Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Load Testing Elasticity and Performance Isolation in Shared Execution Environments

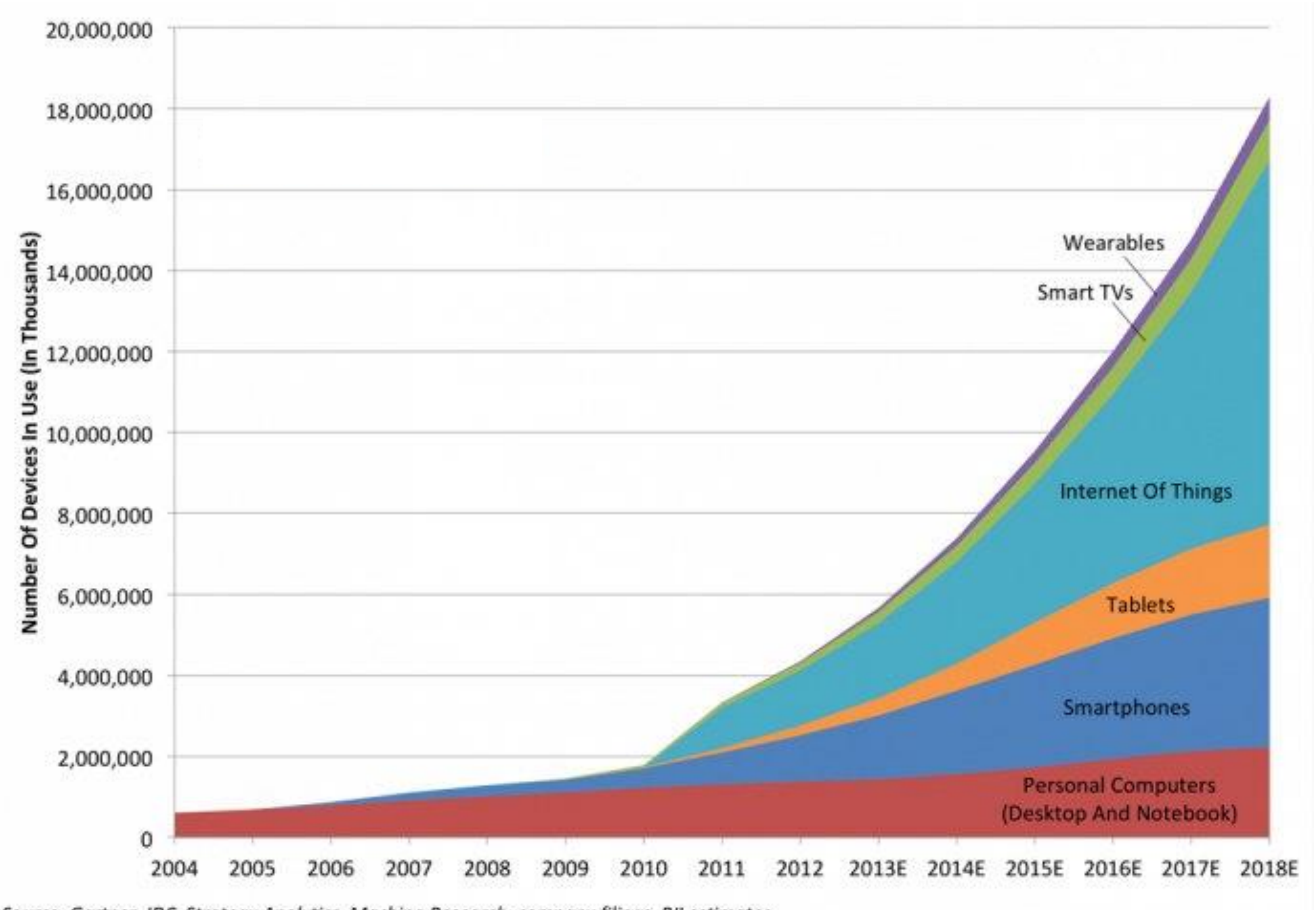
Samuel Kounev

Chair of Software Engineering
University of Würzburg

http://se.informatik.uni-wuerzburg.de/

Keynote talk, LT 2015 @ ICPE 2015, Austin, USA, Feb 1, 2015

Explosion of IT Service Clients



Source: Gartner, IDC, Strategy Analytics, Machina Research, company filings, BII estimates

