

# Cloud WorkBench

A Web-Based Framework for Benchmarking Cloud Services

Joel Scheuner

University of Zurich, Switzerland



Universität  
Zürich<sup>UZH</sup>

14.08.2014



# Cloud Computing - Essential Characteristics

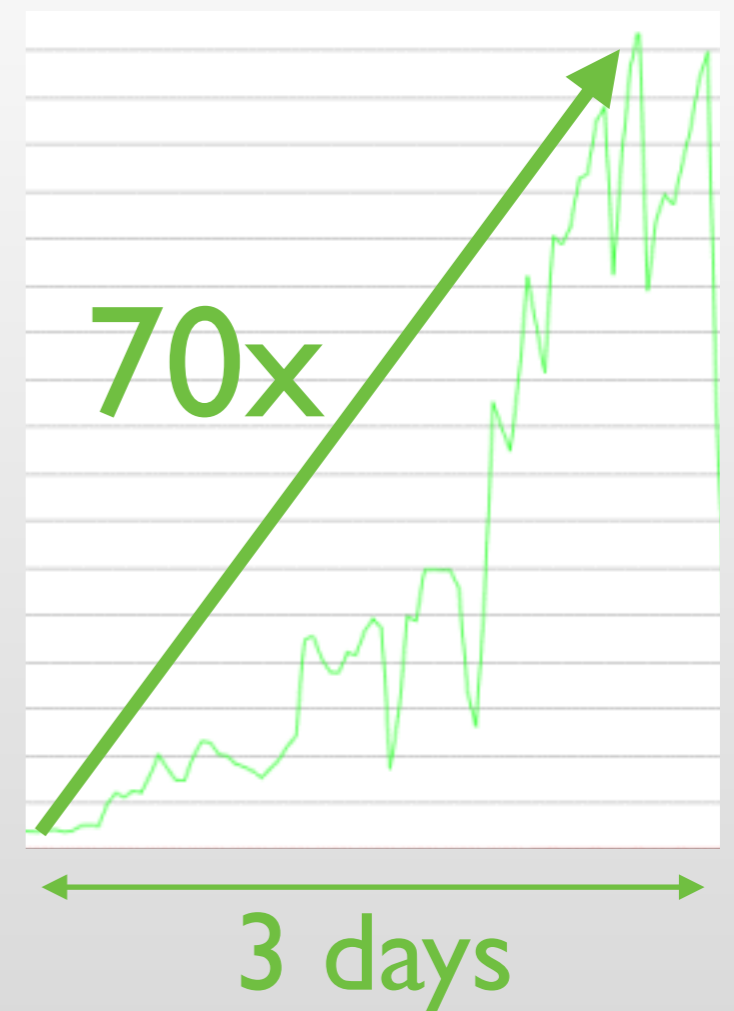
## I. On-demand Self-service

**Launch Instance**

```
ec2-run-instances \\  
ami-896c96fe \\  
-t t1.micro \\  
-k my-key-pair
```

# Cloud Computing - Essential Characteristics

1. On-demand Self-service
2. Rapid Elasticity



# Cloud Computing - Essential Characteristics

1. On-demand Self-service
2. Rapid Elasticity
3. Utility-based Pricing



# Cloud Computing - Essential Characteristics

1. On-demand Self-service
2. Rapid Elasticity
3. Utility-based Pricing
4. Resource Pooling



# Cloud Computing - Essential Characteristics

1. On-demand Self-service
2. Rapid Elasticity
3. Utility-based Pricing
4. Resource Pooling
5. Broad Network Access



# Infrastructure-as-a-Service (IaaS)

- Processing, storage, networks
- Virtual Machines (VMs)



Google Compute Engine

>23 Instance Types



# Differences between IaaS Services

- Performance
- Hardware being served
- Reliability
- Costs

Even for services  
with the same  
specification!

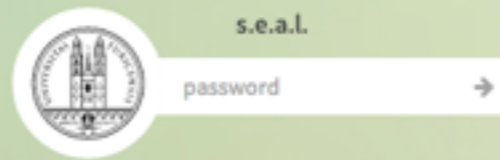
# Benchmark

- Performance test
- Types
  - Micro-Benchmarks
  - Application Benchmarks



# Demo

16:00:00



# Cloud WorkBench

The screenshot displays the Cloud WorkBench dashboard for an execution on May 30, 2014. The main status is 'FINISHED' with a green checkmark. Key metrics are shown in colored boxes: '6 minutes' for Benchmark Duration and '15 minutes' for Execution Duration. A central log window shows the following output:

```
compile-benchmark.rb  
[2014-05-29T22:08:47+00:00] INFO: cookbook_file[/usr/local/cloud-benchmark/cloudscale-compile-benchmark.rb] owner changed to 1000  
[2014-05-29T22:08:47+00:00] INFO: cookbook_file[/usr/local/cloud-benchmark/cloudscale-compile-benchmark.rb] group changed to 1000  
[2014-05-29T22:08:47+00:00] INFO: template[/usr/local/cloud-benchmark/start.sh] created file /usr/local/cloud-benchmark/start.sh  
[2014-05-29T22:08:47+00:00] INFO: template[/usr/local/cloud-benchmark/start.sh] group changed to 1000  
[2014-05-29T22:08:47+00:00] INFO: template[/usr/local/cloud-benchmark/start.sh] mode changed to 755  
[2014-05-29T22:08:47+00:00] INFO: Chef Run complete in 188.462181994 seconds  
[2014-05-29T22:08:47+00:00] INFO: Running report handlers  
[2014-05-29T22:08:47+00:00] INFO: Report handlers complete
```

The interface also includes a sidebar with navigation options (Dashboard, BENCHMARK, OTHERS), a 'Start Execution' button, a 'Delete' button, and a 'Schedule' section with a cron expression '15 3,7,11,15,19,23 \* \* \*' and a 'Metrics' section with a '+ Create New Metric Definition' button.

