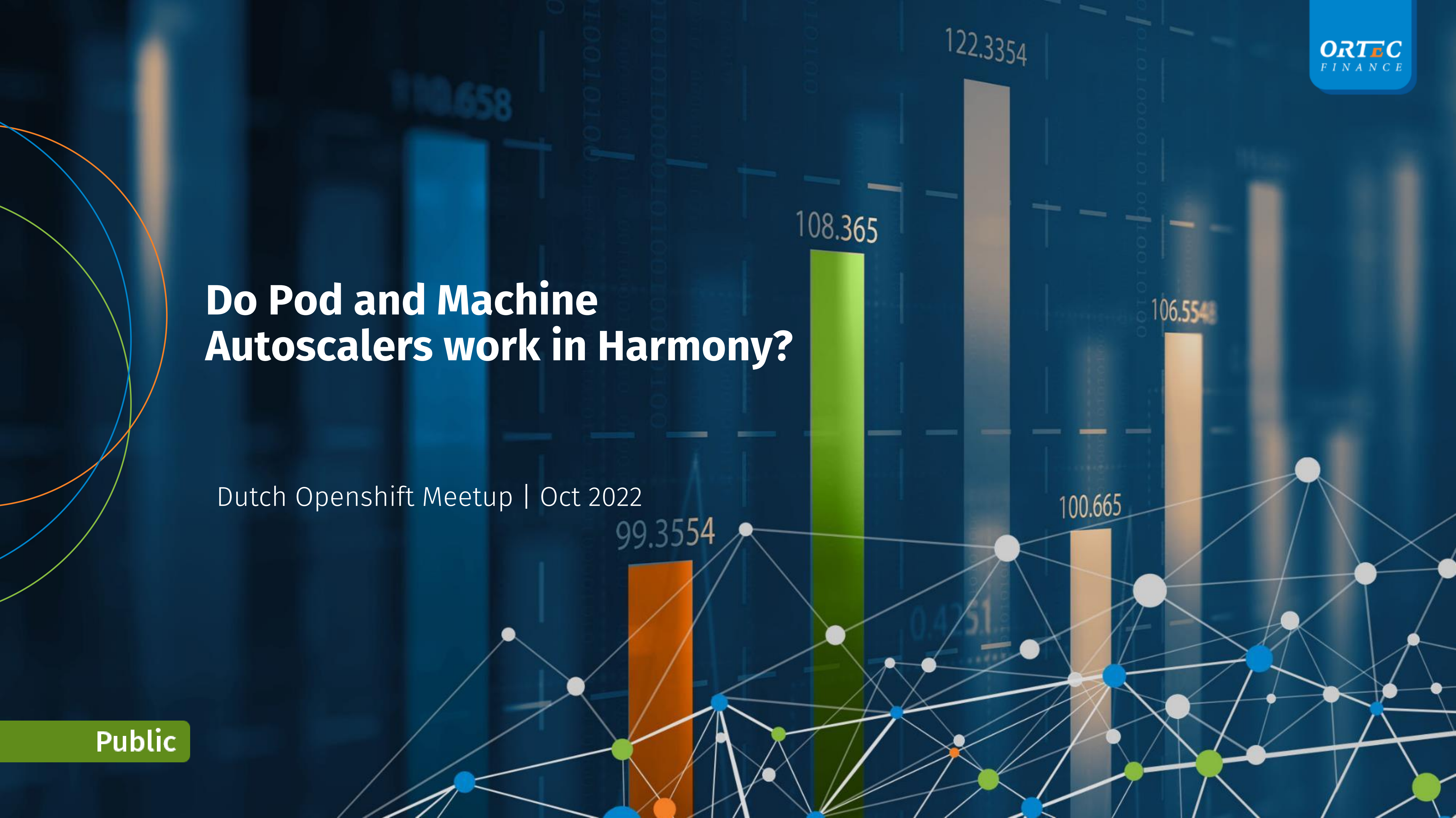
Do Pod and Machine Autoscalers work in Harmony?

