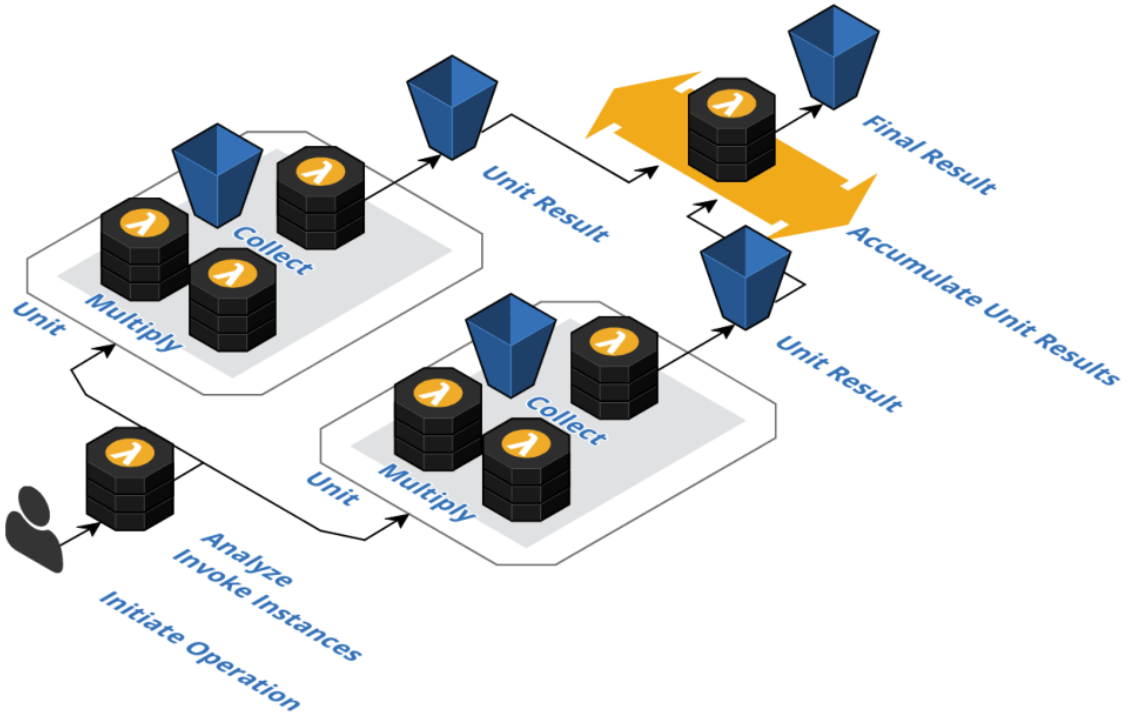
Benchmarking FaaS Platforms: Call for Community Participation

Jörn Kuhlenkamp | ISE | WoSC 2018 | 20.12.2018

FaaS Example: Matrix Multiplication



How can app developers obtain evidence for quality-driven design decisions?

➔ Benchmarking

Motivation

Select challenges of “good” benchmarking:

Relevance:	multiple workloads, qualities, and platform features of interest
Reproducibility:	completeness of documented testbed, execution, and results
Fairness:	equal support of different SUTs
Usability:	tooling, cost of execution

➔ Highly desirable to build on existing body of work for high quality evidence

How can experimenters accurately and efficiently identify the SOTA for FaaS platform benchmarking?

Contributions:

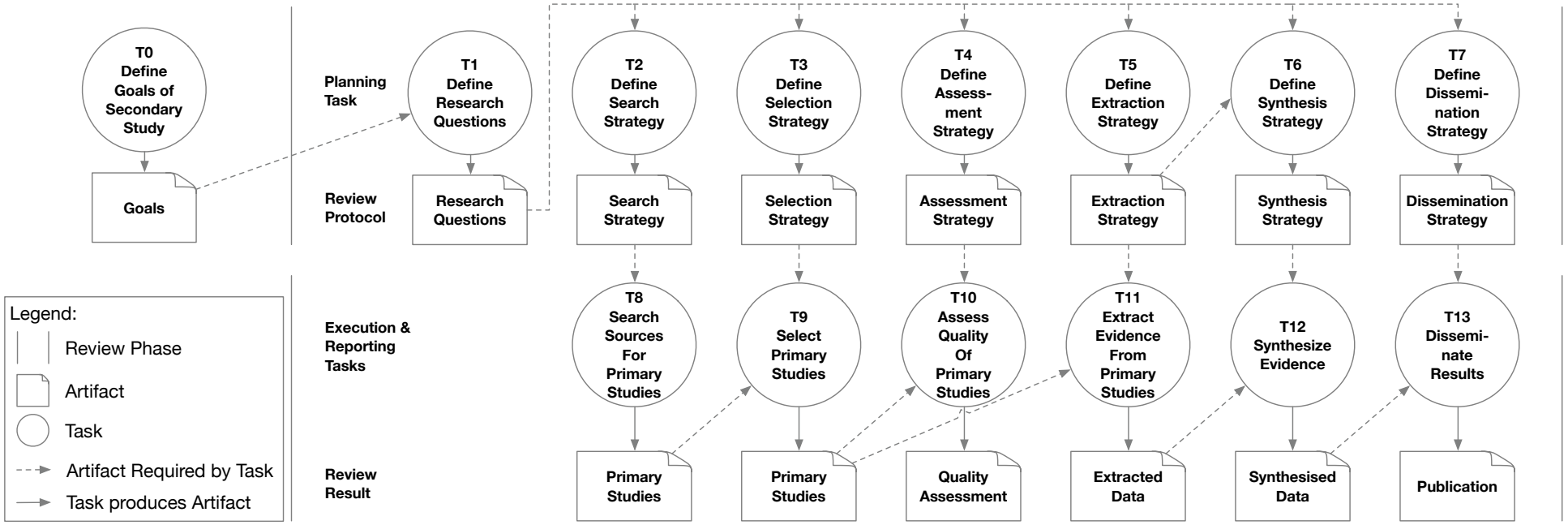
(1) Review protocol for a systematic literature review (SLR)