Open Source

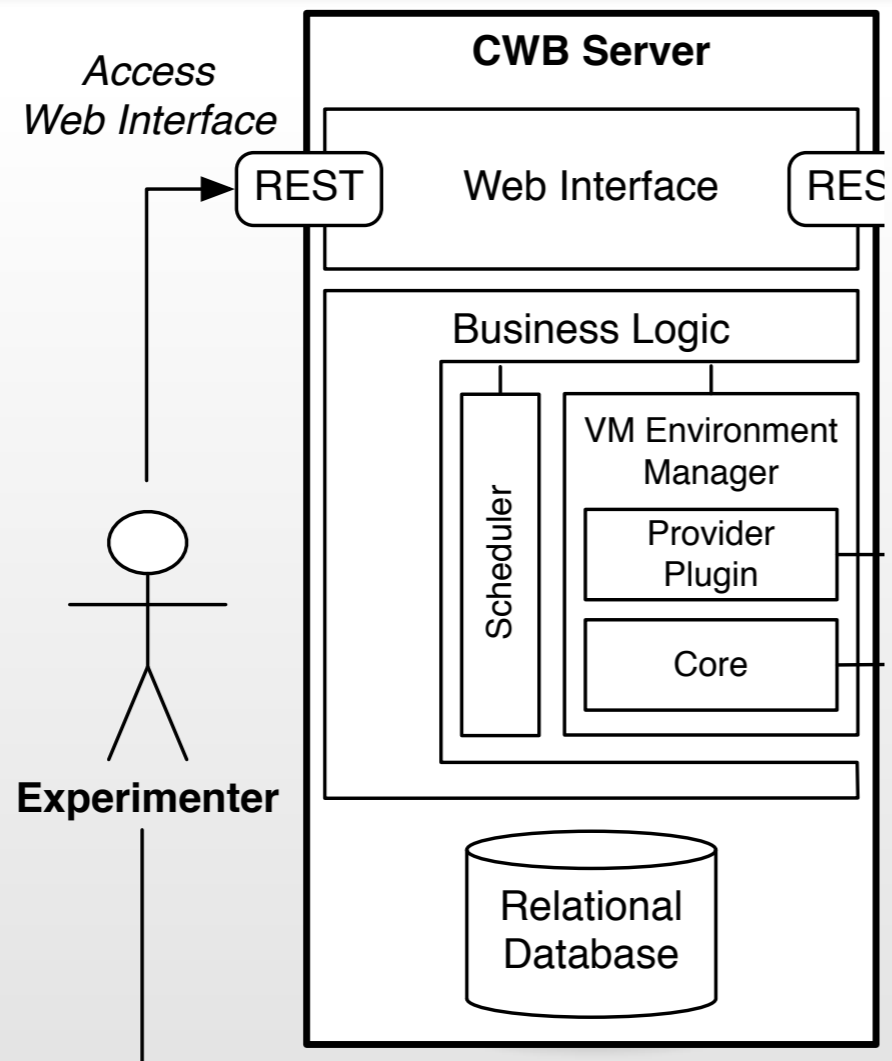
<https://github.com/sealuzh/cloud-workbench>

# Research Questions

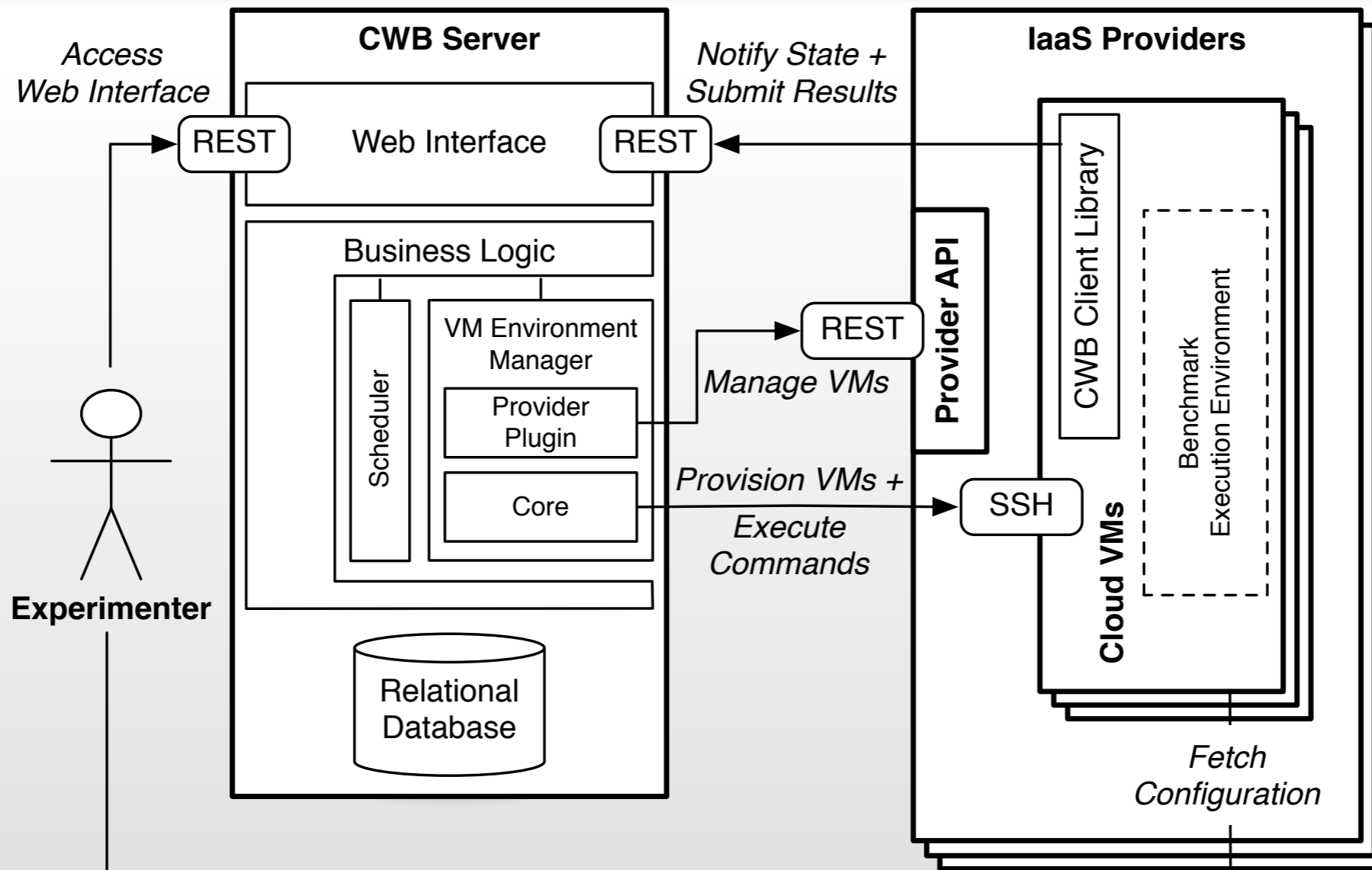
I *How can common IaaS cloud benchmarks from literature be defined in a modular and portable manner?*