Dutch Openshift Meetup | Oct 2022

Public

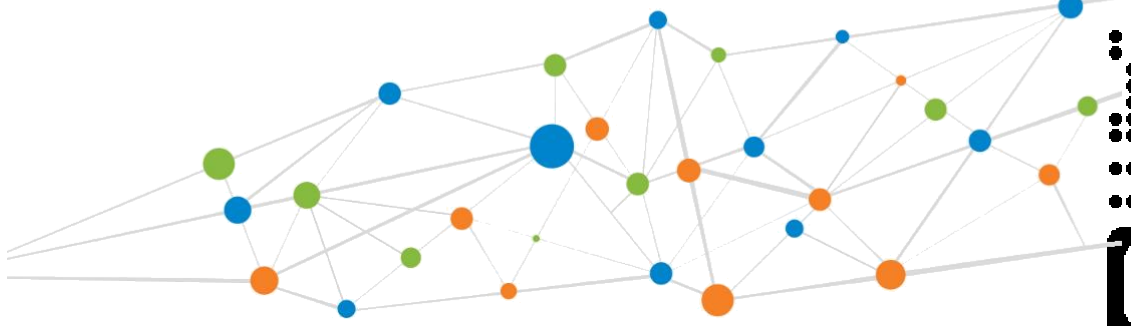


Do Pod and Machine Autoscalers work in Harmony?

A performance review of ARO and ROSA

Agenda:

- Context – What type of workloads are we talking about?
- Pod Autoscalers
  - Knative + Demo
  - KEDA + Demo
- Auto scaling Machines + Demo
  - ARO
  - ROSA



What we do & who we are

We enable people to manage the complexity of investment decision making

Ortec Finance is a global technology and solutions provider

Global client base

North America

OMERS
THE WORLD BANK
OPTrust
HOOPP Healthcare of Ontario Pension Plan
UNIVERSITY OF CALIFORNIA Office of the Chief Investment Officer

Europe

BT PENSION SCHEME
USS
AP FJÄRDE AP-FONDEN
Andra AP-fonden
a.s.r. de nederlandse verzekerings maatschappij voor alle verzekeringen
SEB
apg
ST. JAMES'S PLACE WEALTH MANAGEMENT
KEVA
PGGM
ING
pensionskasse STADT ZÜRICH
InvestmentDataServices A company of Allianz
achmea Investment Management
ABB
ROBECO The Investment Engineers
elo
CERN
ABN·AMRO
PGB pensioendiensten
AEGON
Pensioenfonds Rail & OV

Asia

Korea Fixed Income Research Institute 한국채권연구원
income made different

Pacific

QIC
acc
cbus
futurefund Australia's Sovereign Wealth Fund
QSuper
NEW ZEALAND SUPERANNUATION FUND
AustralianSuper

Our Cloud-Native Journey

- We like Managed over DIY:
  - OpenShift is a very mature, battle-tested, enterprise-level Kubernetes toolkit. It's a 'batteries-included' platform
  - We leverage opinionated Red Hat stack (e.g. prefer supported operators)

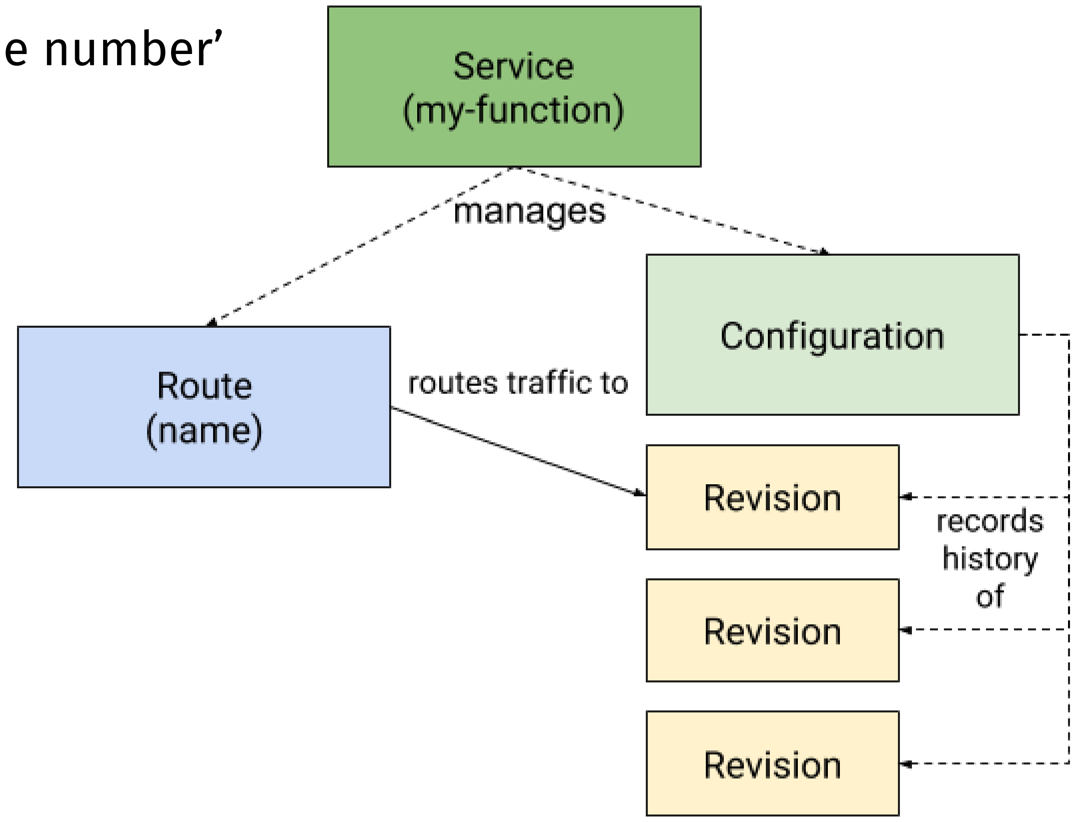
- We don't lift and shift. – Modernize Apps first

