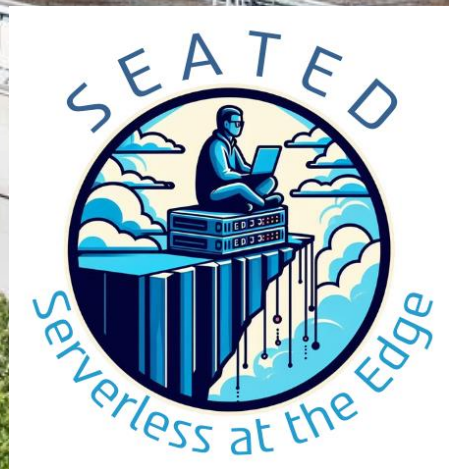
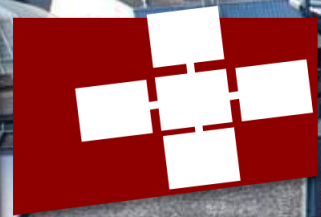


Marcin Copik, Alexandru Calotoiu, Michał Podstawski, Laurin Brandner, Larissa Schmid, Nico Graf, Grzegorz Kwaśniewski, Paweł Żuk, Sascha Kehrlı, Torsten Hoefler, and many others

Evaluating FaaS Systems with the Serverless Benchmark Suite SeBS



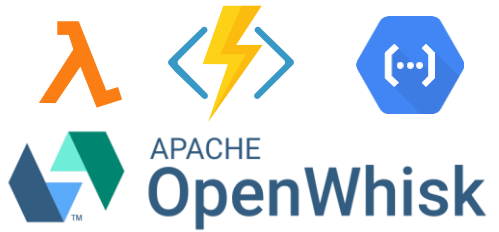
ACM HPDC 2024
Pisa, Italy

SeBS: The Serverless Benchmark Suite

 [spcl/serverless-benchmarks](https://github.com/spcl/serverless-benchmarks)

SeBS: The Serverless Benchmark Suite

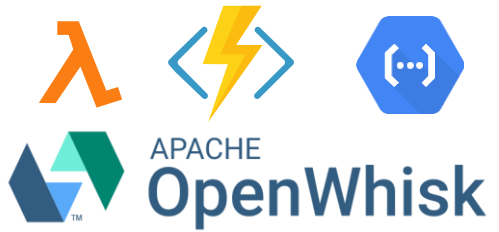
Cloud-Agnostic



 [spcl/serverless-benchmarks](https://github.com/spcl/serverless-benchmarks)

SeBS: The Serverless Benchmark Suite

Cloud-Agnostic



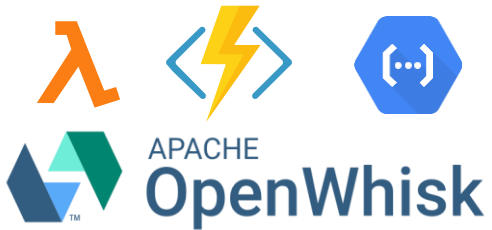
Representative Benchmarks



 [spcl/serverless-benchmarks](https://github.com/spcl/serverless-benchmarks)

SeBS: The Serverless Benchmark Suite

Cloud-Agnostic



Representative
Benchmarks



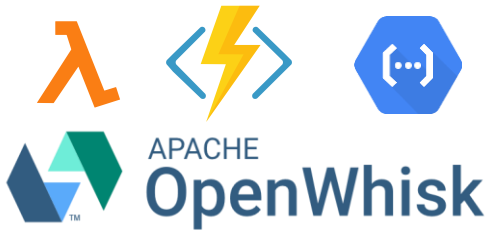
Reproducible
Experiments

Performance & Cost
Invocation Overhead
Container Eviction

 [spcl/serverless-benchmarks](https://github.com/spcl/serverless-benchmarks)

