Scalability Analysis of Cloud Computing Services

Gunnar Brataas[1], Nikolas Herbst[2], Simon Ivansek[3], Jure Polutnik[3]

1: SINTEF Digital, Trondheim, Norway

2: University of Wurzburg, Germany

3: XLAB, Ljubljana, Slovenia

Self-Organizing Self-Managing Clouds Workshop

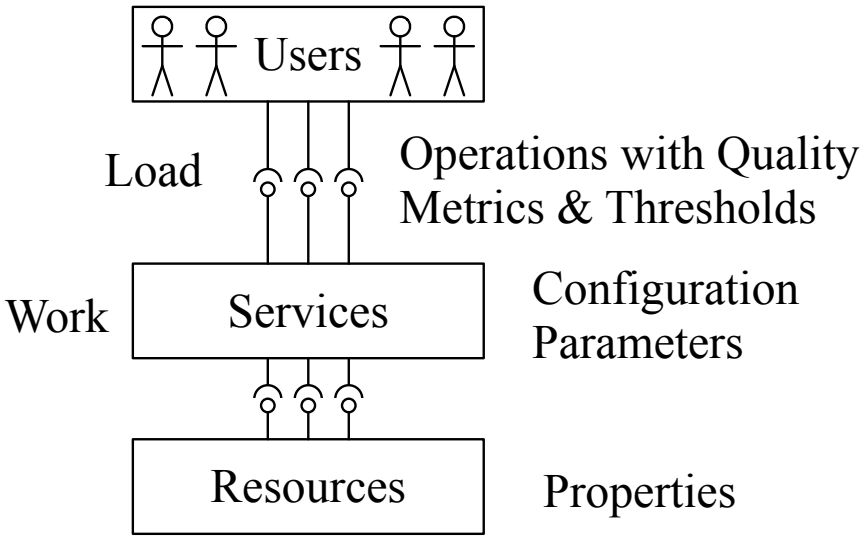
Part of: International Conference on Autonomic Computing

Columbus, Ohio, 17 July 2017

Research Questions and Corresponding Contributions

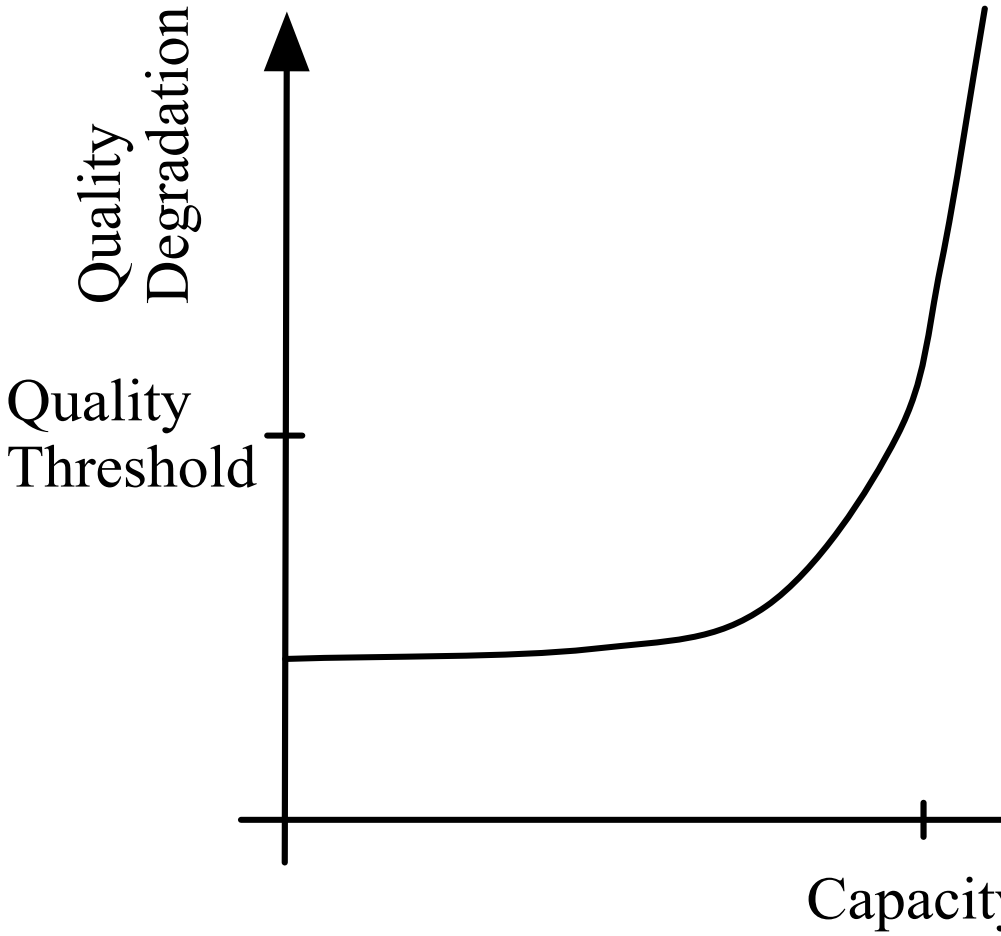
- RQ1: What are the essential influencing factors on cloud computing service scalability?
  - A set of equations with explicit work, quality thresholds and resource space
  - Deployment configurations with T-shirt sizes: small, medium, large
- RQ2: Which metrics are suitable to capture this insight?
  - Capacity is the base metric
  - Resource scalability metric
  - Cost scalability metric
- RQ3: How shall we structure the measurements?
  - Get overview with few measurements
  - Scale bottleneck VMs and exploit costly VMs
  - 53 measurements with 21 AWS configurations
  - 20 measurements with 2 OpenStack configurations
  - Result: Deployment controllers know the optimal configurations for a given workload