- Migrating Legacy Microsoft HPC workloads this year

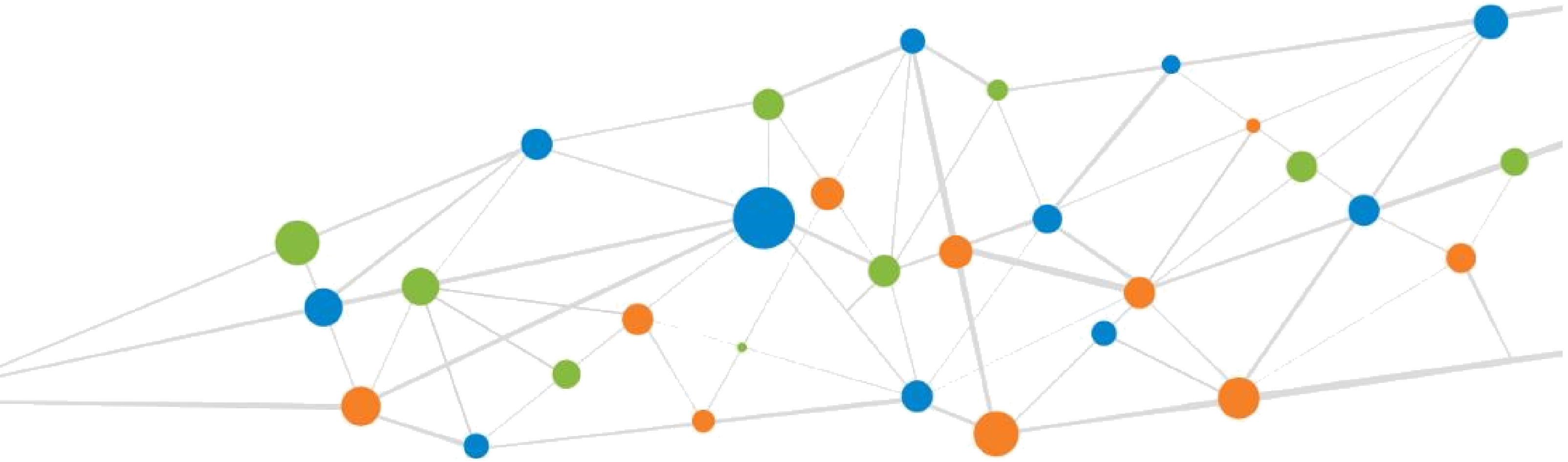
Openshift Serverless

The ability to scale to zero – “Some people call this Serverless”

- Benchmark:
  - dotnet core implementation of naive ‘highest prime number’
  - We kept it synchronous (no Knative Eventing)
  - Node selectors enabled