SeBS: The Serverless Benchmark Suite

Cloud-Agnostic



Representative
Benchmarks




Reproducible
Experiments

Performance & Cost
Invocation Overhead
Container Eviction

Adoption & Community


122 citations

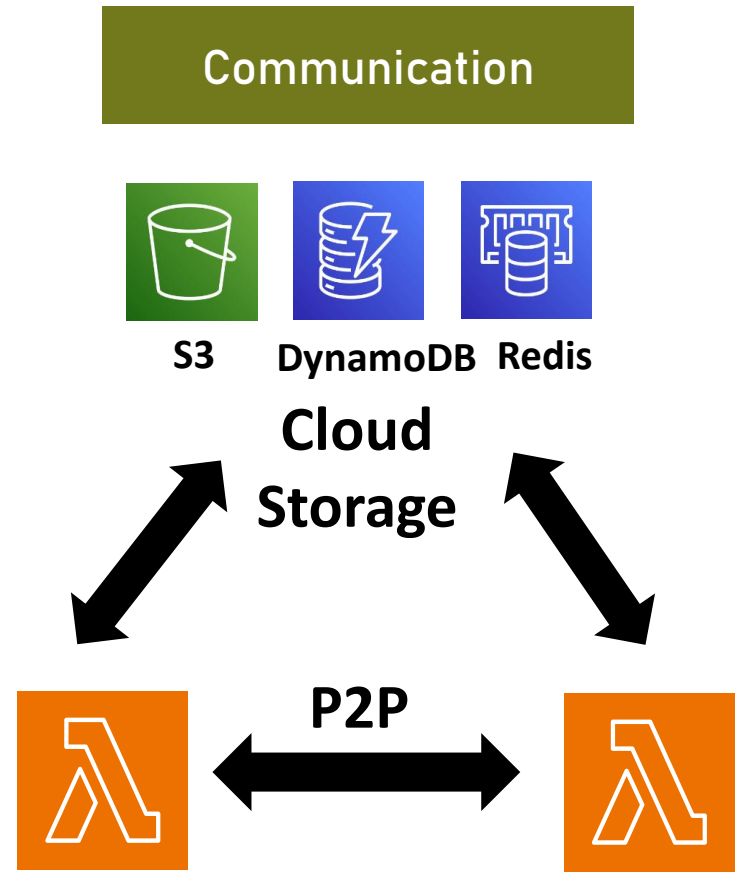
 122 stars