(2) Call for community participation

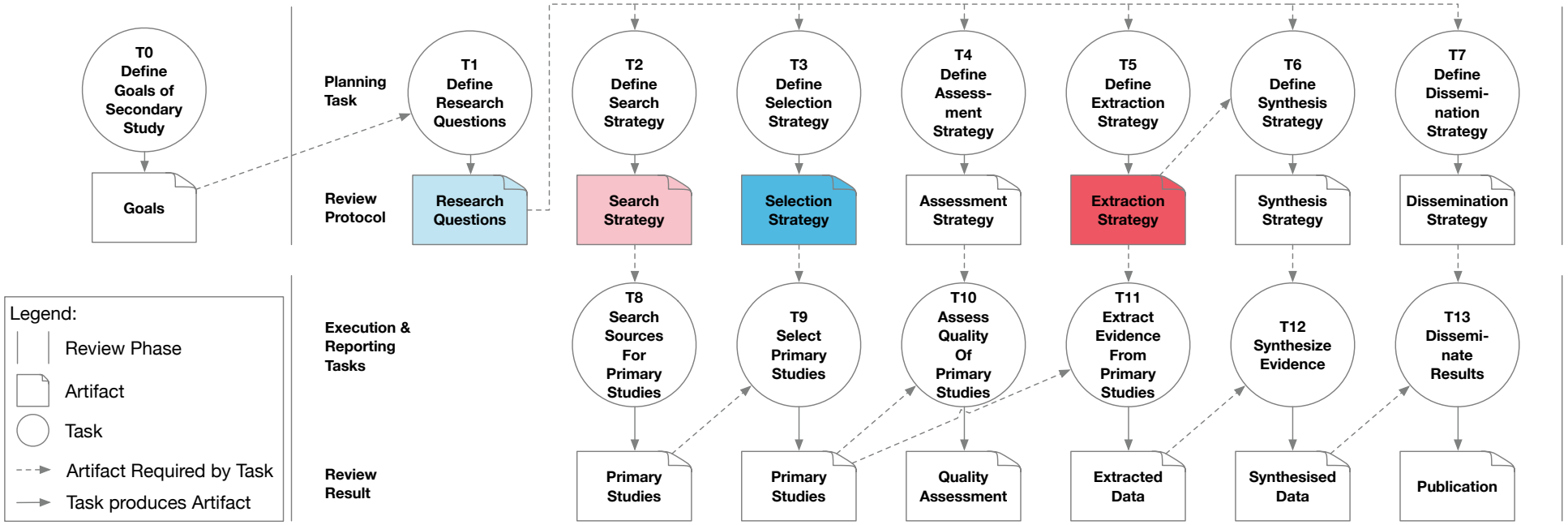
(3) Preliminary results

Review Protocol

Overview


T0
Define
Goals of
Secondary
Study
Goals
Planning
Task
T1
Define
Research
Questions
Review
Protocol
Research
Questions
T2
Define
Search
Strategy
T3
Define
Selection
Strategy
T4
Define
Assess-
ment
Strategy
T5
Define
Extraction
Strategy
T6
Define
Synthesis
Strategy
T7
Define
Dissemi-
nation
Strategy
Search
Strategy
Selection
Strategy
Assessment
Strategy
Extraction
Strategy
Synthesis
Strategy
Dissemination
Strategy
Legend:
Review Phase
Artifact
Task
Artifact Required by Task
Task produces Artifact
Execution &
Reporting
Tasks
T8
Search
Sources
For
Primary
Studies
T9
Select
Primary
Studies
T10
Assess
Quality
Of
Primary
Studies
T11
Extract
Evidence
From
Primary
Studies
T12
Synthesize
Evidence
T13
Dissemi-
nate
Results
Review
Result
Primary
Studies
Primary
Studies
Quality
Assessment
Extracted
Data
Synthesised
Data
Publication

Overview


T0 Define Goals of Secondary Study
Goals
Planning Task
T1 Define Research Questions
Review Protocol
Research Questions
T2 Define Search Strategy
Search Strategy
T3 Define Selection Strategy
Selection Strategy
T4 Define Assess-ment Strategy
Assessment Strategy
T5 Define Extraction Strategy
Extraction Strategy
T6 Define Synthesis Strategy
Synthesis Strategy
T7 Define Dissemi-nation Strategy
Dissemination Strategy
Legend:
Review Phase
Artifact
Task
Artifact Required by Task
Task produces Artifact
Execution & Reporting Tasks
T8 Search Sources For Primary Studies
T9 Select Primary Studies
T10 Assess Quality Of Primary Studies
T11 Extract Evidence From Primary Studies
T12 Synthesize Evidence
T13 Dissemi-nate Results
Review Result
Primary Studies
Primary Studies
Quality Assessment
Extracted Data
Synthesised Data
Publication

Select Details

Select Details

Research Questions	Search	Selection	Extraction
FaaS experiments in literature: SUTs? Treatments? Qualities? Designs? Reproducible?			

Select Details

Research Questions	Search	Selection	Extraction
FaaS experiments in literature: SUTs? Treatments? Qualities? Designs? Reproducible?	Seed: 5 publications Search: Snowballing		

Select Details

Research Questions	Search	Selection	Extraction
FaaS experiments in literature: SUTs? Treatments? Qualities? Designs? Reproducible?	Seed: 5 publications Search: Snowballing	Scientific? After Jan 1st 2015? FaaS platform is SUT? Experiment? Design? Results?	

Select Details

Research Questions	Search	Selection	Extraction
FaaS experiments in literature: SUTs? Treatments? Qualities? Designs? Reproducible?	Seed: 5 publications Search: Snowballing	Scientific? After Jan 1st 2015? FaaS platform is SUT? Experiment? Design? Results?	Quality/Features SUT Load Generator Measurements Treatments Analysis

Observed Limitations

Select limitations of review protocol:

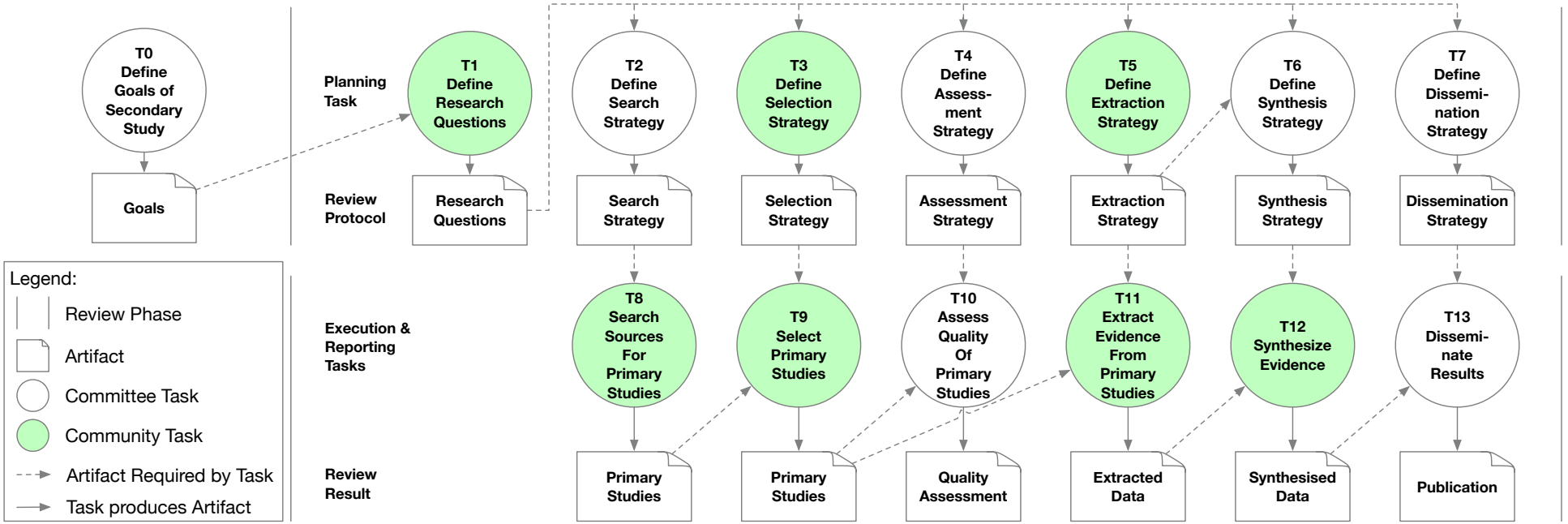
- Outdated publications: due to long publication and short development cycles?
- Incomplete experiment descriptions: due to space constraints?
- Researchers are limited resources and tasks are partially hard to automate?
- “Reinventing the wheel/experiment”?