DEMO



ORTEC FINANCE

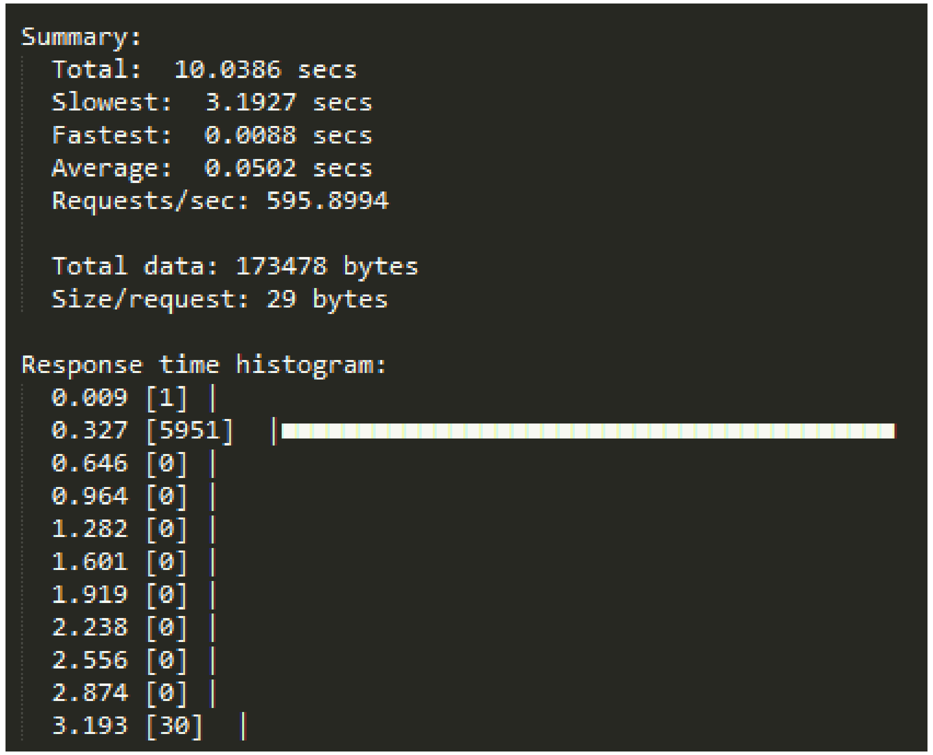
ARO vs Rosa performance review

ARO VM: Standard_E4s_v3

ROSA VMs: m5.xlarge

Benchmark: hey -z 10s -c 30 "<app_url>/100000"

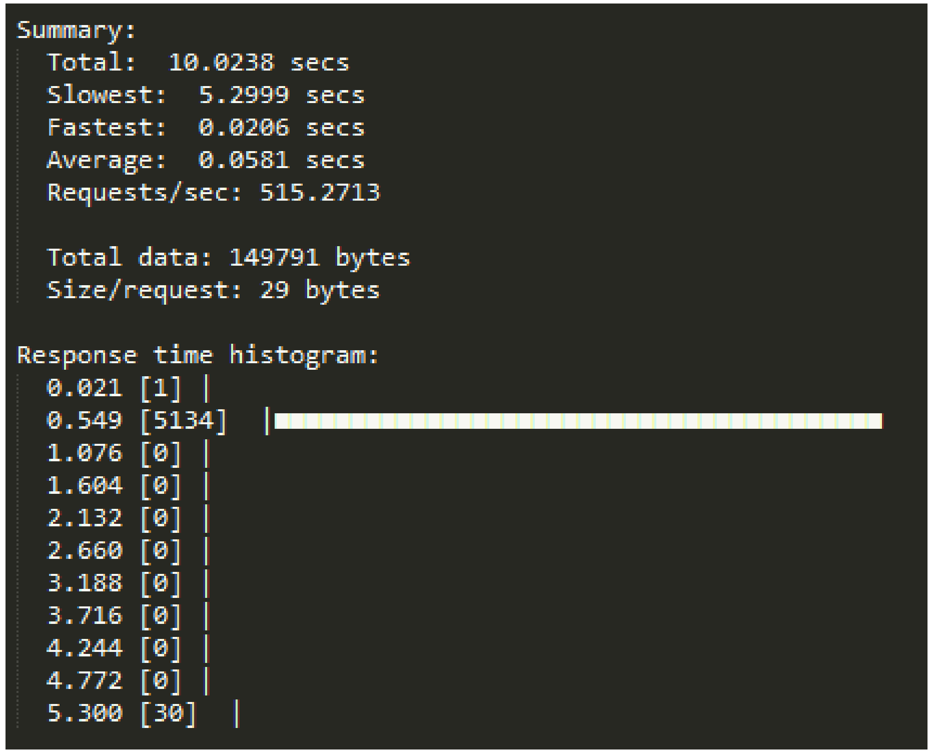
- ARO slightly faster for same workload on different VM type



```
Summary:
  Total:   10.0386 secs
  Slowest:   3.1927 secs
  Fastest:   0.0088 secs
  Average:   0.0502 secs
  Requests/sec: 595.8994

  Total data: 173478 bytes
  Size/request: 29 bytes

Response time histogram:
  0.009 [1] |
  0.327 [5951]  |■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
  0.646 [0] |
  0.964 [0] |
  1.282 [0] |
  1.601 [0] |
  1.919 [0] |
  2.238 [0] |
  2.556 [0] |
  2.874 [0] |
  3.193 [30]  |
```



```
Summary:
  Total:   10.0238 secs
  Slowest:   5.2999 secs
  Fastest:   0.0206 secs
  Average:   0.0581 secs
  Requests/sec: 515.2713

  Total data: 149791 bytes
  Size/request: 29 bytes

Response time histogram:
  0.021 [1] |
  0.549 [5134]  |■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
  1.076 [0] |
  1.604 [0] |
  2.132 [0] |
  2.660 [0] |
  3.188 [0] |
  3.716 [0] |
  4.244 [0] |
  4.772 [0] |
  5.300 [30]  |
```