Growing Data Centers

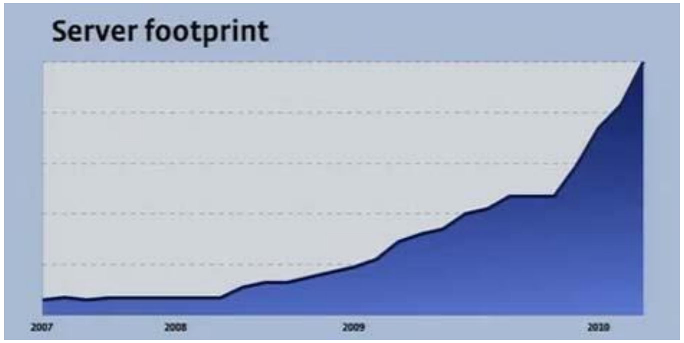
Maiden, North Carolina (Apple)
46 000 m^2

San Antonio (Microsoft)
43 000 m^2

Prineville, Oregon (Facebook)
28 000 m^2

Chicago (Digital Realty)
100 000 m^2

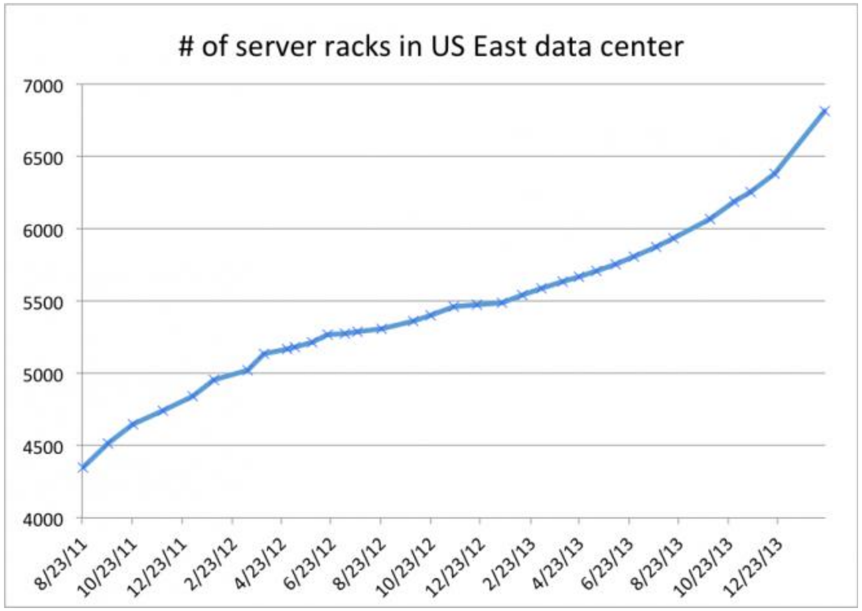
Growing Number of Servers



Facebook Servers

- Google ~ 1 Mil. (2013)
- Microsoft ~ 1 Mil. (2013)
- Facebook ~ 180K (2012)
- OVH ~ 150K (2013)
- Akamai Tech. ~ 127K (2013)
- Rackspace ~ 94K (2013)
- 1&1 Internet ~ 70K (2010)
- eBay ~ 54K (2013)
- HP/EDS ~ 380K (2013)
- …