61 forks
16 contributors


Google
Summer of Code

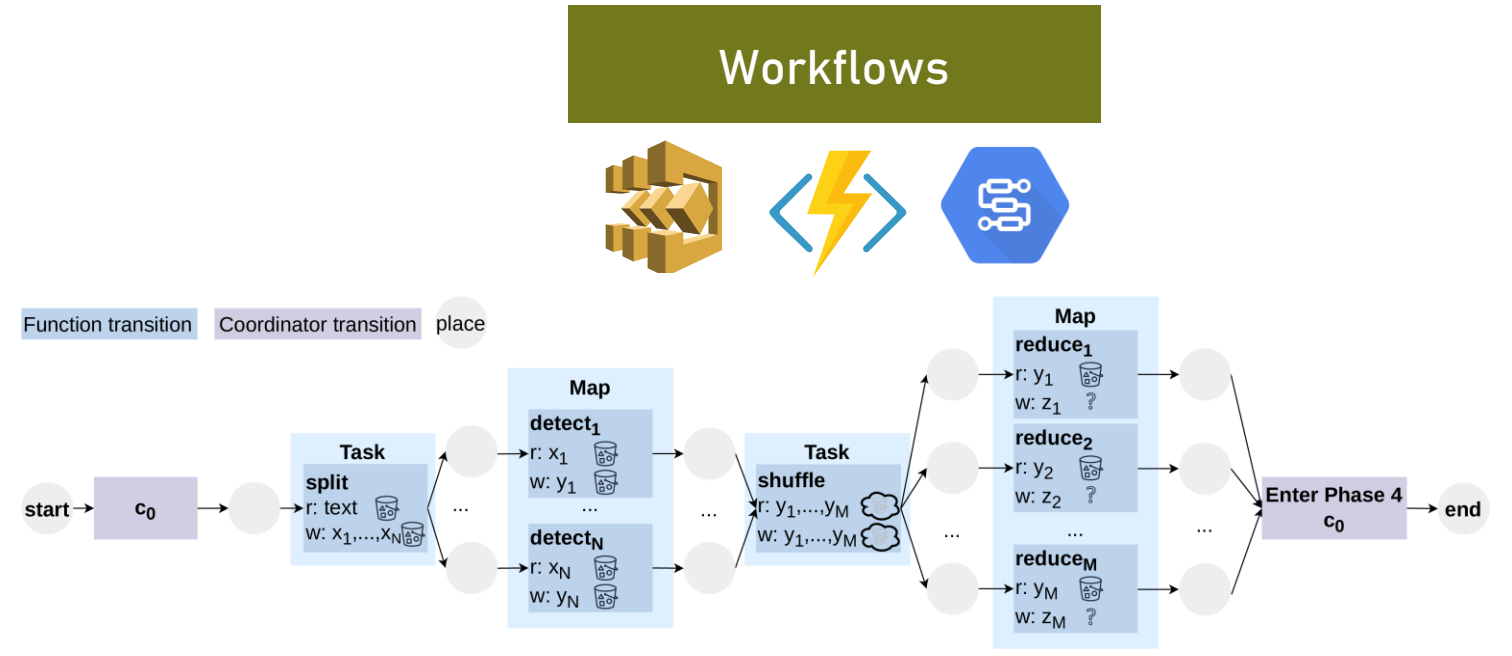
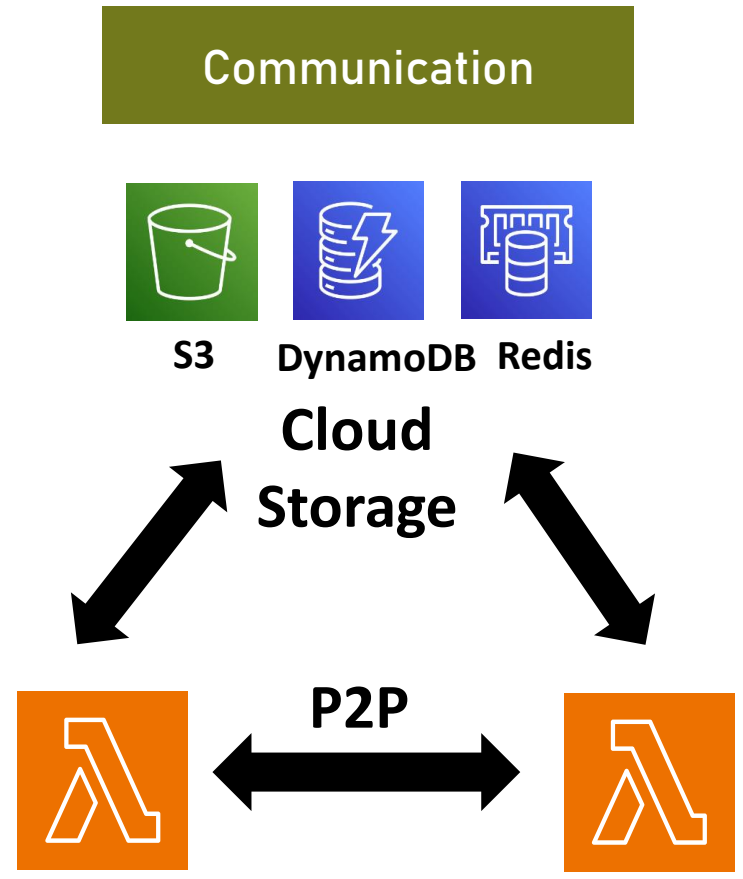
 [spcl/serverless-benchmarks](https://github.com/spcl/serverless-benchmarks)

Serverless has been changing – and so did we!