Approach:

- Call for community participation
- Community-driven knowledge base

Call for Participation

Overview


T0
Define
Goals of
Secondary
Study
Goals
Planning
Task
T1
Define
Research
Questions
Review
Protocol
Research
Questions
T2
Define
Search
Strategy
T3
Define
Selection
Strategy
T4
Define
Assess-
ment
Strategy
T5
Define
Extraction
Strategy
T6
Define
Synthesis
Strategy
T7
Define
Dissemi-
nation
Strategy
Search
Strategy
Selection
Strategy
Assessment
Strategy
Extraction
Strategy
Synthesis
Strategy
Dissemination
Strategy
Legend:
Review Phase
Artifact
Committee Task
Community Task
Artifact Required by Task
Task produces Artifact
Execution &
Reporting
Tasks
T8
Search
Sources
For
Primary
Studies
T9
Select
Primary
Studies
T10
Assess
Quality
Of
Primary
Studies
T11
Extract
Evidence
From
Primary
Studies
T12
Synthesize
Evidence
T13
Dissemi-
nate
Results
Review
Result
Primary
Studies
Primary
Studies
Quality
Assessment
Extracted
Data
Synthesised
Data
Publication

Participation

https://www.tu-berlin.de/?id=199198

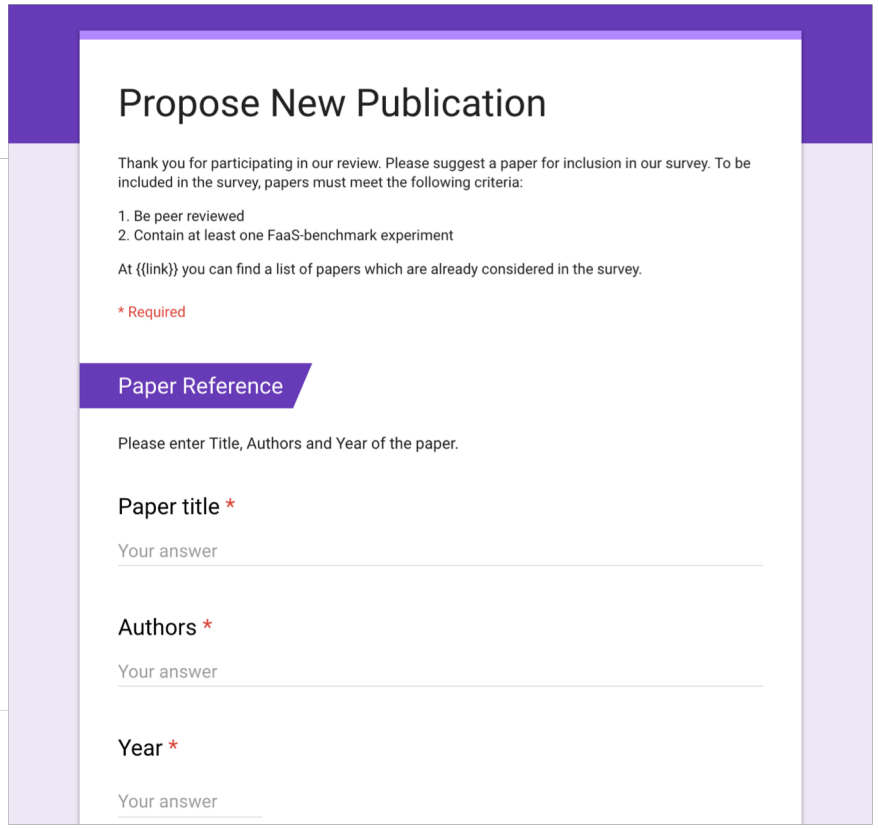
Forms for participation in the community tasks are listed below:

- (T1) Propose Research Question
- (T3) Propose Criteria
- (T5) Propose New Column
- (T8) Propose New Publication
- (T9) Check criteria for publication
- (T11) Propose Experiment Data
- (T12) Add Data Analysis

Archived Versions

Archive		
Reference	**Date**	**Link**
001	09/2018	Link

Snapshot 001



(Personal information, e.g., name, comments, and email, will not be published)