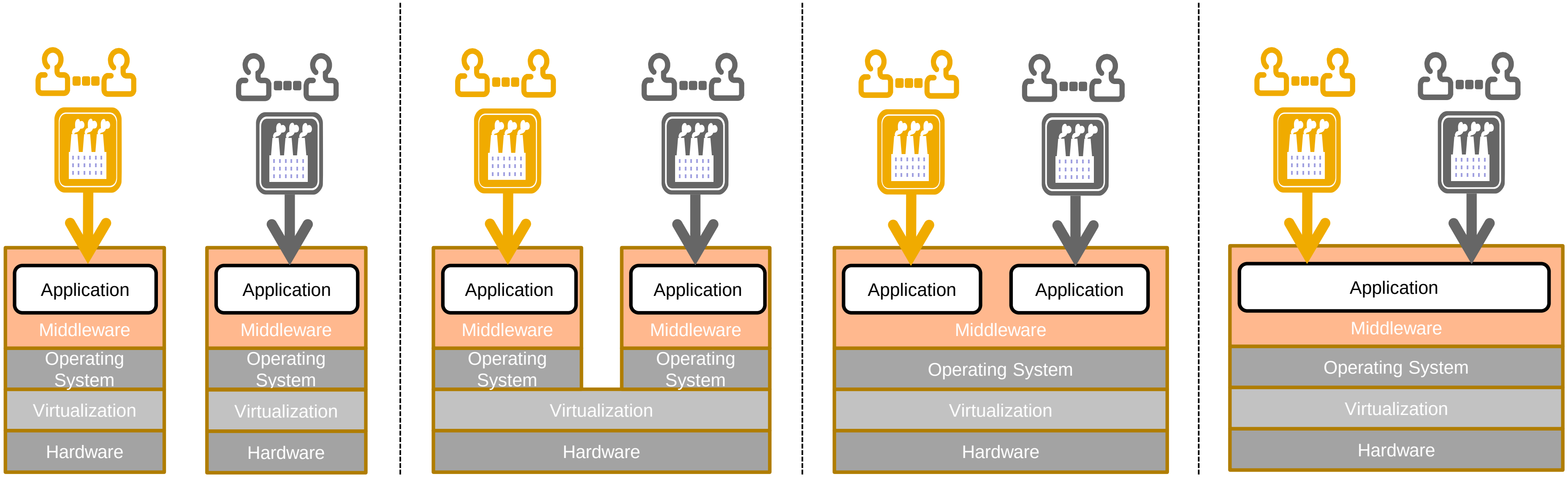
Source: http://www.datacenterknowledge.com



Amazon’s Virginia region [Src: Wired.com]

Increasing Pressure to Raise Efficiency

- Proliferation of shared execution environments
- Different forms of resource sharing (hardware and software)
  - Network, storage, and computing infrastructure
  - Software stacks