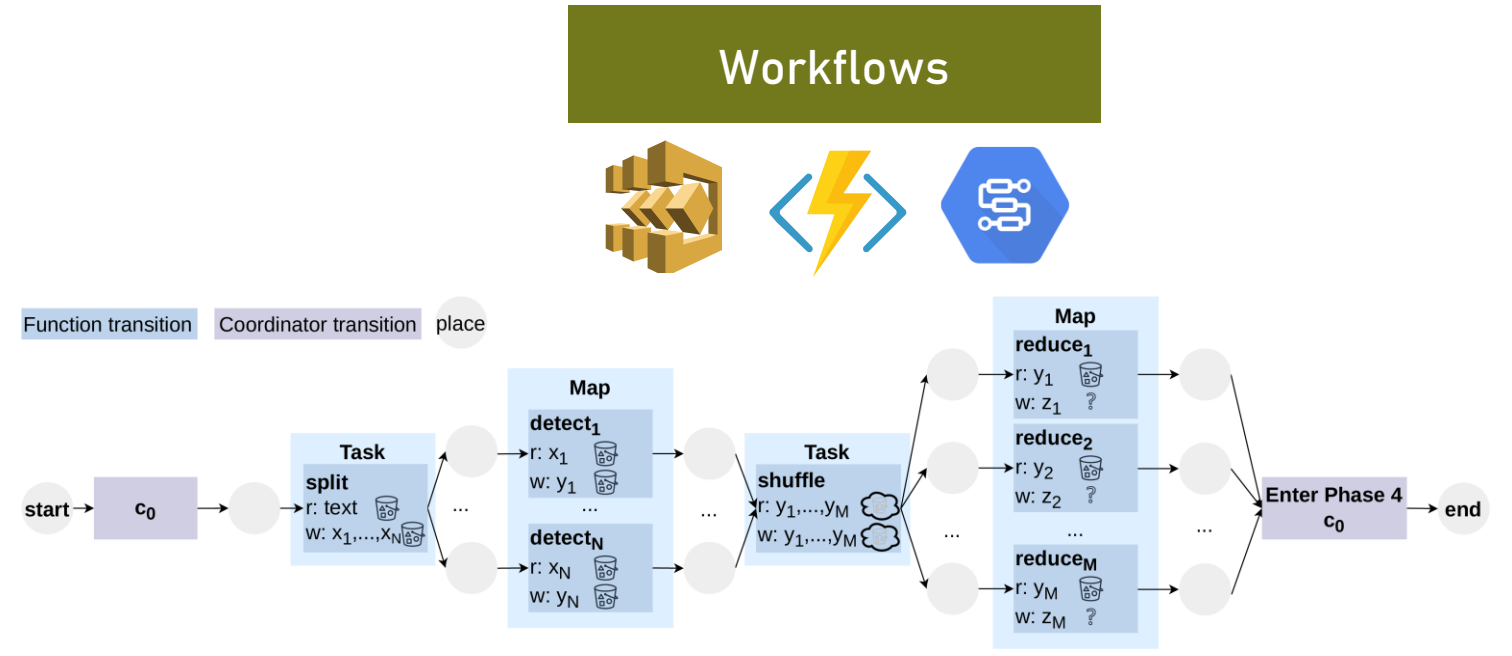
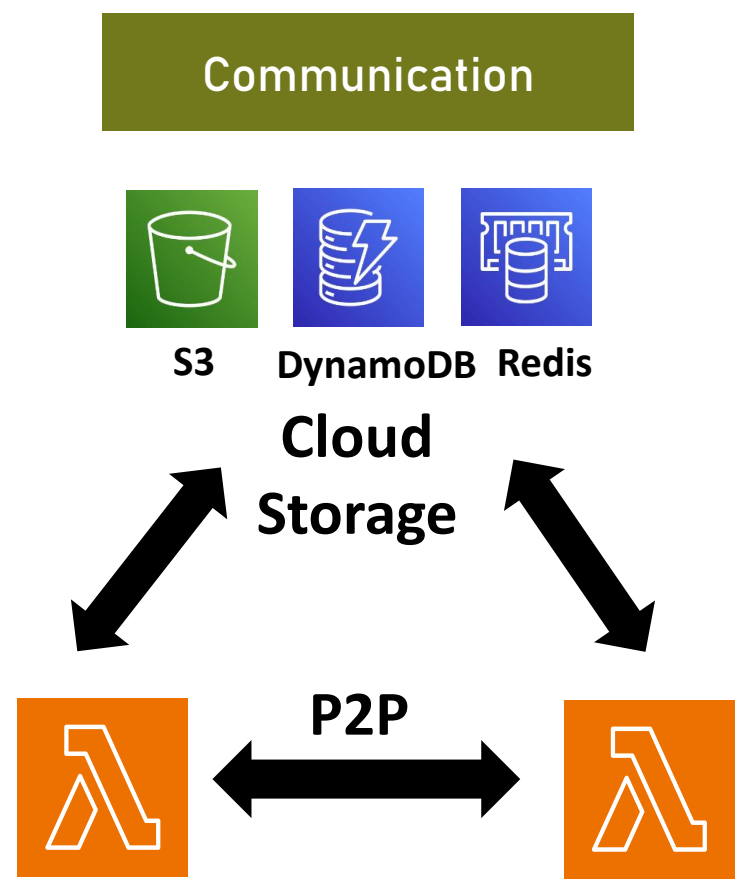
Serverless has been changing – and so did we!