II *How can benchmarks from RQ1 be periodically scheduled and reproducibly executed in cloud environments without manual interaction?*

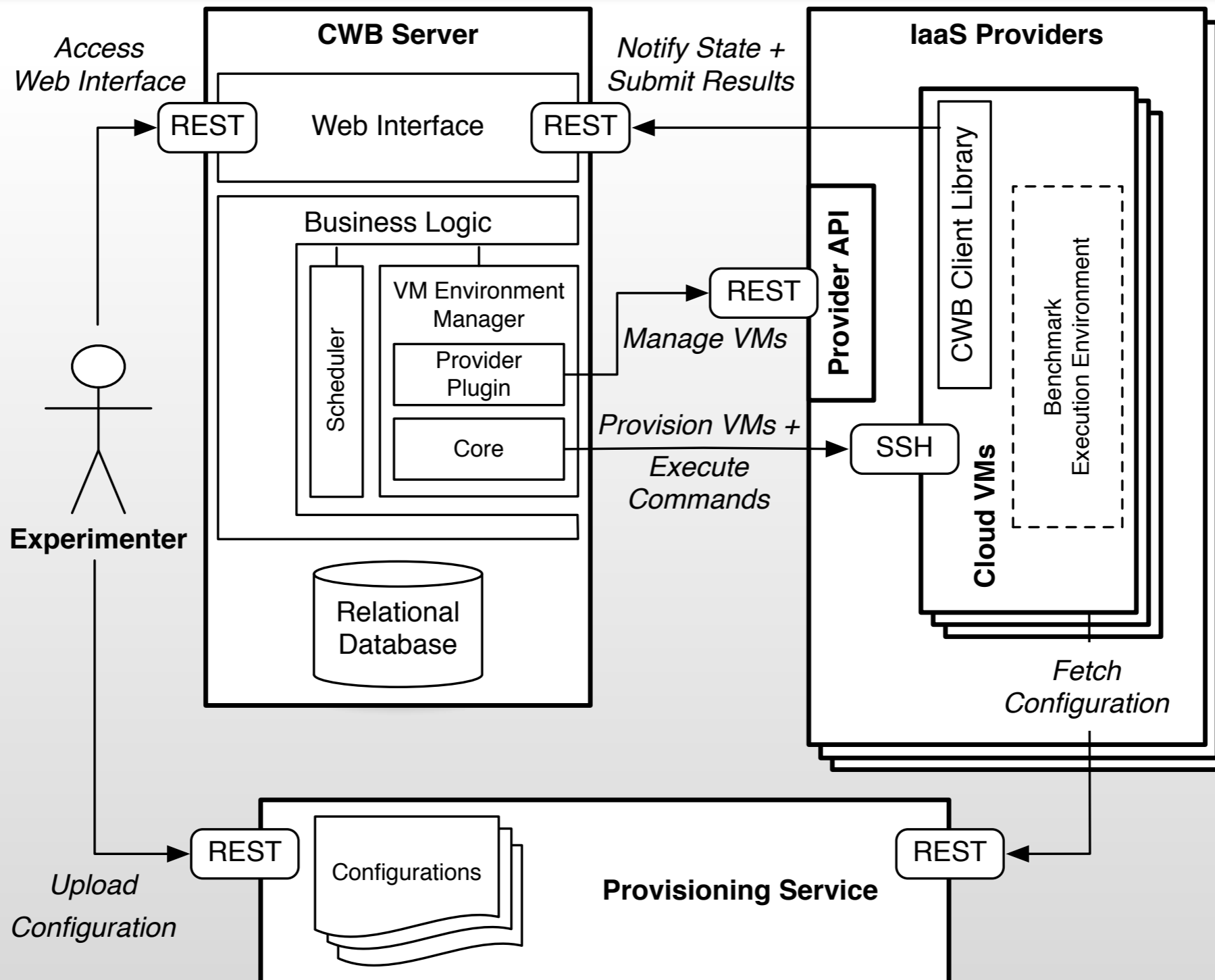
# Overall Architecture



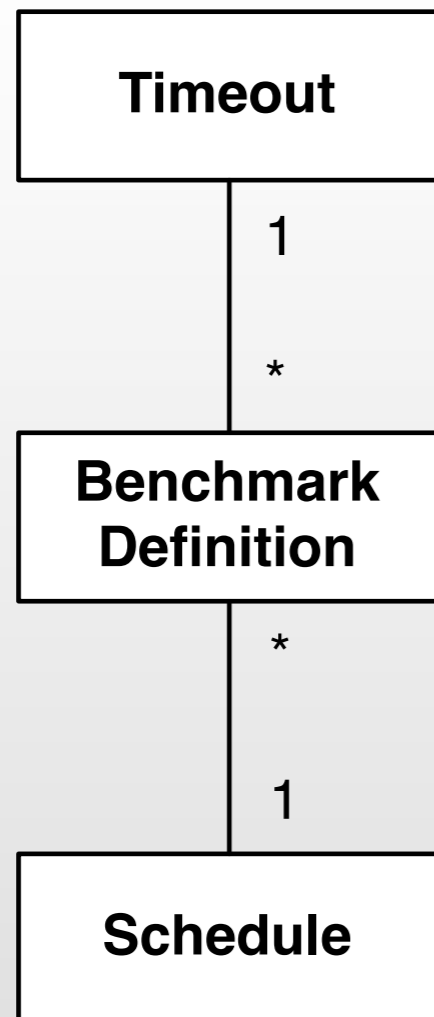
# Overall Architecture



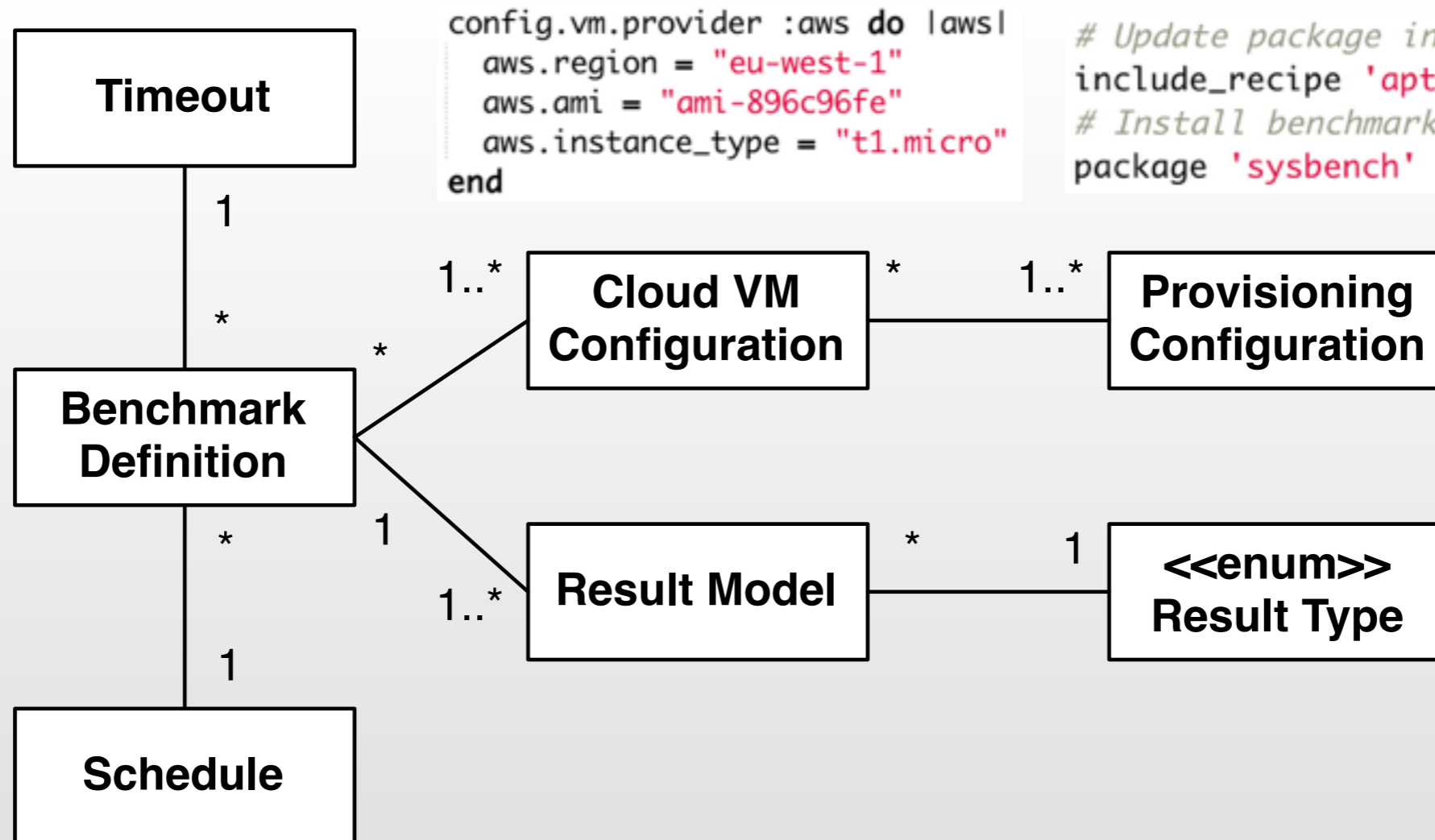