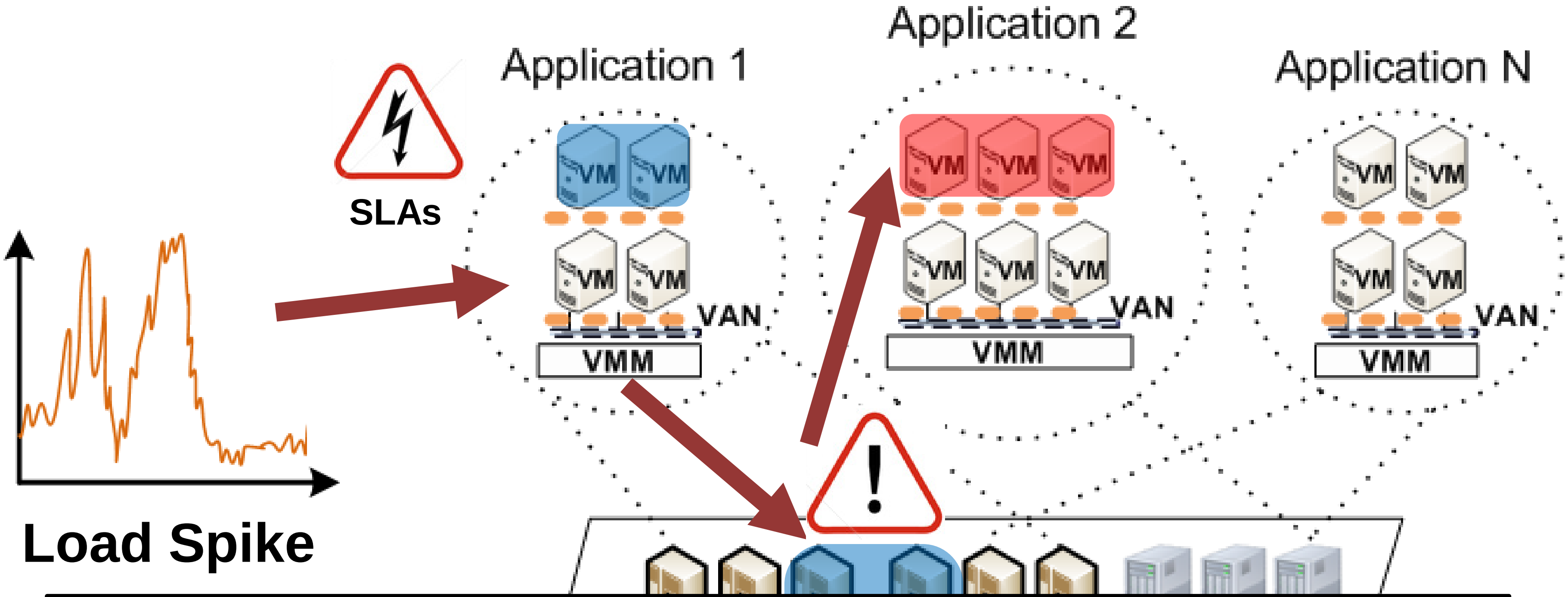
Datacenter Sharing

Virtualization

Shared Middleware

Multi-Tenancy

Challenges



Load Spike

Expand / shrink resources on-the-fly

- When exactly should a reconfiguration be triggered?
- Which particular resources should be scaled?
- How quickly and at what granularity?

Consequences

- Increased system complexity and dynamics
- Diverse vulnerabilities due to resource sharing
- Inability to provide **availability and performance** guarantees

  ⇨ **Major distinguishing factor between service offerings**
- Lack of reliable benchmarks and metrics

“You can’t ***control*** *what you can’t measure?”* (DeMarco)

“If you cannot measure it, you cannot ***improve*** *it”* (Lord Kelvin)

Descartes Tool Chain

http://descartes.tools

Related Tools

- **BUNGEE** – Elasticity benchmarking framework (homepage, publications)
- **LIMBO** – Load intensity modeling tool (homepage, publications)
- **WCF** – Workload classification & forecasting tool (homepage, publications)
- **LibReDE** - Library for resource demand estimation (homepage, publications)
- **hInjector** – Security benchmarking tool (homepage, publications)
- **DML** – Descartes Modeling Language (homepage, publications)
- **DML Bench** (homepage, publications)
- **DQL** – Declarative performance query language (homepage, publications)
- **Further relevant research**
  - **http://descartes-research.net/research/research_areas/**
  - **Self Aware Computing** (publications)

The Focus of this Talk

Metrics and benchmarks for quantitative evaluation of

1. Resource elasticity
2. Performance isolation

in shared execution environments

- Virtualized infrastructures
- Multi-tenant applications

HAVE YOU TESTED YOUR CODE UNDER STRESS?
NO, BUT I'VE WRITTEN IT UNDER STRESS

[geek & poke]

Credits

Nikolas Herbst + MSc students (elasticity)

Rouven Krebs + MSc students (performance isolation)

Part I: Resource Elasticity

Main references

N. Herbst, A. Weber, H. Groenda and S. Kounev. **BUNGEE: Benchmarking Resource Elasticity of Cloud Environments**. Submitted to *6th ACM/SPEC Intl. Conf. on Performance Engineering (ICPE 2015)*.

N. Herbst, S. Kounev and R. Reussner. **Elasticity in Cloud Computing: What it is, and What it is Not**. In *Proc. of the 10th Intl. Conf. on Autonomic Computing (ICAC 2013)*, San Jose, CA, June 24-28, 2013. USENIX. [slides | http | .pdf]