Serverless has been changing – and so did we!



Serverless has been changing – and so did we!



Profiling Multi-function Applications

```

import faas_profiler_python as fp

@fp.profile()
def serverless_handler(*args, **kwargs):
    pass
  
```


SeBS in Practice

 [spcl/serverless-benchmarks](https://github.com/spcl/serverless-benchmarks)

SeBS in Practice

 [spcl/serverless-benchmarks](https://github.com/spcl/serverless-benchmarks)

What do you need?



- ❖ Docker running on your system 
- ❖ Python 3.7+
- ❖ libcurl
- ❖ Virtualenv
- ❖ Works on Linux. WSL should also work 😊



SeBS in Practice

spcl/serverless-benchmarks

What do you need?

- ❖ Docker running on your system 
- ❖ Python 3.7+
- ❖ libcurl
- ❖ Virtualenv
- ❖ Works on Linux. WSL should also work 😊 

Where to execute functions?

- ❖ Cloud? You need credentials and set up account.
- ❖ Open source? Deploy OpenWhisk – instructions are provided.
- ❖ Local test environment? 

What do you need?

AWS Lambda

AWS provides one year of free services, including a significant amount of computing time in AWS Lambda. To work with AWS, you need to provide access and secret keys to a role with permissions sufficient to manage functions and S3 resources. Additionally, the account must have `AmazonAPIGatewayAdministrator` permission to set up automatically AWS HTTP trigger. You can provide a [role](#) with permissions to access AWS Lambda and S3; otherwise, one will be created automatically. To use a user-defined lambda role, set the name in config JSON - see an example in `config/example.json`.

You can pass the credentials either using the default AWS-specific environment variables:

```
export AWS_ACCESS_KEY_ID=XXXX
export AWS_SECRET_ACCESS_KEY=XXXX
```

SeBS in Practice

 [spcl/serverless-benchmarks](https://github.com/spcl/serverless-benchmarks)

DEMO #1: Deploy function on AWS!

DEMO #1: Deploy function on AWS!

```
git clone --recursive git@github.com:spcl/serverless-benchmarks.git
```

```
./install.py
```

```
source python-venv/bin/activate
```


DEMO #1: Deploy function on AWS!

```
git clone --recursive git@github.com:spcl/serverless-benchmarks.git
```

```
./install.py
```

```
source python-venv/bin/activate
```

```
./sebs.py benchmark invoke 110.dynamic-html test --repetitions 5 --deployment aws --language python --language-version 3.8 --config config/example.json
```

```
./sebs.py benchmark process --config config/example.json
```

SeBS Modularity

 [spcl/serverless-benchmarks](https://github.com/spcl/serverless-benchmarks)

SeBS Modularity

 [spcl/serverless-benchmarks](https://github.com/spcl/serverless-benchmarks)

Add new platform?

- ❖ Define packaging function.
- ❖ Add API calls to create/update/delete function.
- ❖ Add function triggers (e.g. retrieve URL)

SeBS Modularity

 [spcl/serverless-benchmarks](https://github.com/spcl/serverless-benchmarks)

Add new platform?

- ❖ Define packaging function.
- ❖ Add API calls to create/update/delete function.
- ❖ Add function triggers (e.g. retrieve URL)

Add new experiment?

- ❖ Functions and resources automatically deployed.
- ❖ Just add your own logic.

SeBS Modularity

 [spcl/serverless-benchmarks](https://github.com/spcl/serverless-benchmarks)

Add new platform?

- ❖ Define packaging function.
- ❖ Add API calls to create/update/delete function.
- ❖ Add function triggers (e.g. retrieve URL)

Add new experiment?

- ❖ Functions and resources automatically deployed.
- ❖ Just add your own logic.

Add new function?

- ❖ Define dependencies.
- ❖ Define input generation and data.
- ❖ Add code!