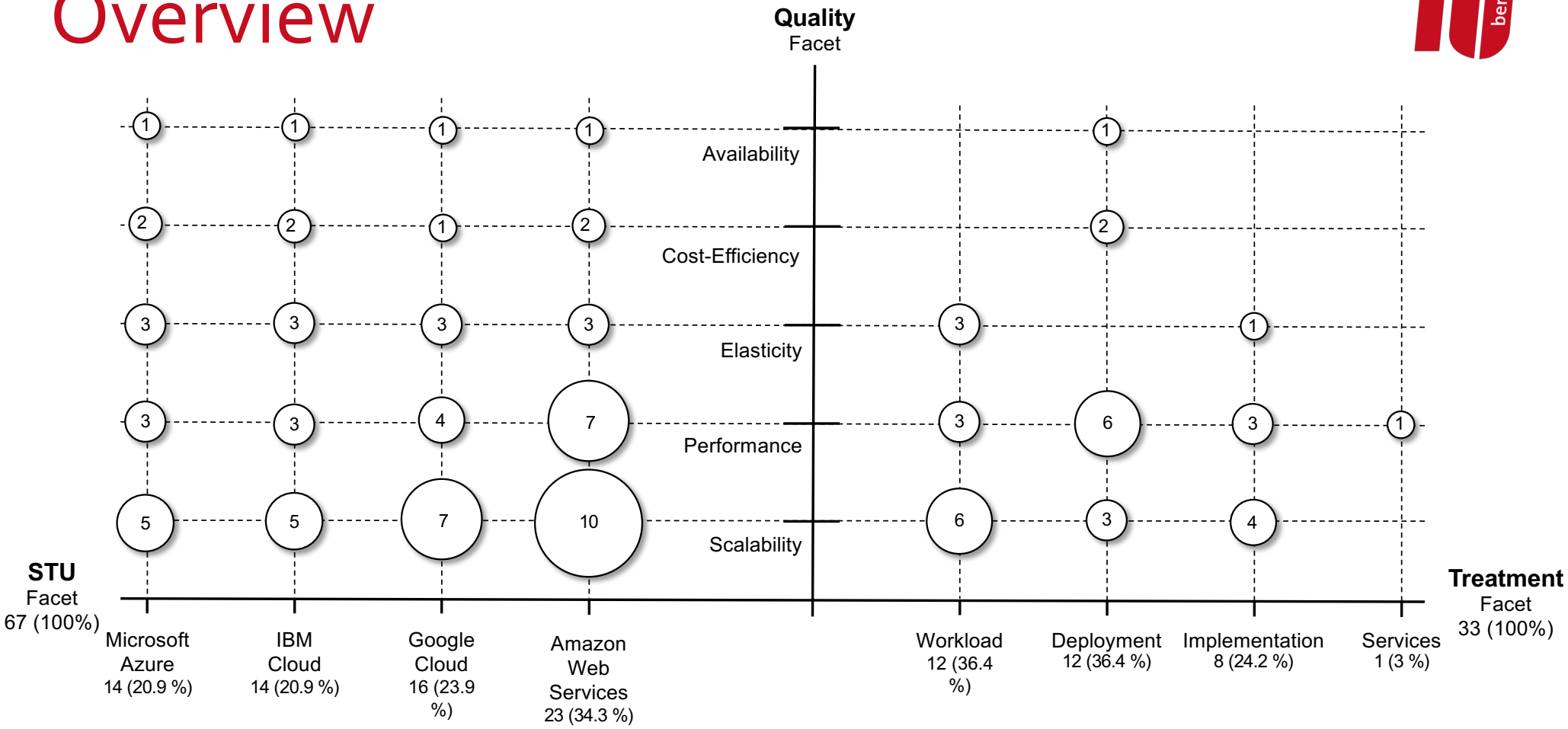
Ex: Select Evidence for Snapshot 001

References

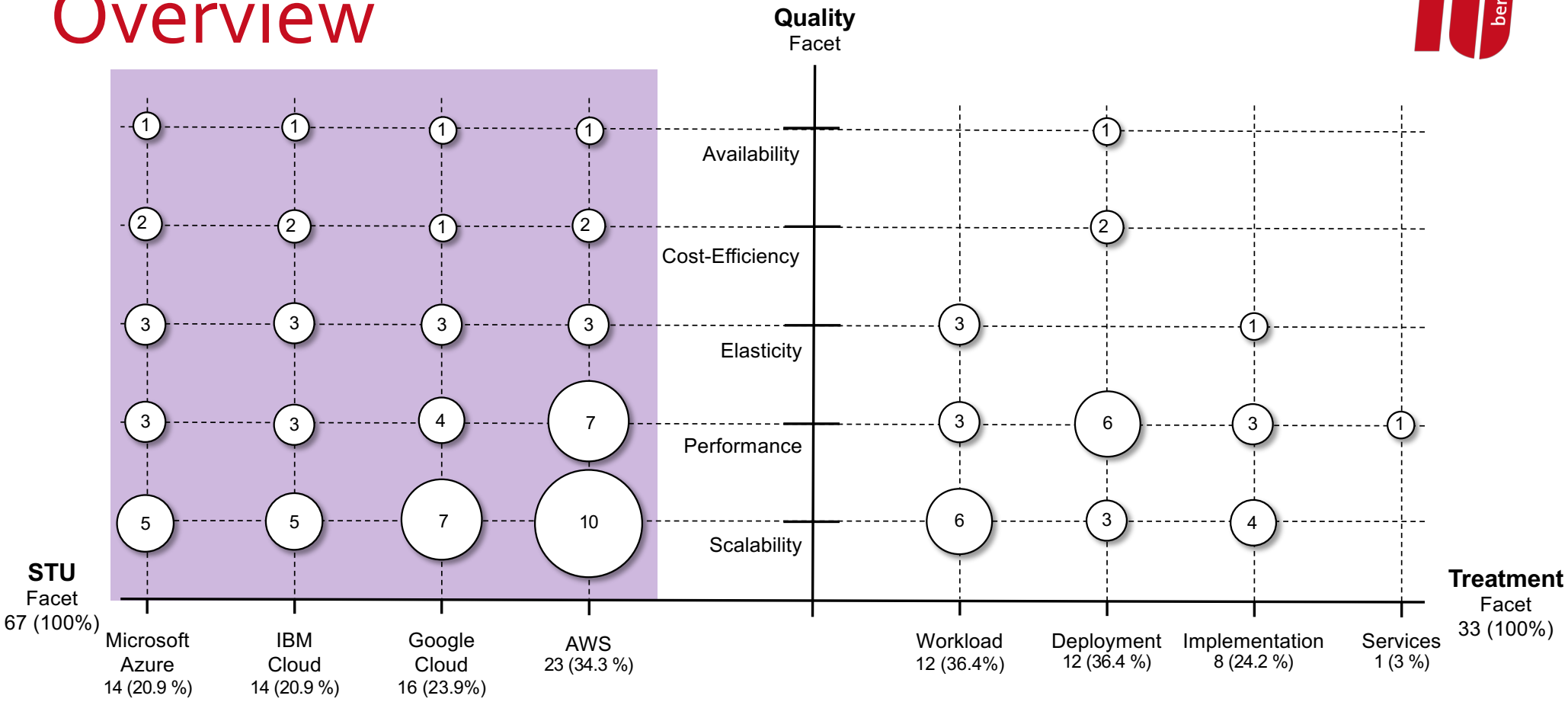
	References				Exp. Goal			Parameters					
								Parameter 1 (P1)			Paramter 2 (P2)		
	Exp. Ref. [E#]	Exp. Ref Paper	Exp. Name [Text]	Lit. Ref. [#]	Abstraction	Quality	Feature	Name	Unit	Domain	Name	Unit	Domain
4	E1a	E1a	Concurrency Test	1	Feature	Scalability	Parallel Container Sheduling	Concurrent Pending Req.	#	[1,15]	FaaS Service	Text	{GCF, AWS, AF, IBM}
5	E1b	E1b	Backoff Test	1	Feature	Elasticity	Depovisioning Time	Time Since Last Execution	m	[1,30]	FaaS Service	Text	{GCF, AWS, AF, IBM}
6	E2a	E2a	Max Requests	2	Feature	Scalability	-	Concurrent Pending Req.	#k	{0.5, 1, 2, 3, 10}	FaaS Service	Text	{GCF, AWS, AF, IBM}
7	E2b	E2b	Max Cpu Perf.	2	Feature	Scalability	-	Concurrent Pending Req.	#	{1, 100, 3000}	FaaS Service	Text	{GCF, AWS, AF, IBM}
8	E2c	E2c	Max Disk Perf.	2	Feature	Scalability	-	Concurrent Pending Req.	#	{1, 100}	FaaS Service	Text	{GCF, AWS, AF, IBM}
9	E2d	E2d	Max Net Perf.	2	Feature	Scalability	-	Concurrent Pending Req.	#	{1, 100}	FaaS Service	Text	{GCF, AWS, AF, IBM}
10	E2e	E2e	Dynamic Workload	2	Feature	Elasticity	-	Concurrent Pending Req.	#	[10,90]	FaaS Service	Text	{GCF, AWS, AF, IBM}
11	E2f	E2f	Update Function	2	Feature	Maintainability	-	Code Version	#	[1,2]	Func. Config.	#	[1,2]
12	E2g	E2g	FaaS vs. VMs	2	N/A	Cost/Performance		Compute Service	Text	{FaaS, VM}	FaaS Service	Text	{GCF, AWS, AF, IBM}
13	E2h	E2h	Trigger	2	Feature	Performance	-	Trigger	Text	{HTTP, Object, Database}	FaaS Service	Text	{GCF, AWS, AF, IBM}
14	E2i	E2i	Programming Platform	2	Feature	Performance	-	Prog. Platform	Text	{Node.js, Java, C#, Pyt 2, Pyt. 3}	FaaS Service	Text	{GCF, AWS, AF, IBM}
15	E3a	E3a	Compute - Fibonacci	3	Feature	Cost/Performance	CPU*	Func Mem	MB	{128,256,512,1024}	FaaS Service	Text	{AWS, AF, IBM}
16	E3b	E3b	Compute - Pi	3	Application	Performance	Math	Threads	#	[1,20]	FaaS Service	Text	N/A
17	E3c	E3c	I/O - Face Detection	3	Application	Performance	Computer Graphics	Threads	#	[1,20]	FaaS Service	Text	N/A
18	E3d	E3d	Password Cracking	3	Application	Performance	Crypto	Mappers	#	[1,9]	FaaS Service	Text	{AWS, native}
19	E3e	E3e	Precipitation Forecast	3	Application	Scalability	Merelogy	Lines in WL	#	[1,30]	-	-	-
20	E4a	E4a	Load Burst Test	4	Feature	Elasticity	-	Service	Text	{AWS, BeanStock}	-	-	-
21	E4b	E4b	Start Up Time	4	Feature	Scalability	Startup Latancy	Service	Text	{AWS, BeanStock}			
22	E5a	E5a	Linpack - Exe. Delay	5	Feature	Elasticity	Exec Delay	FaaS Service	Text	{AWS, GCF, IBM}	-	-	-
23	E5b	E5b	Linpack - Flops	5	Application	Scalability	Math	FaaS Service	Text	{AWS, GCF, IBM}	Func Mem	MB	{256,512,1024,1536,204
24	E7a	E6a	Supercomputer Test	7		Scalability	-	Worker	#	[3600]			
25	E10a	E7a	CPU Benchmark	10	Feature	Performance	Infrastructure	FaaS Service	Text	{GCF, AWS, AF, AOW}	Function Memory	MB	{128,256,512,1024}
26	E10b	E7b	Overhead	10		Performance	Startup Latancy	FaaS Service	Text	{GCF, AWS, AF, AOW}	-	-	-
27	E10c	E7c	Supercomputer Test	10		Scalability	-	Fork/Complexity	#	{10,20,...,100}	Function Memory	MB	{128,256,512,1024}
28	E11a	E8a	Matrix Mutliplication	11	Application	Scalability	Math	Worker	#	{500,1000,..,3000}			
29	E15a	E9a	Image Processing	15	Application	Scalability	Graphics	Request	#	{0,...,6000}	-	-	-
30	E30a		Fourier Transformation	30	Feature	Performance	Math	load	#	[13,...,21]	Function Memory	MB	{128,256,512,1024}
31	E30b		Matrix Mutliplication	30	Feature	Performance	Math	size	#	[1,...,10]	Function Memory	MB	{1024,2048}
32	E30c		Sleep	30	Feature	Performance	Exec Perf.	duration	#	[1,...,13]	Function Memory	MB	{128,256,512,1024,204