Further references

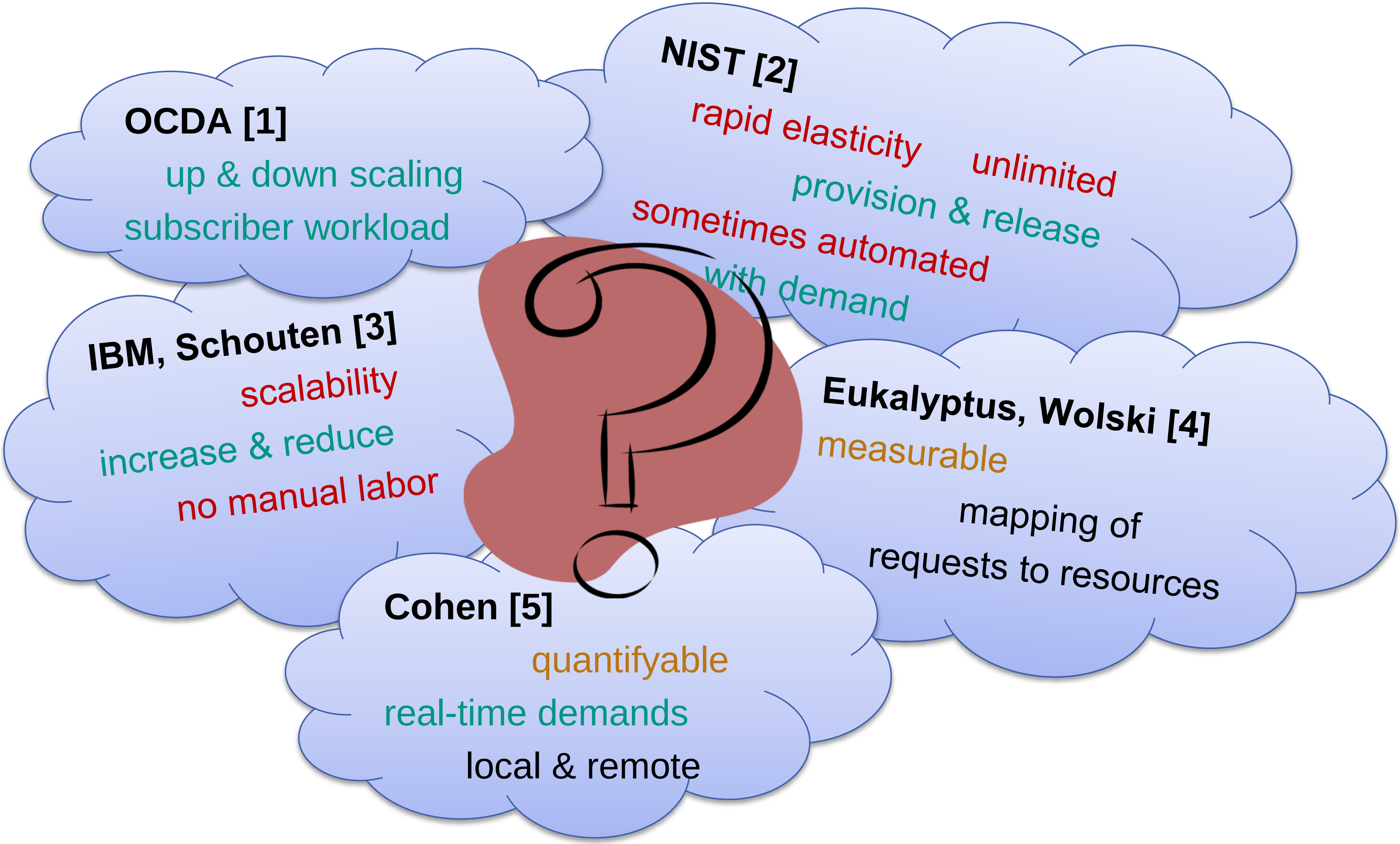
N. Herbst, N. Huber, S. Kounev and E. Amrehn. **Self-Adaptive Workload Classification and Forecasting for Proactive Resource Provisioning**. *Concurrency and Computation - Practice and Experience, John Wiley and Sons, Ltd.*, 26(12):2053-2078, 2014. [DOI | http]

J. von Kistowski, N. Herbst and S. Kounev. **LIMBO: A Tool For Modeling Variable Load Intensities** (Demonstration Paper). In *Proc. of the 5th ACM/SPEC Intl. Conf. on Performance Engineering (ICPE 2014)*, Dublin, Ireland, March 22-26, 2014. ACM. [DOI | slides | http | .pdf]

J. von Kistowski, N. Herbst and S. Kounev. **Modeling Variations in Load Intensity over Time**. In *Proc. of the 3rd Intl. Workshop on Large-Scale Testing (LT 2014), co-located with ICPE 2014*, Dublin, Ireland, March 22, 2014. ACM. [DOI | slides | http | .pdf]

A. Weber, N. Herbst, H. Groenda and S. Kounev. **Towards a Resource Elasticity Benchmark for Cloud Environments**. In *Proc. of the 2nd Intl. Workshop on Hot Topics in Cloud Service Scalability (HotTopiCS 2014), co-located with ICPE 2014*, March 22, 2014. ACM. [slides | .pdf]