# Overall Architecture



# Benchmark Anatomy



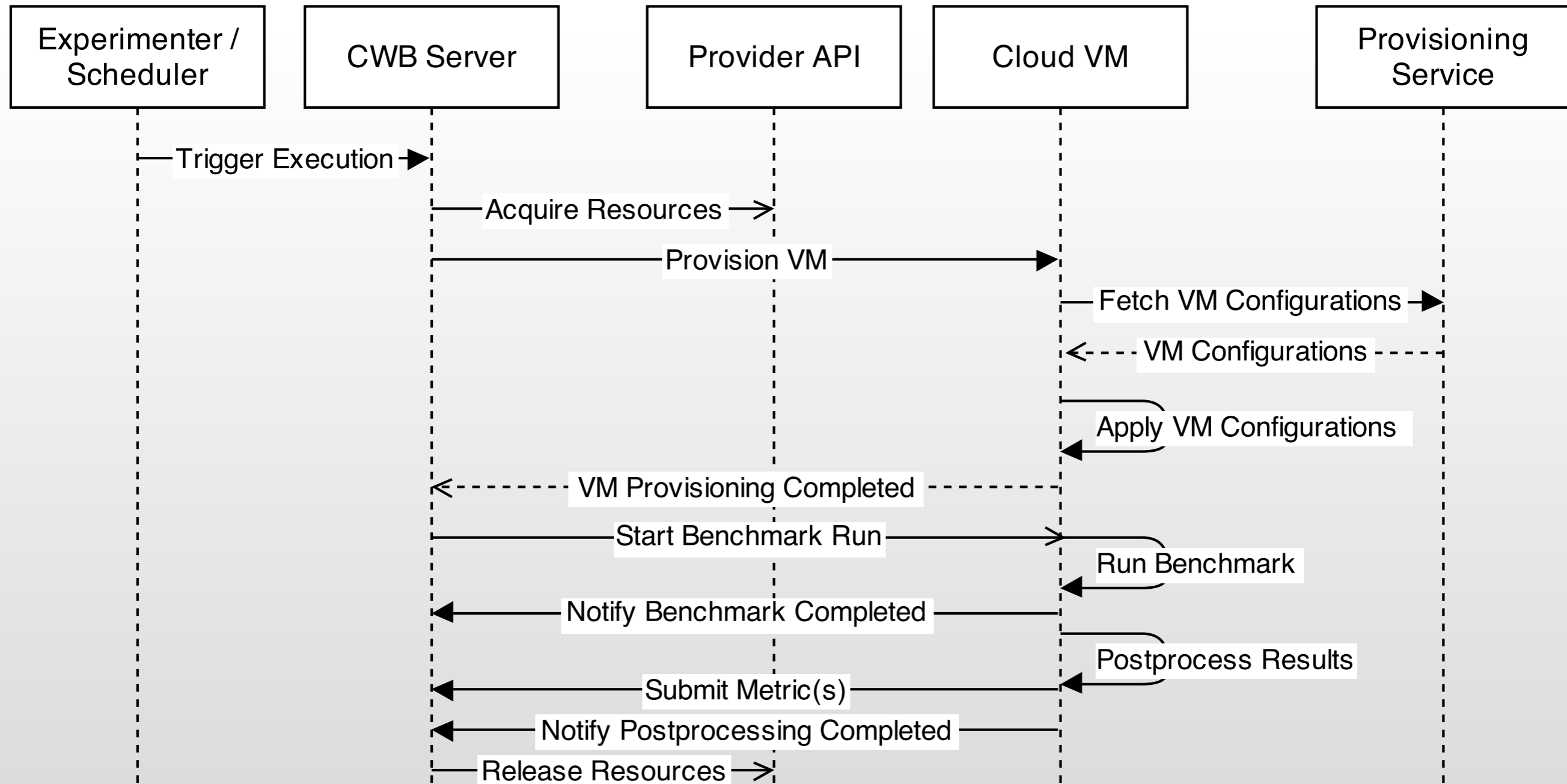
# Benchmark Anatomy



```
config.vm.provider :aws do |aws|
  aws.region = "eu-west-1"
  aws.ami = "ami-896c96fe"
  aws.instance_type = "t1.micro"
end
```

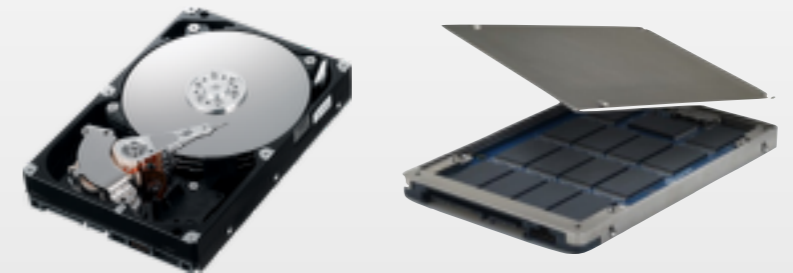
```
# Update package index
include_recipe 'apt'
# Install benchmark via package manager
package 'sysbench'
```

# Benchmark Execution



# Case Study

- Raw sequential write speed
- HDD vs. SSD storage
- Different instance types



# Questions

1

*When do larger instance types perform better than smaller instance types?*

2

*When should larger instance types be preferred over the better block storage type?*