Core Concepts



- **Operation:** (function, call) way of interacting with a service
- **Operation mix:** probabilities of operations
- **Work:** amount of data to be processed, stored or communicated
- **Load:** how often an operation is invoked
  - **Open:** arrival rate or inter-arrival time
  - **Closed:** number of users and think time
- **Quality Metrics:** a measure of a quality, e.g. 90 percentile response times
- **Quality Thresholds:** the border been acceptable and non-acceptable quality, e.g. 3 seconds
- **Resources:** hardware (and software) platform
- **Configuration Parameters**: tune resources

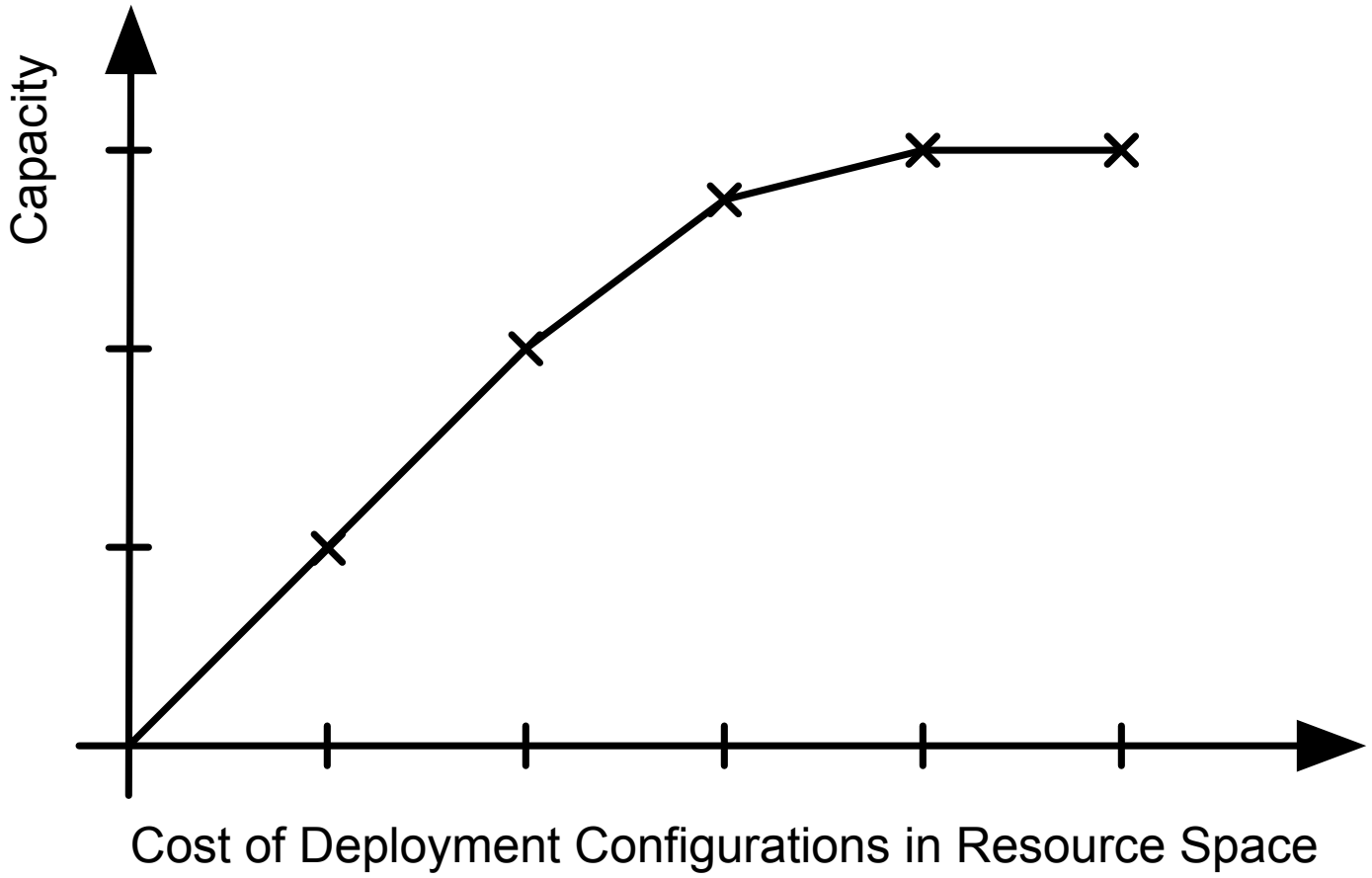
Capacity for Closed System



- Fixed:
  - Work parameters
  - Operation mix
  - Think time
  - Quality thresholds
  - Deployment configuration
- Variable:
  - # users in system
- Output:
  - Capacity: highest # users satisfying quality thresholds