What People Say Elasticity is…



OCDA [1]

up & down scaling

subscriber workload

NIST [2]

rapid elasticity

unlimited

provision & release

sometimes automated

with demand

IBM, Schouten [3]

scalability

increase & reduce

no manual labor

Eukalyptus, Wolski [4]

measurable

mapping of requests to resources

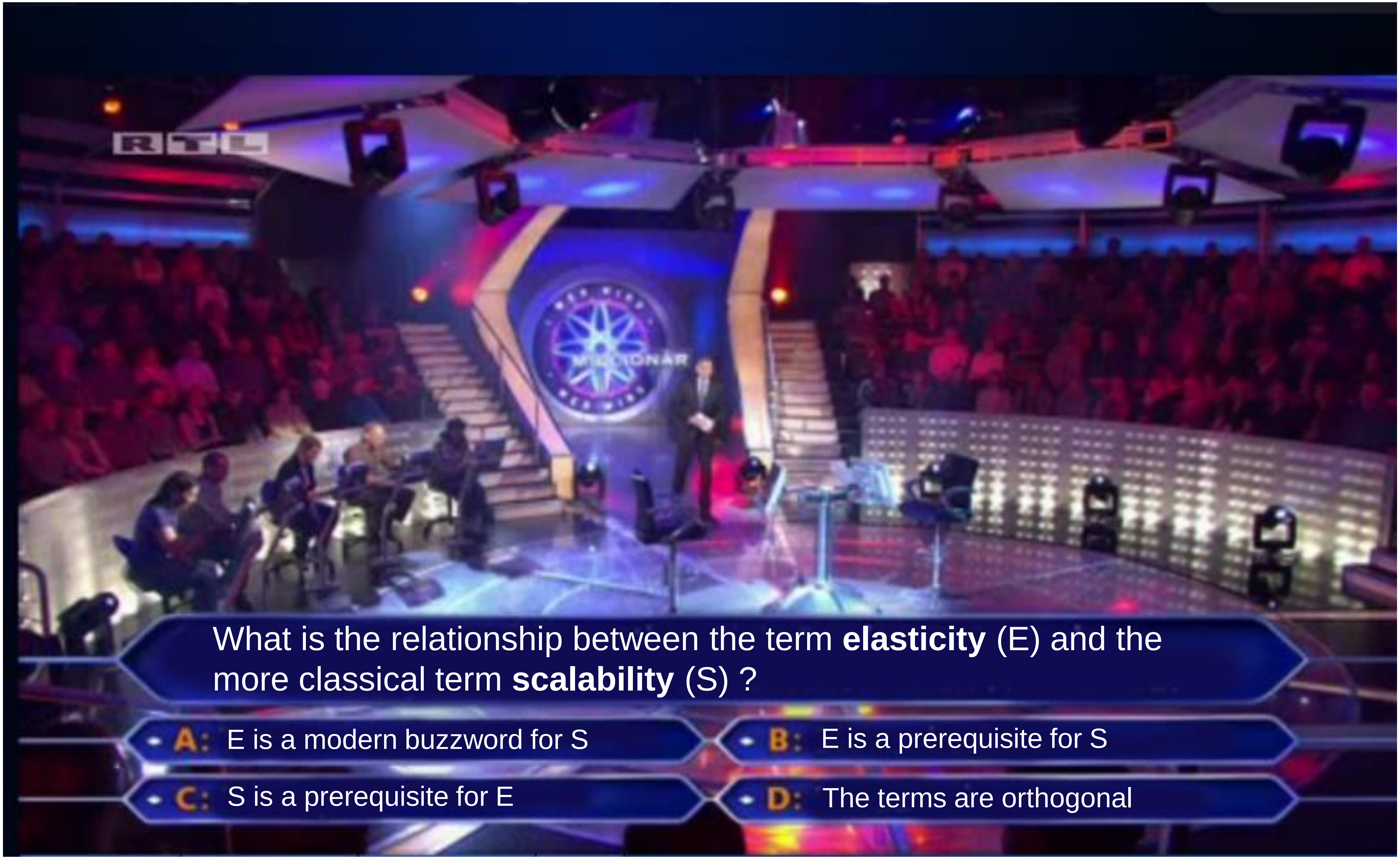
Cohen [5]

quantifyable

real-time demands

local & remote

Elasticity vs. Scalability



What is the relationship between the term **elasticity** (E) and the more classical term **scalability** (S) ?