Prelimanry Results

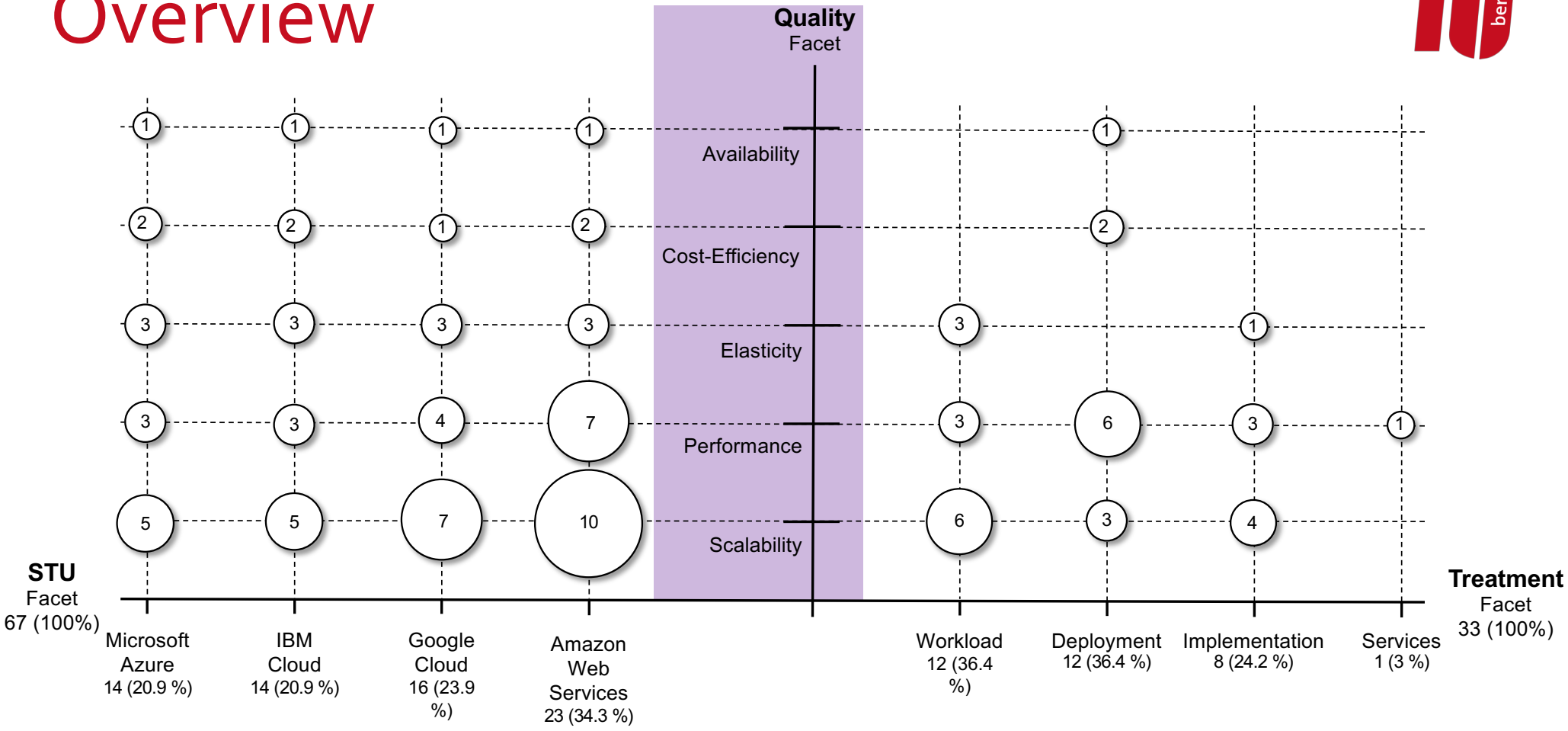
Overview


Quality
Facet
Availability
Cost-Efficiency
Elasticity
Performance
Scalability
STU
Facet
67 (100%)
Microsoft Azure
14 (20.9 %)
IBM Cloud
14 (20.9 %)
Google Cloud
16 (23.9 %)
Amazon Web Services
23 (34.3 %)
Treatment
Facet
33 (100%)
Workload
12 (36.4 %)
Deployment
12 (36.4 %)
Implementation
8 (24.2 %)
Services
1 (3 %)