3

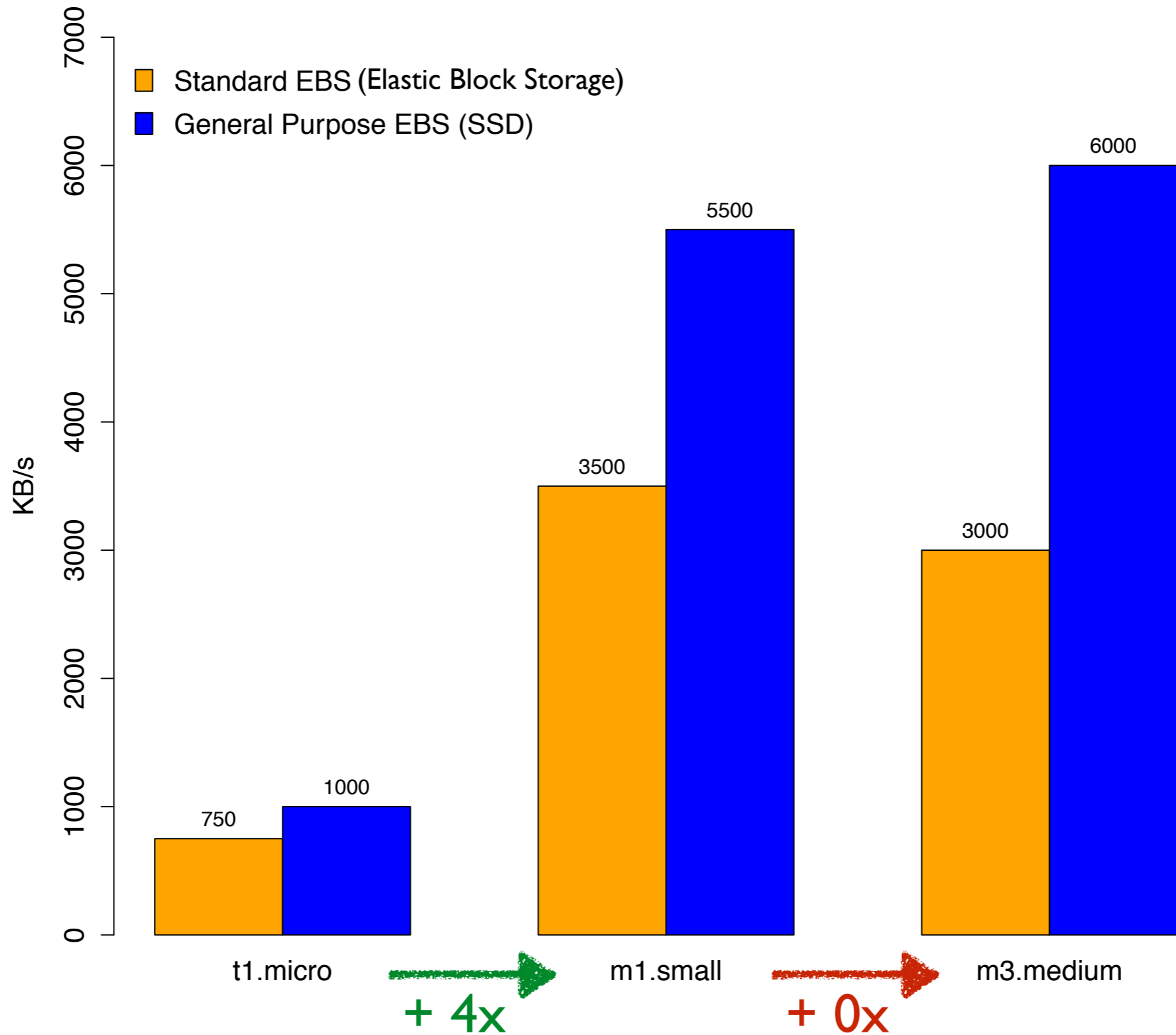
*How do instance types and block storage types influence performance variability?*

# Experiment Setup

- Amazon EC2 Ireland (eu-west-1)
- Ubuntu 14.04 (trusty)
- FIO benchmark
- 20. - 23. June 2014
- 8 - 12 repetitions