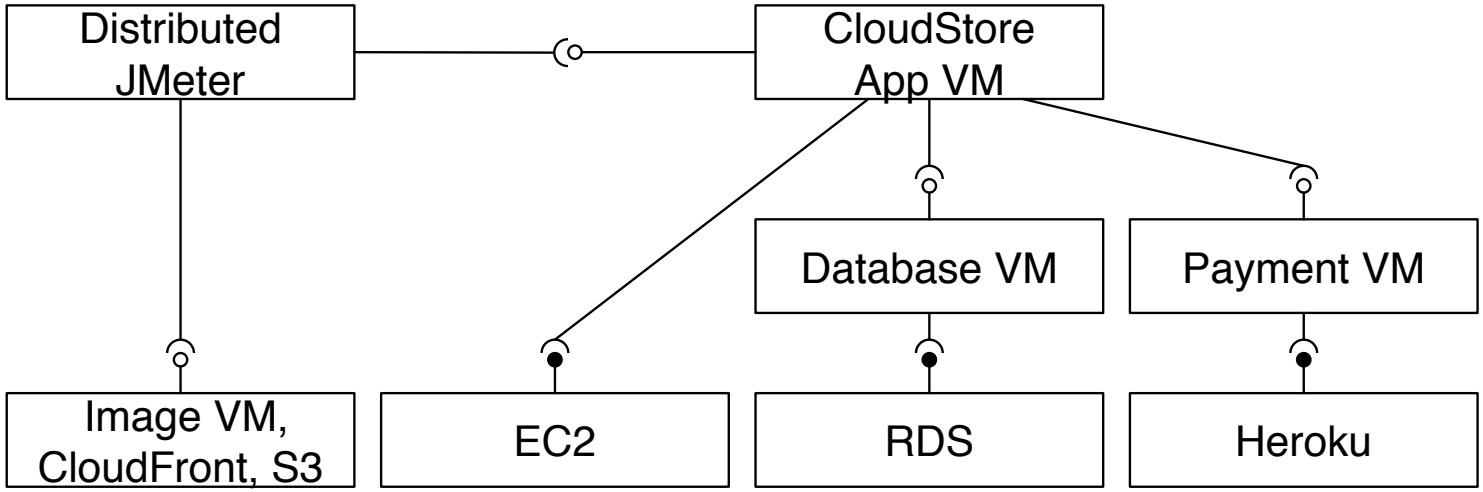
Scalability

"The ability of a service to increase its capacity
by consuming more resources
in the resource space."



CloudStore

- A cloud-based implementation of the deprecated benchmark specification TPC-W
  - TPC-W still widely used in the scientific community
  - Realistic three-tier enterprise application
- Online book store with 14 operations for browsing and buying books
- Metric: 90 percentile response times
- Each operation has individual response time thresholds, from 3 to 20 seconds
- Work parameters: Focus on 10 000 books and 288 000 customers