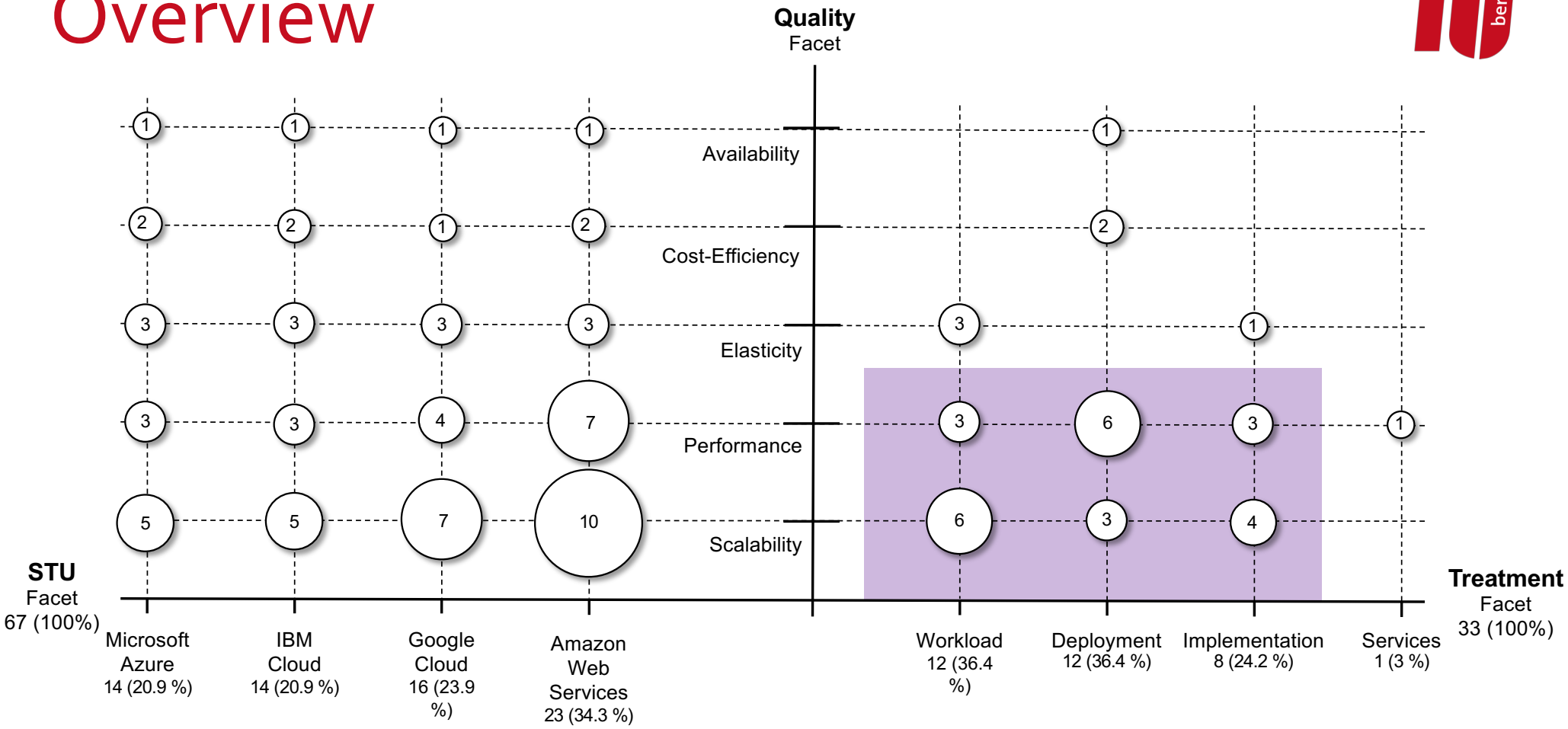
Overview


Quality
Facet
Availability
Cost-Efficiency
Elasticity
Performance
Scalability
STU
Facet
67 (100%)
Microsoft
Azure
14 (20.9 %)
IBM
Cloud
14 (20.9 %)
Google
Cloud
16 (23.9%)
AWS
23 (34.3 %)
Workload
12 (36.4%)
Deployment
12 (36.4 %)
Implementation
8 (24.2 %)
Services
1 (3 %)
Treatment
Facet
33 (100%)

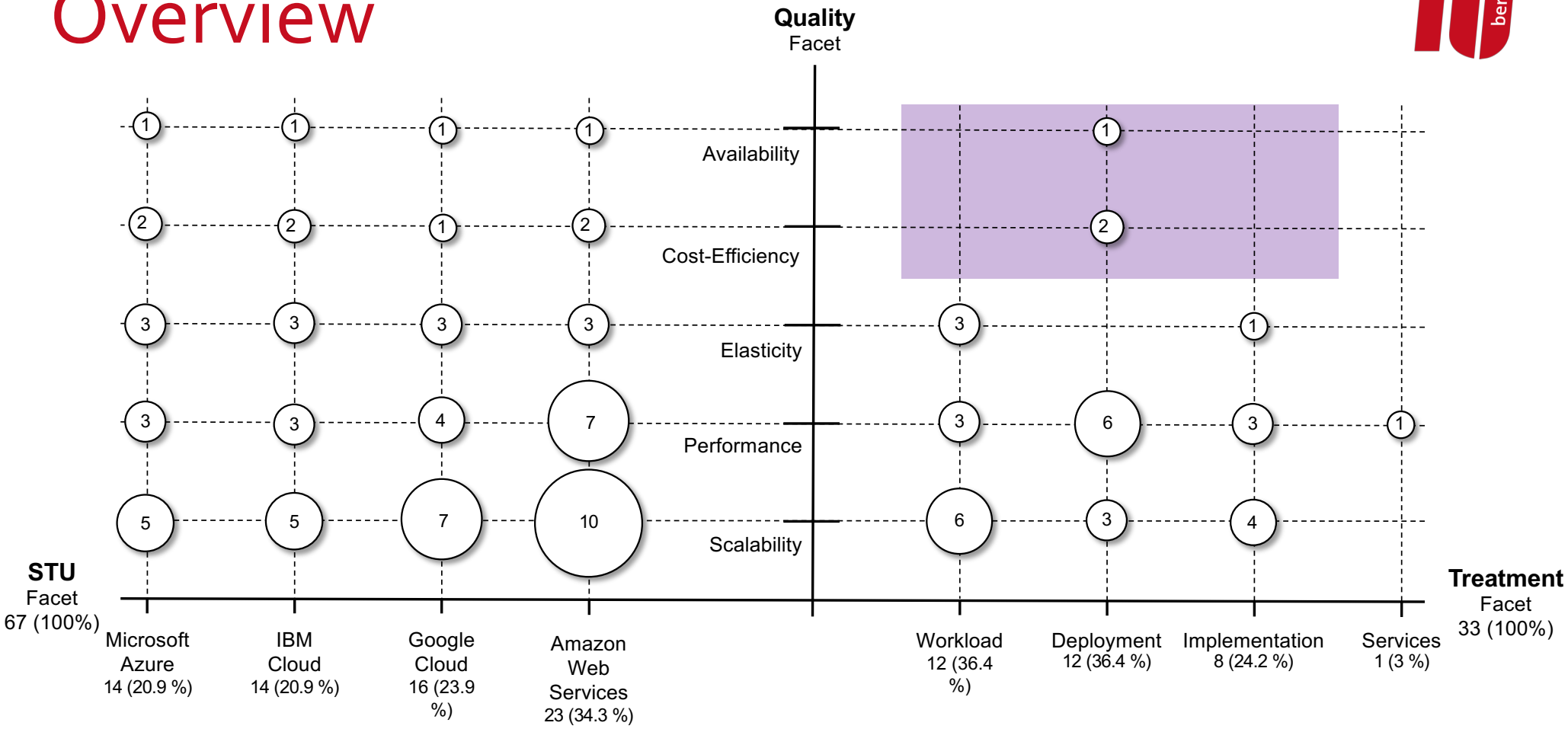
Overview


Quality
Facet
Availability
Cost-Efficiency
Elasticity
Performance
Scalability
STU
Facet
67 (100%)
Microsoft Azure
14 (20.9 %)
IBM Cloud
14 (20.9 %)
Google Cloud
16 (23.9 %)
Amazon Web Services
23 (34.3 %)
Treatment
Facet
33 (100%)
Workload
12 (36.4 %)
Deployment
12 (36.4 %)
Implementation
8 (24.2 %)
Services
1 (3 %)