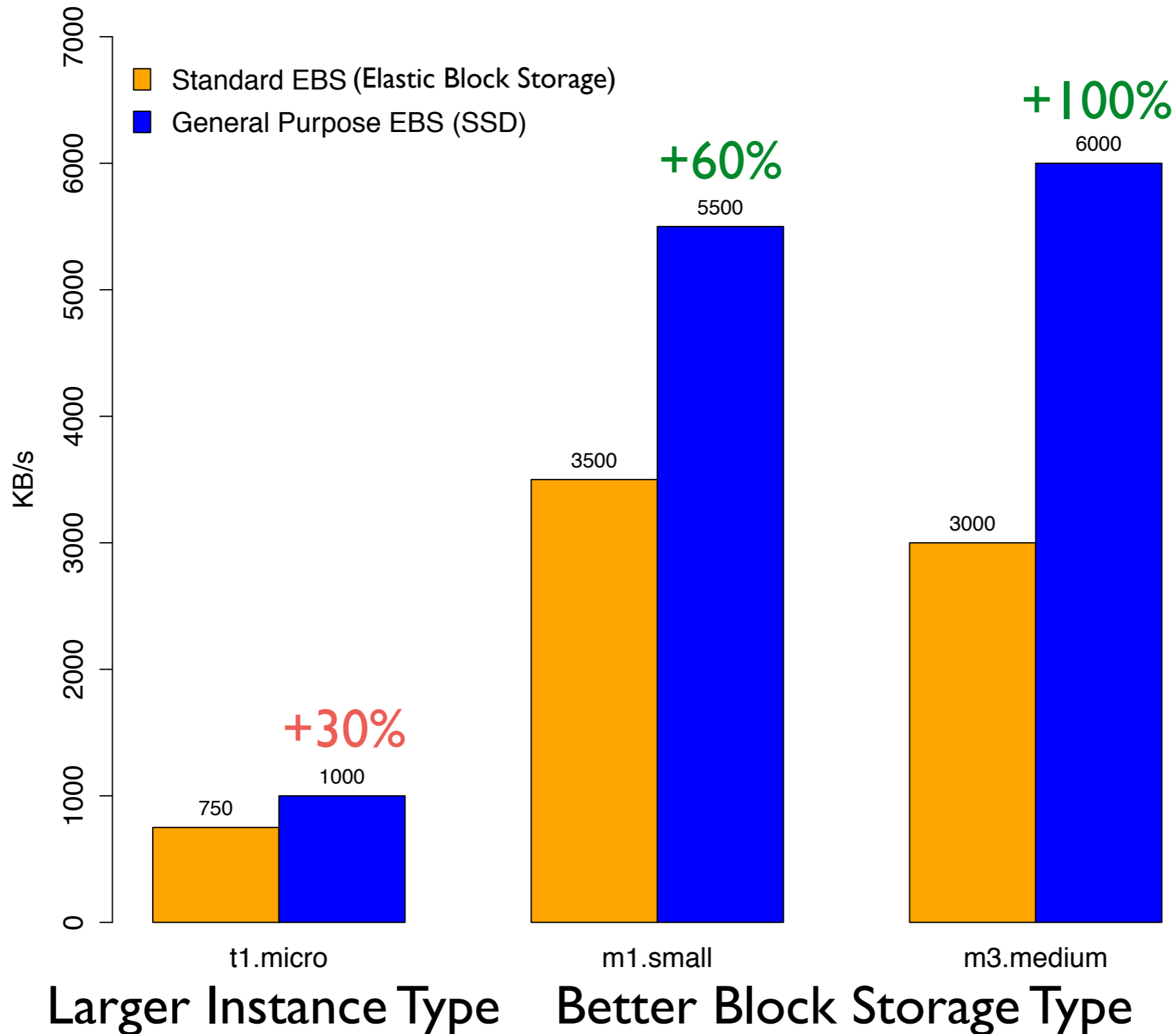
Instance Type	Price per Hour
t1.micro	\$ 0.020
m1.small	\$ 0.047
m3.medium	\$ 0.077

# When do larger instance types perform better than smaller instance types?



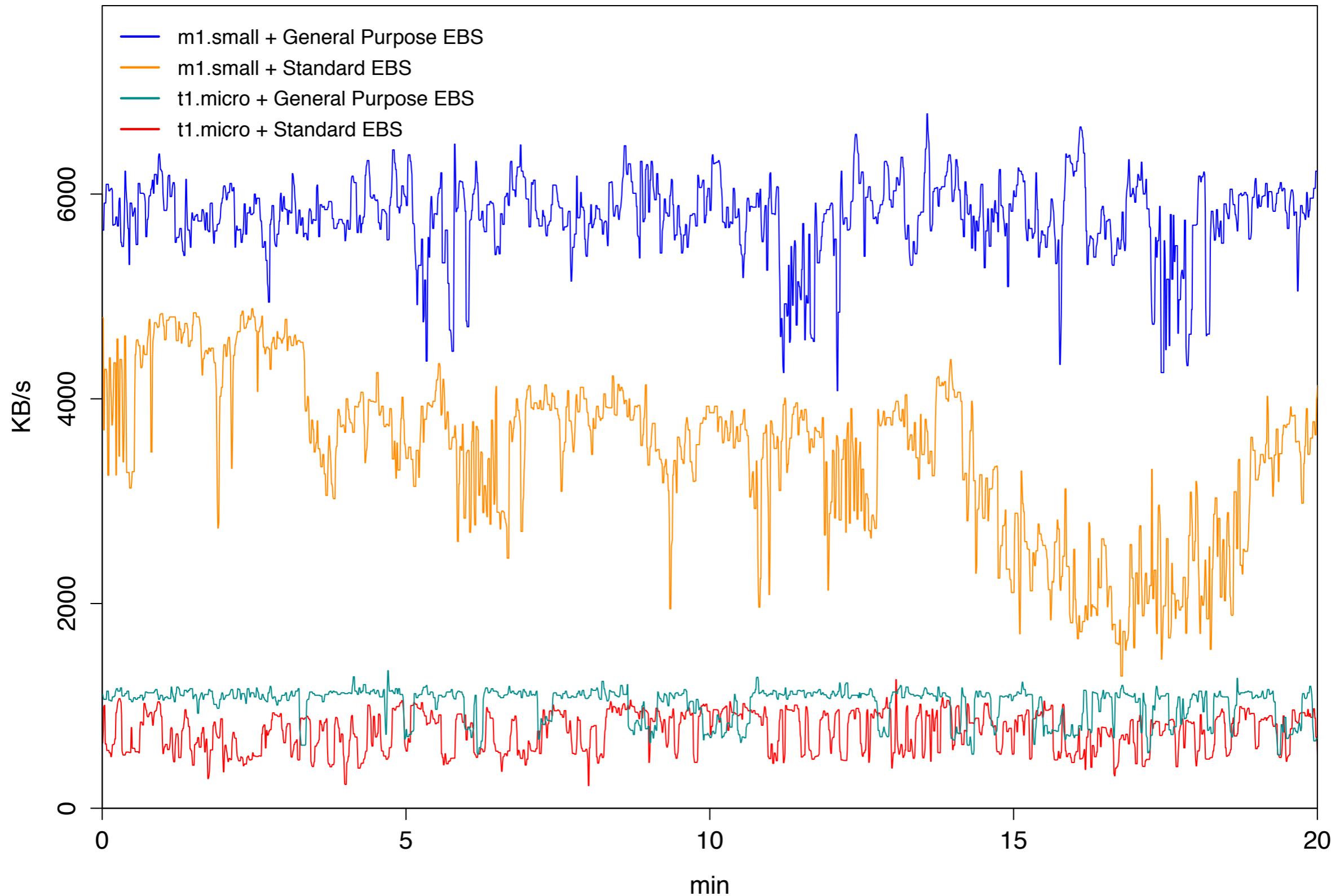
# 2

*When should larger instance types be preferred over the better block storage type?*