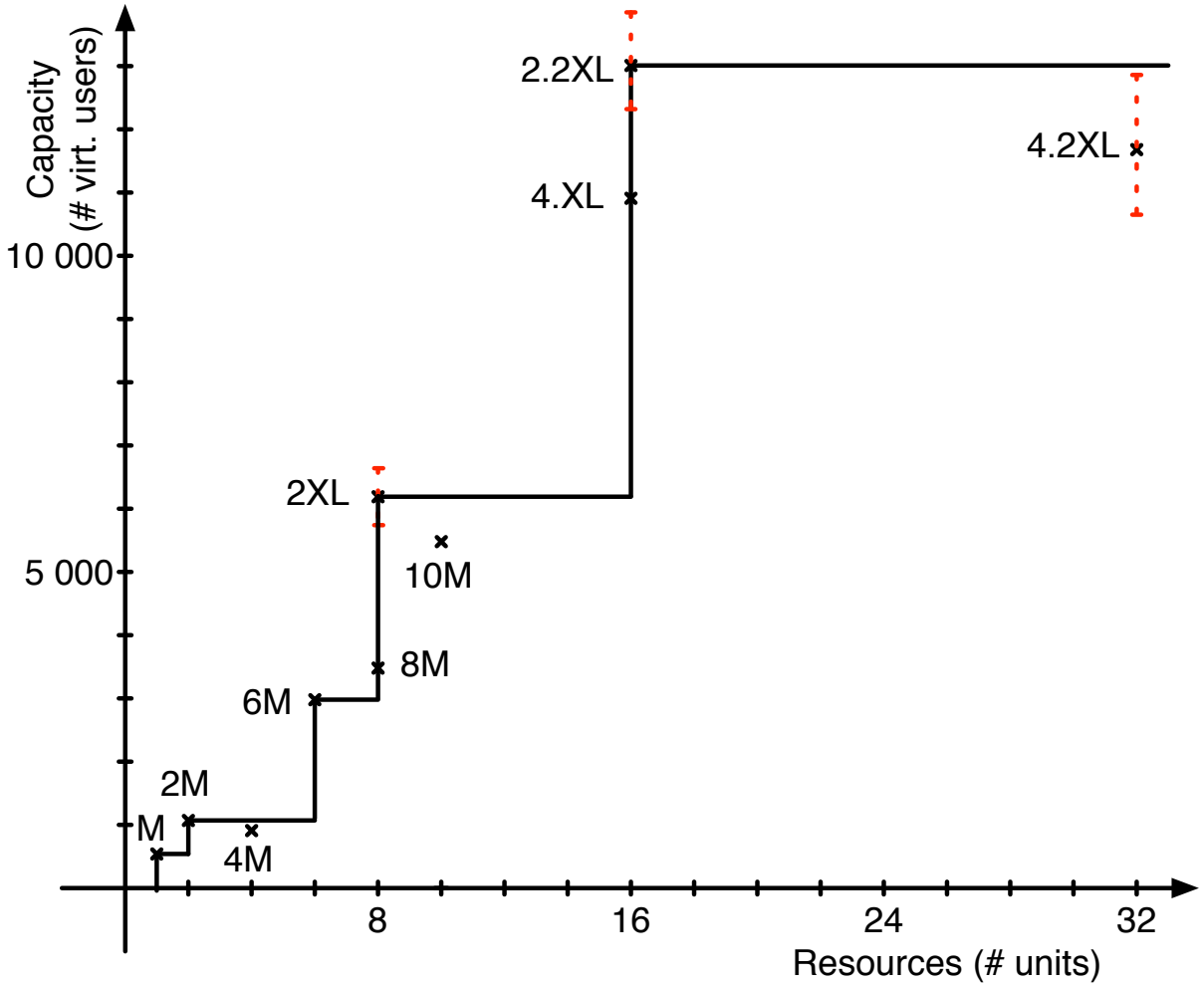


CloudStore AWS Measurements

Instances	Cost	Cap.	#	Conf.	U_{App}	U_{DB}
M:L	0.27	563	1		75	6
2.M:L	0.35	1 125	1		85	14
3.M:L	0.42	1 625	1		68	17
M:XL	0.47	563	1		87	3
2.M:XL	0.55	1 125	1		71	5
4.M:XL	0.69	2 250	1		74	12
XL:XL	0.69	3 406	6	$\pm$ 155	66	18
6.M:XL	0.84	3 438	1		75	20
M:2XL	0.86	563	1		76	2
2.M:2XL	0.93	1 094	1		69	3
8.M:XL	0.98	5 313	1		93	33
2.XL:XL	0.99	9 313	4	$\pm$ 1 021	89	55
4.M:2XL	1.08	938	1		99	3
6.M:2XL	1.22	3 500	1		81	10
8.M:2XL	1.37	4 500	1		80	14
2XL:2XL	1.37	6 219	4	$\pm$ 450	58	15
10.M:2XL	1.52	5 500	1		66	15
4.XL:XL	1.57	10 302	6	$\pm$ 582	44	71
4.XL:2XL	1.96	10 938	1		21	69
2.2XL:2XL	1.96	13 039	8	$\pm$ 764	56	38
4.2XL:2XL	3.13	11 709	6	$\pm$ 1 102	25	58