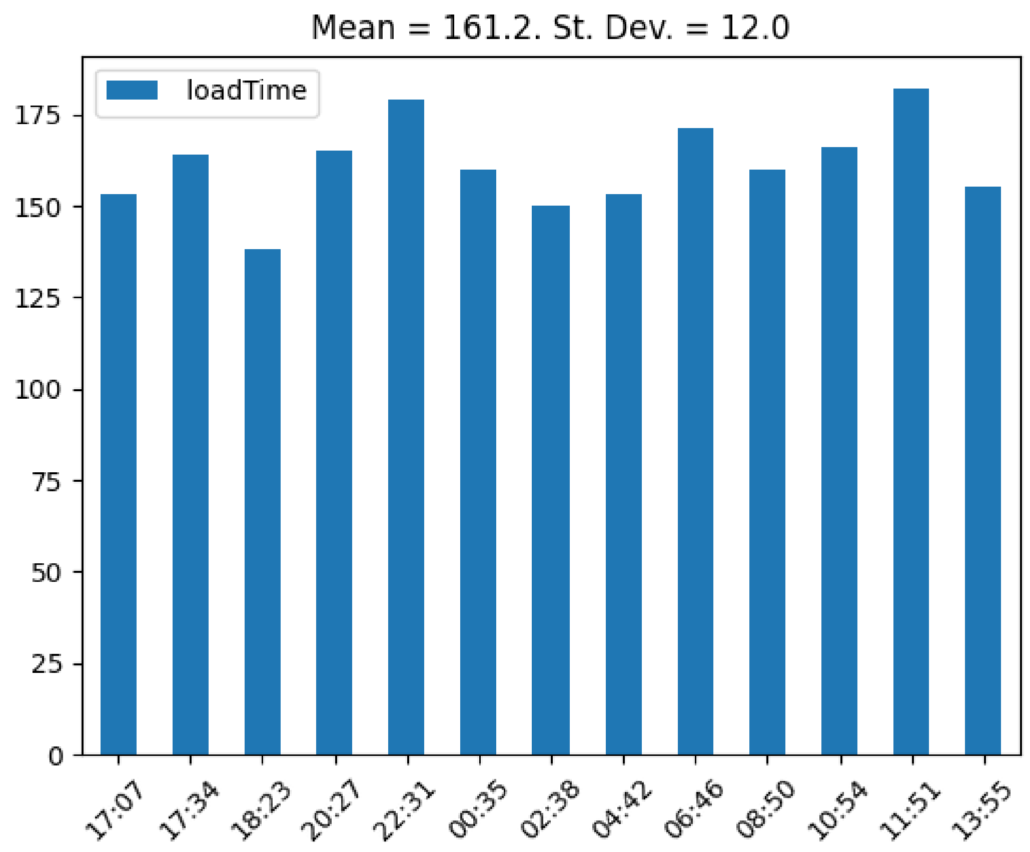
ARO vs Rosa performance review II

ROSA:

Provisioning time per machine family in seconds	
M5.xlarge	201 s

ARO:

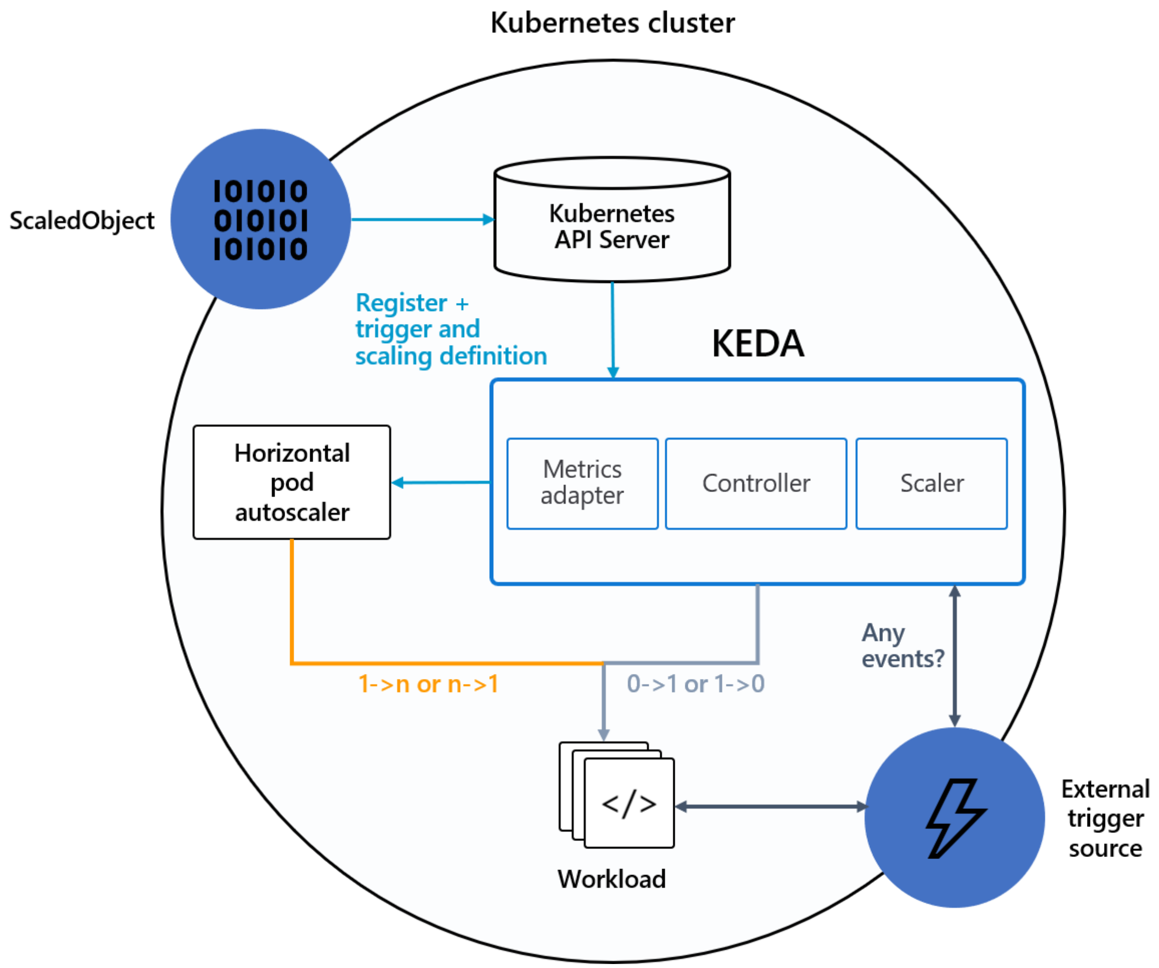
Provisioning time per machine family in seconds	
Standard_D4s_v3	161 s
Compute optimized 32 cores (F32)	230 s
Compute optimized 16 cores (F16)	232s
Compute optimized 8 cores (F8)	244 s