# 3

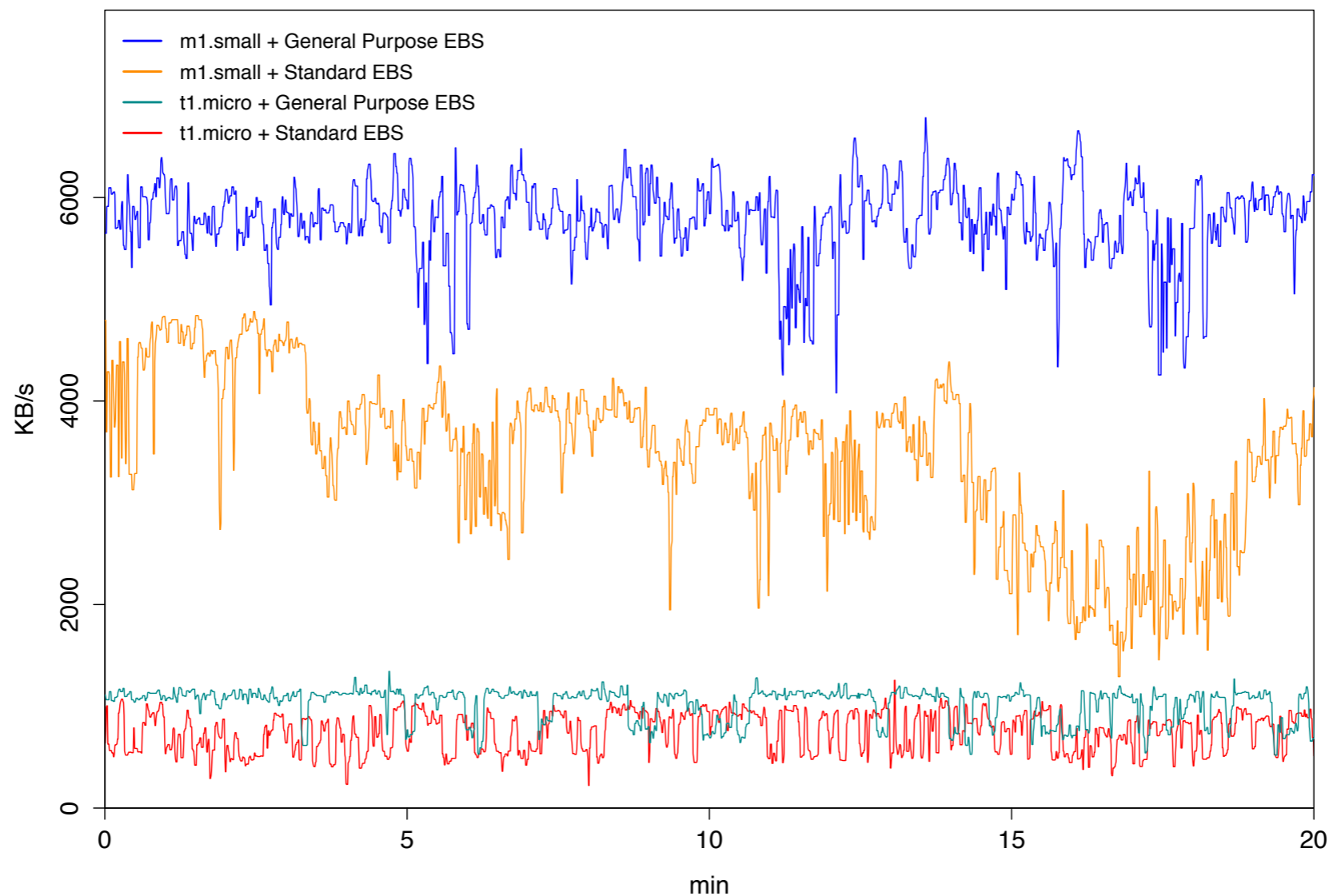
*How do instance types and block storage types influence performance variability?*



# 3

*How do instance types and block storage types influence performance variability?*

s in % of $\bar{x}$	t1.micro	m1.small	m3.medium
<b>Standard EBS</b>	<b>20%</b> (20-50%)	<b>20%</b> (10-20%)	<b>30%</b> (15-60%)
<b>General Purpose EBS</b>	<b>10%</b> (20-40%)	<b>10%</b> (5-15%)	<b>10%</b> (5-10%)



# Conclusion I

*How can common IaaS cloud benchmarks from literature be defined in a modular and portable manner?*

- Entirely define benchmarks by means of code
- Apply common software engineering techniques
- Make components and benchmarks configurable

# Conclusion II

II *How can benchmarks from RQI be periodically scheduled and reproducibly executed in cloud environments without manual interaction?*

- Use system utilities and existing tools to build a fully automated benchmark execution environment
- Eliminate any error-prone human interactions threatening reproducibility

# Outlook

- Large-scale evaluation for a World Wide Web Conference paper
- Extend CWB in a master project
  - Web-based benchmarking studio
  - Support entire lifecycle