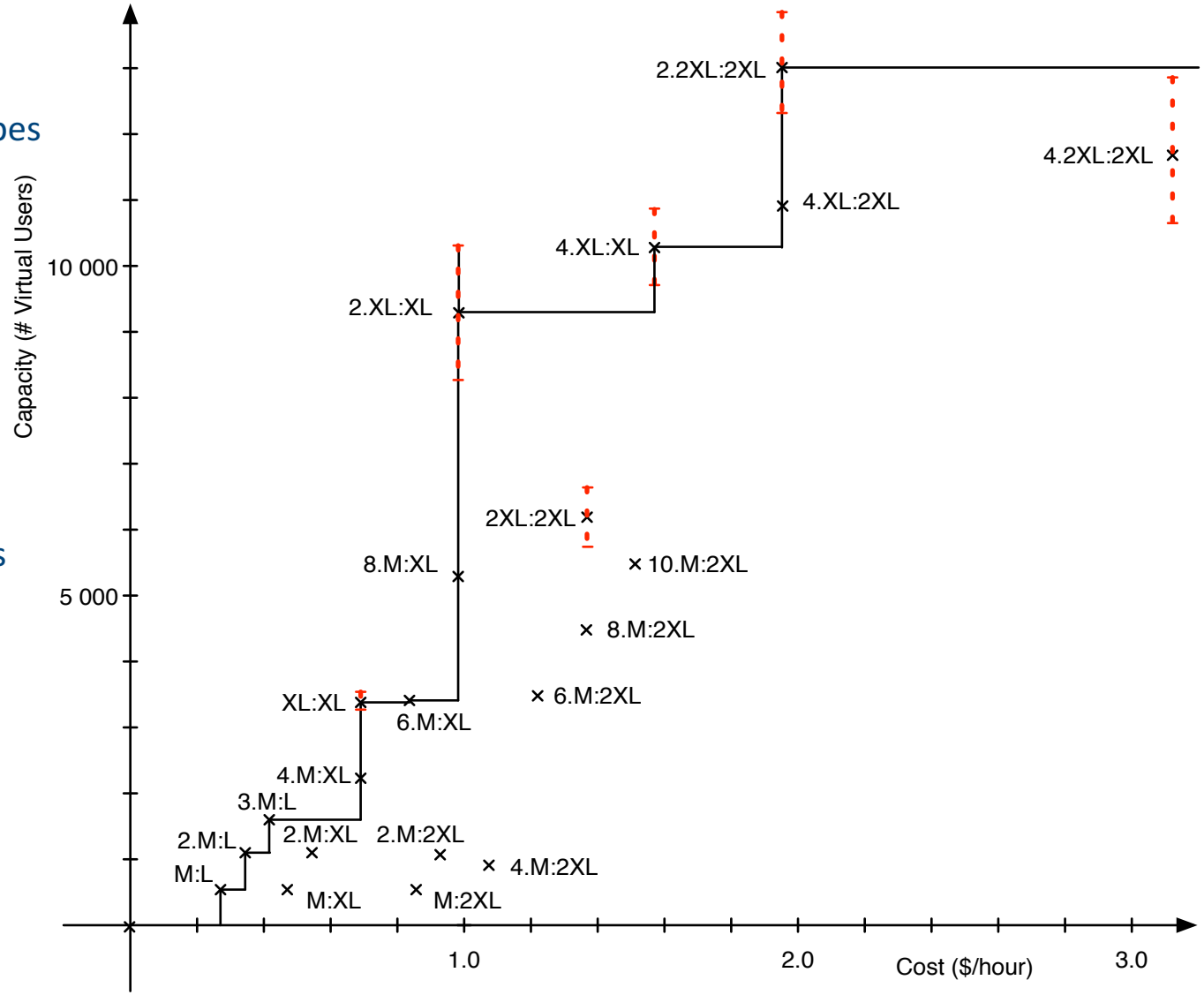
Resource Scalability Metric for CloudStore

- One database instance
  - Always db.m3.2xlarge
- Application instances
  - m3 instance types
  - Vertical & horizontal scaling
  - 2XL=2.XL=4.L=8.M
    - $ 0.585 +- 0.001
- Work
  - 10 000 books
  - 288 000 customers
- Quality metric
  - 90 percentile resp.
- Quality threshold
  - 3 to 20 sec



Cost Scalability Metric for CloudStore

- Database instances
  - db.m3 instance types
  - Vertical scaling
- Application instance
  - m3 instance types
  - Vertical & horizontal scaling
- Work
  - 10 000 books
  - 288 000 customers
- Quality metric
  - 90 percentile resp.
- Quality threshold
  - 3 to 20 sec