DEMO #2: Functions need data!

```
./sebs.py storage start minio --port 9011 --output-json out_storage.json
```

```
./sebs.py local start 210.thumbnailer small local_benchmarks.json --language python --language-version 3.8 --  
config config/example.json --storage-configuration out_storage.json
```

```
curl $(jq -r '.functions[0].url' local_benchmarks.json) --request POST  
T --data "$(jq '.inputs[0]' local_benchmarks.json)" --header 'Content-Type: application/json'
```

SeBS in Practice

 [spcl/serverless-benchmarks](https://github.com/spcl/serverless-benchmarks)

DEMO #3: Run experiment on AWS.

```
./sebs.py experiment invoke perf-cost --config config/experiment.json --deployment aws --language python --  
language-version 3.8
```

Serverless is changing – and so are we!

Serverless is changing – and so are we!

New Benchmarks

New Serverless Applications

Storage & Queue Triggers

Large Applications

Elasticity & Scaling Experiments

Serverless is changing – and so are we!

New Benchmarks

New Serverless Applications

Storage & Queue Triggers

Large Applications

Elasticity & Scaling Experiments

New Platforms



fission



Serverless is changing – and so are we!

New Benchmarks

New Serverless Applications

Storage & Queue Triggers

Large Applications

Elasticity & Scaling Experiments

Heterogeneous Serverless

AI/ML is Difficult Without GPUs

Trade-offs of GPU Sharing

New Platforms



fission



Serverless is changing – and so are we!

New Benchmarks

New Serverless Applications

Storage & Queue Triggers

Large Applications

Elasticity & Scaling Experiments

Heterogeneous Serverless

AI/ML is Difficult Without GPUs

Trade-offs of GPU Sharing

New Platforms






Long-Term Stability

How Does Serverless Performance Change Over Time?

What Causes High Tail Latency and Outliers?

What Comes Next for Serverless?

More of SPCL's research:

-  youtube.com/@spcl **180+ Talks**
-  twitter.com/spcl_eth **1.4K+ Followers**
-  github.com/spcl **3.8K+ Stars**

... or spcl.ethz.ch



**SeBS
Paper**



**SeBS
Repo**




What Comes Next for Serverless?

What will be the runtime of the future?

More of SPCL's research:

 youtube.com/@spcl **180+ Talks**

 twitter.com/spcl_eth **1.4K+ Followers**

 github.com/spcl **3.8K+ Stars**

... or spcl.ethz.ch



**SeBS
Paper**



**SeBS
Repo**



What Comes Next for Serverless?


What will be the runtime of the future?

Where are limits of scalability and resource allocation?

More of SPCL's research:

 youtube.com/@spcl **180+ Talks**

 twitter.com/spcl_eth **1.4K+ Followers**

 github.com/spcl **3.8K+ Stars**

... or spcl.ethz.ch



**SeBS
Paper**



**SeBS
Repo**



What Comes Next for Serverless?

What will be the runtime of the future?

Where are limits of scalability and resource allocation?

Are we going to break free from the vendor lock-in?

More of SPCL's research:

 youtube.com/@spcl **180+ Talks**

 twitter.com/spcl_eth **1.4K+ Followers**

 github.com/spcl **3.8K+ Stars**

... or spcl.ethz.ch



**SeBS
Paper**



**SeBS
Repo**



What Comes Next for Serverless?

What will be the runtime of the future?

Where are limits of scalability and resource allocation?

Are we going to break free from the vendor lock-in?

What will be the next serverless programming model?

More of SPCL's research:

 youtube.com/@spcl **180+ Talks**

 twitter.com/spcl_eth **1.4K+ Followers**

 github.com/spcl **3.8K+ Stars**

... or spcl.ethz.ch



**SeBS
Paper**



**SeBS
Repo**



What Comes Next for Serverless?

What will be the runtime of the future?

Where are limits of scalability and resource allocation?


Are we going to break free from the vendor lock-in?


What will be the next serverless programming model?

Serverless Needs Open Standard for Benchmarking!

More of SPCL's research:

 youtube.com/@spcl **180+ Talks**

 twitter.com/spcl_eth **1.4K+ Followers**

 github.com/spcl **3.8K+ Stars**

... or spcl.ethz.ch



SeBS Paper



SeBS Repo