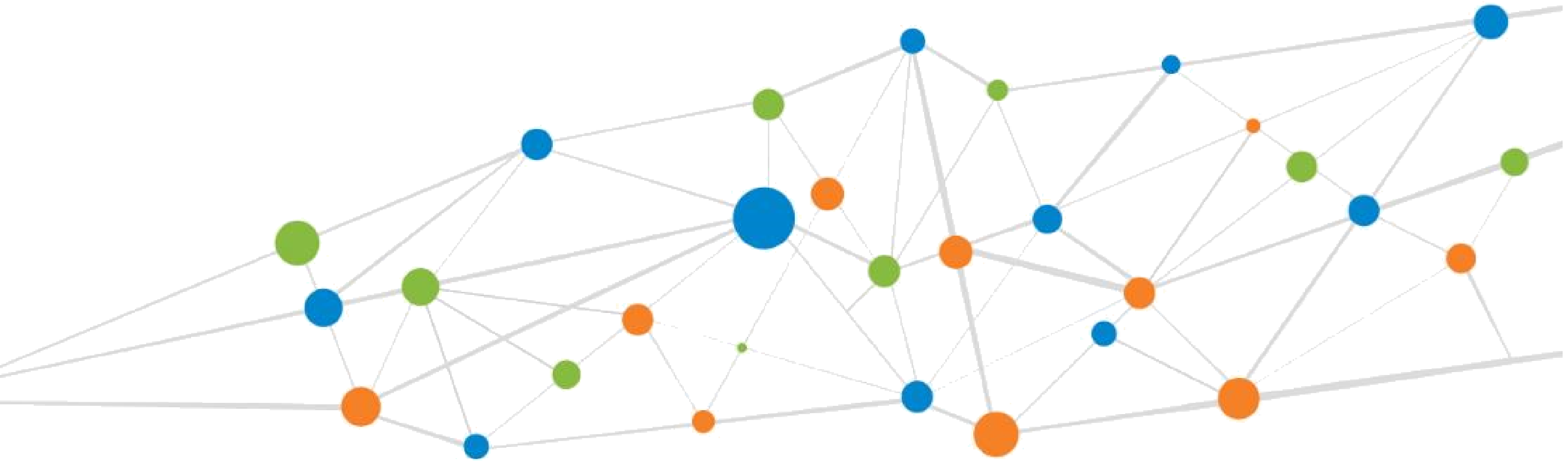
KEDA

Kubernetes Event Driven Autoscaler

- Benchmark:
  - Same prime number algo
  - Async; Producer / Consumer set up
  - No Kafka, but AMQ Broker



DEMO



KEDA

- Operates on standard k8s resources
- Can scale existing deployed apps
- Pull based approach
- Doesn't manage data delivery
- K8s Horizontal Pod Autoscaler (HPA)
- Focus is on event driven autoscaling

- Operates on Knative Service
- Existing apps must be converted
- Push based approach
- Manages data delivery (Eventing)
- Knative Autoscaler
- Demand-based autoscaling (HTTP)

https://developers.redhat.com/devnation/tech-talks/autoscaling-KEDA-knative

Wrap-up – Performance ROSA vs ARO

- ARO provisioning performance is likely to be ~10% better.
- ROSA supports Spot instances for machinesets
- ROSA machinesets can only be configured in the OCM or with the rosa cli
- ROSA cluster can hold up to 180 nodes while ARO cluster only scale to 40
- Default worker machine types are very similar in performance for both ROSA and ARO
- Long running workloads can rely on Machine autoscaling (ROSA had troubles scaling up from 0 machines)
- Short running workloads will rely on ‘one hot spare’-approach

Wrap-up – Scaling Pods from 0 to 100s

- For (long) parallel multiproces computations we prefer the **KEDA** setup
  - + Operates on standard K8 resources
  - + Can scale existing apps
  - + Tweakable autoscaling (e.g. deal with cold starts)
  - - Not all community KEDA scalers not production ready (EDA for Prometheus is shipped in 4.11)

- For (short) multithreaded Compute we want to investigate **Openshift Serverless** further:
  - + Simplified deployment syntax
  - + Also includes traffic distribution
  - - Async support requires rigorous design
  - - Not designed for long running tasks (e.g. Kourier time-outs in 300s)
  - - Overshoots in #pods (in default settings)

Contact me

Joris Cramwinckel

Head of Cloud Native Platform

+316 200 002 86

Joris.Cramwinckel@ortec-finance.com

ortecfinance.com

Rotterdam	Amsterdam	London
Pfäffikon	Toronto	New York
Melbourne		

Public