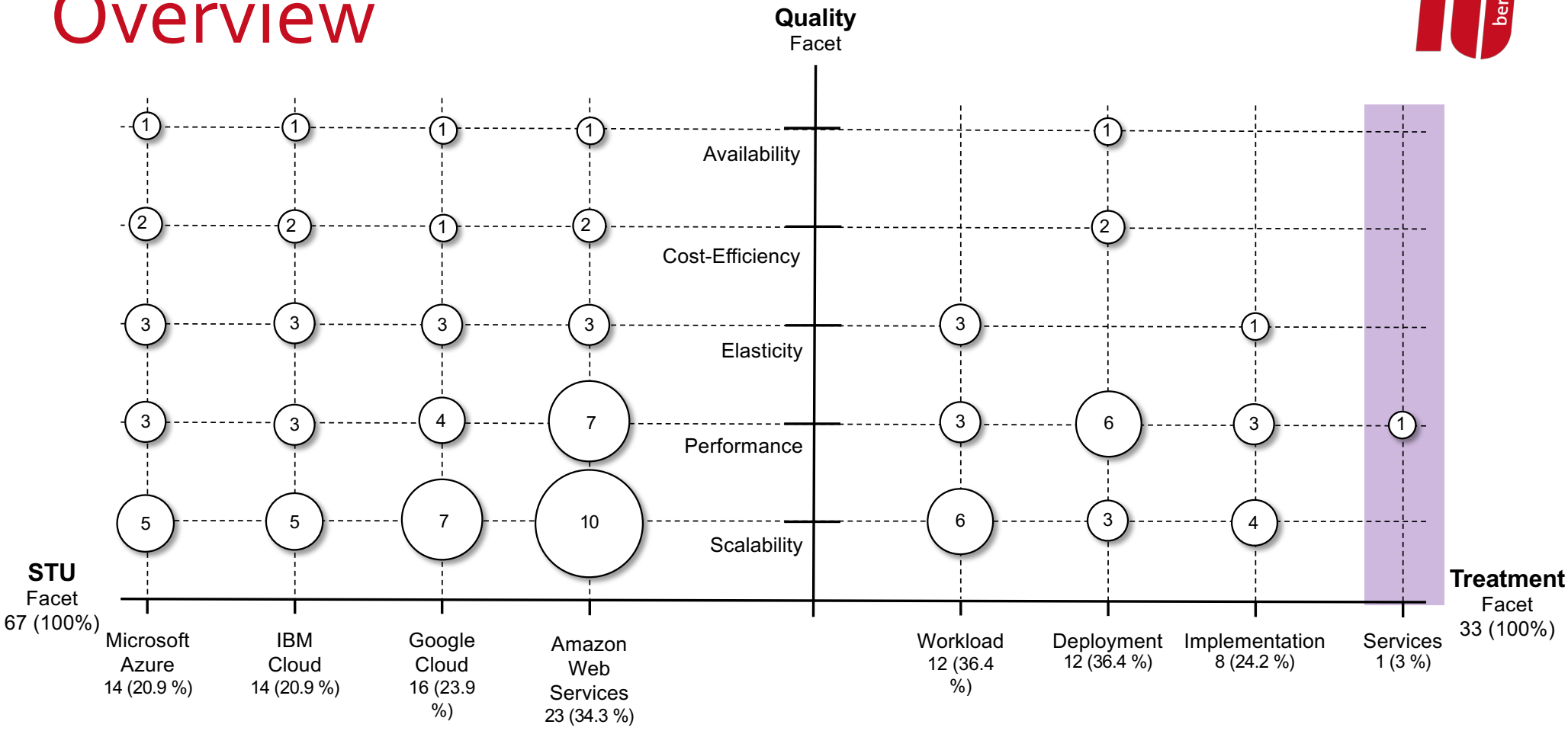
Overview


Quality
Facet
Availability
Cost-Efficiency
Elasticity
Performance
Scalability
STU
Facet
67 (100%)
Microsoft Azure
14 (20.9 %)
IBM Cloud
14 (20.9 %)
Google Cloud
16 (23.9 %)
Amazon Web Services
23 (34.3 %)
Treatment
Facet
33 (100%)
Workload
12 (36.4 %)
Deployment
12 (36.4 %)
Implementation
8 (24.2 %)
Services
1 (3 %)

Overview


Quality
Facet
Availability
Cost-Efficiency
Elasticity
Performance
Scalability
STU
Facet
67 (100%)
Microsoft Azure
14 (20.9 %)
IBM Cloud
14 (20.9 %)
Google Cloud
16 (23.9 %)
Amazon Web Services
23 (34.3 %)
Treatment
Facet
33 (100%)
Workload
12 (36.4 %)
Deployment
12 (36.4 %)
Implementation
8 (24.2 %)
Services
1 (3 %)

Overview


Quality
Facet
Availability
Cost-Efficiency
Elasticity
Performance
Scalability
STU
Facet
67 (100%)
Microsoft Azure
14 (20.9 %)
IBM Cloud
14 (20.9 %)
Google Cloud
16 (23.9 %)
Amazon Web Services
23 (34.3 %)
Workload
12 (36.4 %)
Deployment
12 (36.4 %)
Implementation
8 (24.2 %)
Services
1 (3 %)
Treatment
Facet
33 (100%)

Designs

- No (de facto) standards
  - Wide variation of approaches and designs
  - Common tool for setup: Serverless-framework
- Deployment package
  - Trivial functions, such as sleep or No-Op functions
  - Trivial algorithms provided in pseudo-code
  - Complex algorithms as …
    - native FaaS programming code
    - binary packages which are executed using a FaaS wrapper function.
- Workload generation (trigger events)
  - Direct generation by a workload generator
  - Indirect generation by an downstream service

Reproducability

Reference		Workload Generator		Function Implementation			Platform Configuration		Services used	R-Score
Pub.	Exp.	Tool	Distance	Functionality	Type	Sources	Programming Environment	Memory		
[15]	E1a	Perf Tool	Region	Empty	Trivial	yes	js	512	No	4.0
	E1b	Perf Tool	Region	Empty	Trivial	yes	js	512	No	4.0
[20]	E2a	N/A	N/A	N/A	N/A	no	*	512,1536	N/A	1.5
	E2b	N/A	N/A	Matrix Mult.	Native	no	N/A	512,1536	No	2.5
	E2c	N/A	N/A	N/A (IO)	Native	no	N/A	512,1536	No	2.5
	E2d	N/A	N/A	N/A (net)	Native	no	*	512,1536	Yes	3.0
	E2e	N/A	N/A	Fast	Native	no	js	512,1536	No	3.0
	E2f	N/A	N/A	N/A	Native	no	*	512,1536	No	2.5
	E2g	N/A	N/A	N/A	Native	no	Py, js	3000	N/A	2.5
	E2h	N/A	N/A	N/A	N/A	no	N/A	N/A	Yes	1.5
	E2i	N/A	N/A	Wait	Trivial	no	*	N/A	No	2.5
[11]	E3a	N/A	N/A	Fibonacci	Pseudo	yes	Py	128-1024	No	3.0
	E3b	N/A	N/A	PI calculation	Native	no	Py, Py3	N/A	No	2.5
	E3c	N/A	N/A	Face detection	Native	no	Py	N/A	Yes	2.5
	E3d	N/A	N/A	Pwd Cracking	Native	no	Py	512	N/A	2.0
	E3e	HyperFlow	N/A	Weather	Binary	no	Py	N/A	N/A	2.0
[16]	E4a	N/A	N/A	Idle 200ms	Trivial	no	N/A	N/A	Yes	2.0
	E4b	N/A	N/A	Idle 200ms	Trivial	no	N/A	N/A	Yes	2.0
[21]	E5a	Custom	N/A	Linpack	Binary	no	N/A	512	N/A	2.0
	E5b	Custom	N/A	Linpack	Binary	no	N/A	256-2048	N/A	2.0
[22]	E6a	mu	Multi-region	Linpack	Binary	no	mu	N/A	Yes	3.5
[7]	E7a	N/A	Remote	Random gen.	Binary	no	js	128-1024	Yes	3.0
	E7b	N/A	Remote	Linpack	Binary	no	N/A	N/A	Yes	2.5
	E7c	HyperFlow	Remote	Linpack	Binary	no	js	128-1024	No	4.0
[17]	E8a	PyWren	Region	Matrix Mult.	Native	no	Py	N/A	Yes	3.5
[23]	E9a	N/A	N/A	Image Crop	N/A	no	N/A	N/A	Yes	2.5

* = js, Java, C#, Py, Py3

Measurement approach, “raw” measurements, and aggregations?