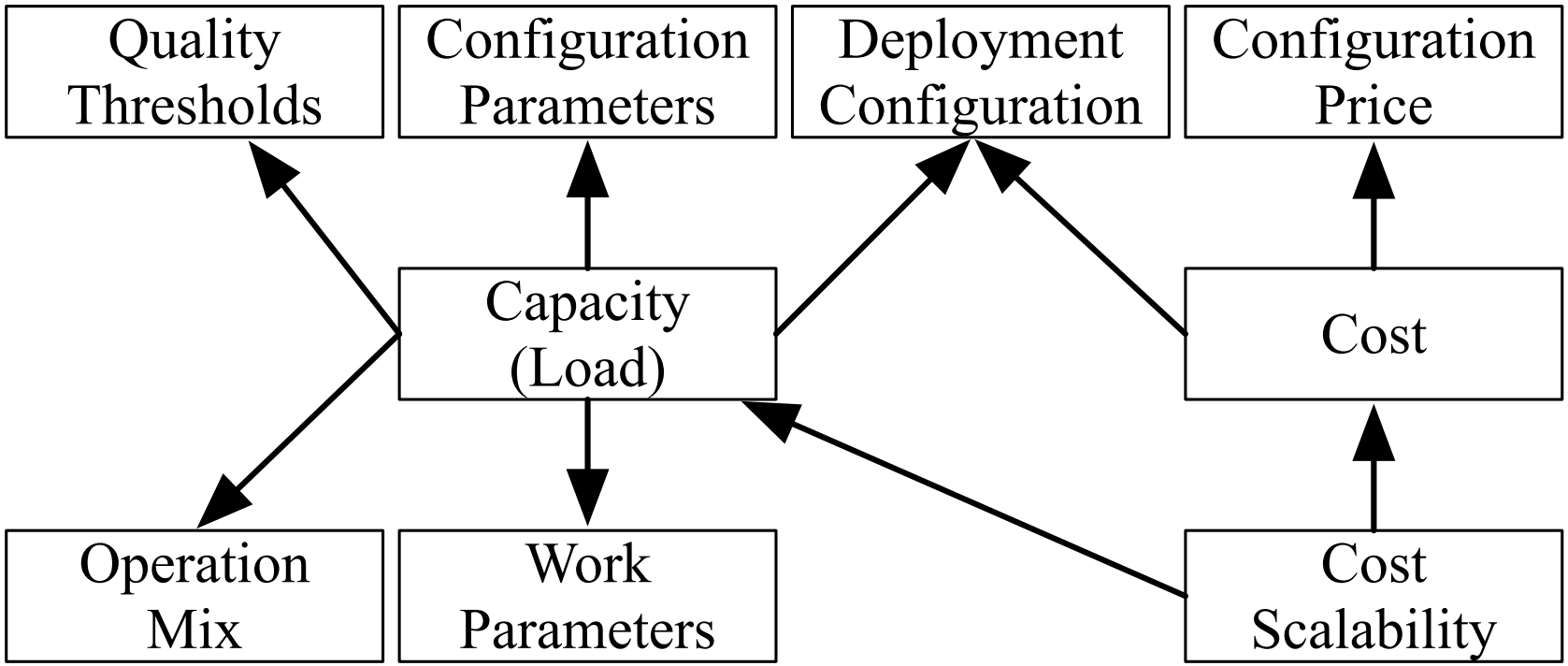


Limitations and Future Work

- CloudStore: three-tier session-based enterprise application
  - Experiences with more types of applications
- Automated resource space exploration
  - Reduce human time, but we still have cost of underlying cloud services
- Use model instead of scalability evaluation
  - To parameterize such a model is demanding, especially with resource demands

Questions?

Relation Between Concepts



Scalability

- Capacity: the maximum load a service can handle as bound by its SLA and a given operation mix, work parameters and think time
- A service has operations: ways of interacting with the service
- Workload:
  - Work: amount of data, e.g., 50 000 documents
  - Load: how often operations are invoked, e.g. 5/sec or 1 000 simult. users
- Quality (Service Level Agreements, SLAs):
  - Quality metrics, e.g. average response times, 90 percentile response times
  - Quality thresholds, e.g. 5 seconds average response times

Related Work

- Lehrig et al: systematic literature review: detail this definition with work, load, resource space
- Tsai et al.: only validated scalability metric for cloud computing: differentiate between work and load
- Bondi: richer with explicit work, load and quality thresholds, resource space
- Kossmann et al.: also focus on app servers and work parameters