Reproducability

Reference		Workload Generator		Function Implementation			Platform Configuration		Services used	R-Score
Pub.	Exp.	Tool	Distance	Functionality	Type	Sources	Programming Environment	Memory		
[15]	E1a	Perf Tool	Region	Empty	Trivial	yes	js	512	No	4.0
	E1b	Perf Tool	Region	Empty	Trivial	yes	js	512	No	4.0
[20]	E2a	N/A	N/A	N/A	N/A	no	*	512,1536	N/A	1.5
	E2b	N/A	N/A	Matrix Mult.	Native	no	N/A	512,1536	No	2.5
	E2c	N/A	N/A	N/A (IO)	Native	no	N/A	512,1536	No	2.5
	E2d	N/A	N/A	N/A (net)	Native	no	*	512,1536	Yes	3.0
	E2e	N/A	N/A	Fast	Native	no	js	512,1536	No	3.0
	E2f	N/A	N/A	N/A	Native	no	*	512,1536	No	2.5
	E2g	N/A	N/A	N/A	Native	no	Py, js	3000	N/A	2.5
	E2h	N/A	N/A	N/A	N/A	no	N/A	N/A	Yes	1.5
	E2i	N/A	N/A	Wait	Trivial	no	*	N/A	No	2.5
[11]	E3a	N/A	N/A	Fibonacci	Pseudo	yes	Py	128-1024	No	3.0
	E3b	N/A	N/A	PI calculation	Native	no	Py, Py3	N/A	No	2.5
	E3c	N/A	N/A	Face detection	Native	no	Py	N/A	Yes	2.5
	E3d	N/A	N/A	Pwd Cracking	Native	no	Py	512	N/A	2.0
	E3e	HyperFlow	N/A	Weather	Binary	no	Py	N/A	N/A	2.0
[16]	E4a	N/A	N/A	Idle 200ms	Trivial	no	N/A	N/A	Yes	2.0
	E4b	N/A	N/A	Idle 200ms	Trivial	no	N/A	N/A	Yes	2.0
[21]	E5a	Custom	N/A	Linpack	Binary	no	N/A	512	N/A	2.0
	E5b	Custom	N/A	Linpack	Binary	no	N/A	256-2048	N/A	2.0
[22]	E6a	mu	Multi-region	Linpack	Binary	no	mu	N/A	Yes	3.5
[7]	E7a	N/A	Remote	Random gen.	Binary	no	js	128-1024	Yes	3.0
	E7b	N/A	Remote	Linpack	Binary	no	N/A	N/A	Yes	2.5
	E7c	HyperFlow	Remote	Linpack	Binary	no	js	128-1024	No	4.0
[17]	E8a	PyWren	Region	Matrix Mult.	Native	no	Py	N/A	Yes	3.5
[23]	E9a	N/A	N/A	Image Crop	N/A	no	N/A	N/A	Yes	2.5

* = js, Java, C#, Py, Py3

Measurement approach, “raw” measurements, and aggregations?

Reproducability

Reference		Workload Generator		Function Implementation			Platform Configuration		Services used	R-Score
Pub.	Exp.	Tool	Distance	Functionality	Type	Sources	Programming Environment	Memory		
[15]	E1a	Perf Tool	Region	Empty	Trivial	yes	js	512	No	4.0
	E1b	Perf Tool	Region	Empty	Trivial	yes	js	512	No	4.0
[20]	E2a	N/A	N/A	N/A	N/A	no	*	512,1536	N/A	1.5
	E2b	N/A	N/A	Matrix Mult.	Native	no	N/A	512,1536	No	2.5
	E2c	N/A	N/A	N/A (IO)	Native	no	N/A	512,1536	No	2.5
	E2d	N/A	N/A	N/A (net)	Native	no	*	512,1536	Yes	3.0
	E2e	N/A	N/A	Fast	Native	no	js	512,1536	No	3.0
	E2f	N/A	N/A	N/A	Native	no	*	512,1536	No	2.5
	E2g	N/A	N/A	N/A	Native	no	Py, js	3000	N/A	2.5
	E2h	N/A	N/A	N/A	N/A	no	N/A	N/A	Yes	1.5
	E2i	N/A	N/A	Wait	Trivial	no	*	N/A	No	2.5
[11]	E3a	N/A	N/A	Fibonacci	Pseudo	yes	Py	128-1024	No	3.0
	E3b	N/A	N/A	PI calculation	Native	no	Py, Py3	N/A	No	2.5
	E3c	N/A	N/A	Face detection	Native	no	Py	N/A	Yes	2.5
	E3d	N/A	N/A	Pwd Cracking	Native	no	Py	512	N/A	2.0
	E3e	HyperFlow	N/A	Weather	Binary	no	Py	N/A	N/A	2.0
[16]	E4a	N/A	N/A	Idle 200ms	Trivial	no	N/A	N/A	Yes	2.0
	E4b	N/A	N/A	Idle 200ms	Trivial	no	N/A	N/A	Yes	2.0
[21]	E5a	Custom	N/A	Linpack	Binary	no	N/A	512	N/A	2.0
	E5b	Custom	N/A	Linpack	Binary	no	N/A	256-2048	N/A	2.0
[22]	E6a	mu	Multi-region	Linpack	Binary	no	mu	N/A	Yes	3.5
[7]	E7a	N/A	Remote	Random gen.	Binary	no	js	128-1024	Yes	3.0
	E7b	N/A	Remote	Linpack	Binary	no	N/A	N/A	Yes	2.5
	E7c	HyperFlow	Remote	Linpack	Binary	no	js	128-1024	No	4.0
[17]	E8a	PyWren	Region	Matrix Mult.	Native	no	Py	N/A	Yes	3.5
[23]	E9a	N/A	N/A	Image Crop	N/A	no	N/A	N/A	Yes	2.5

* = js, Java, C#, Py, Py3

Measurement approach, “raw” measurements, and aggregations?

Conclusion

- Considerable existing body of work
- Single function performance/scalability ⇔ cost-efficiency, service compositions ➔ relevance?
- Rare publishing of implementations/toolkits and "raw" measurements ➔ reproducibility/verifiability?

Please participate! **https://www.tu-berlin.de/?id=199198**

Future work

- Reproduction of experiments with full disclosure of tools and results
- Completion of SLR
- Development of a serverless app for continuous SLR support

Thank You!

Jörn Kuhlenkamp
jk@ise.tu-berlin.de

Sebastian Werner
sw@ise.tu-berlin.de

ISEngineering