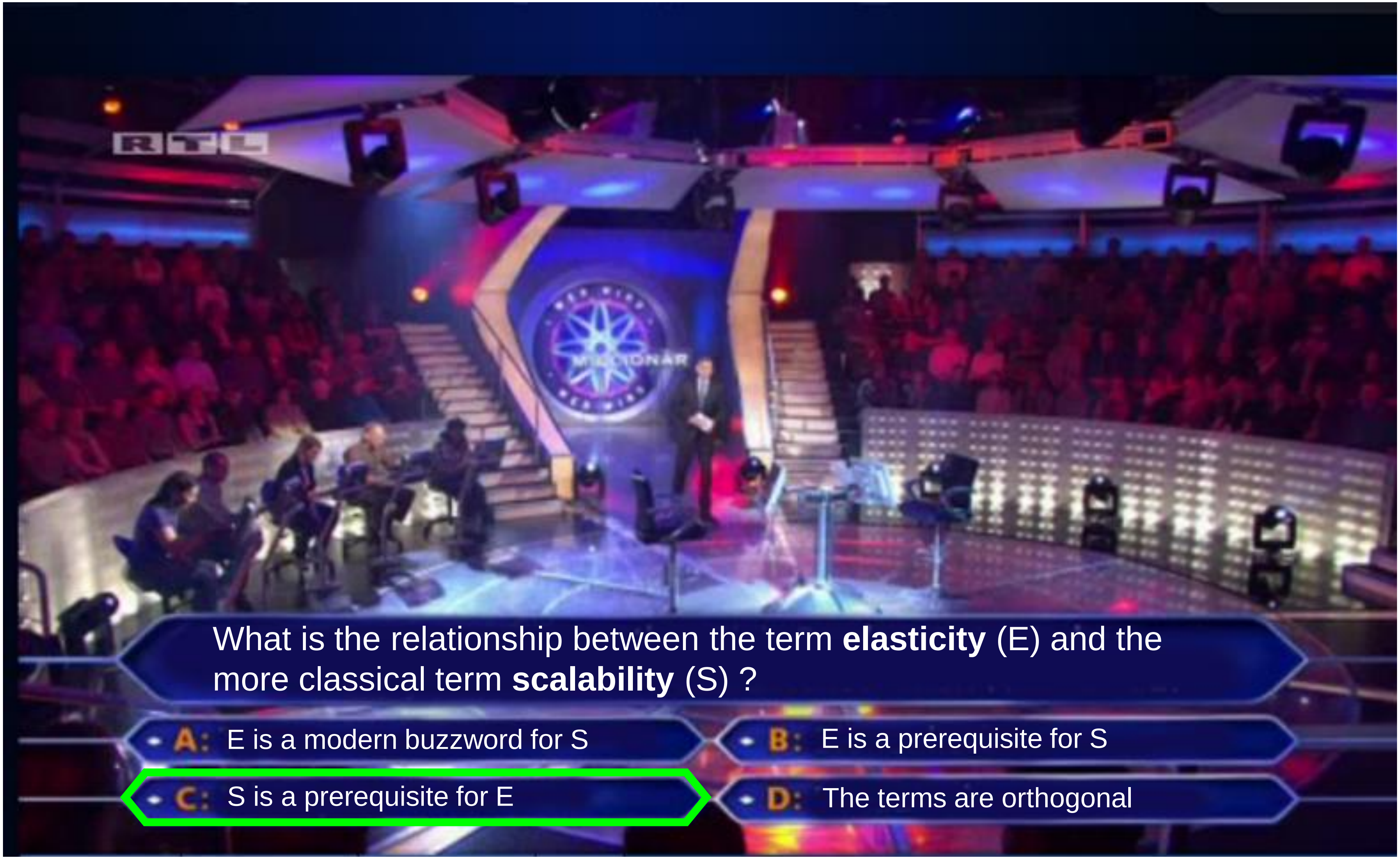
A: E is a modern buzzword for S

B: E is a prerequisite for S

C: S is a prerequisite for E

D: The terms are orthogonal

Elasticity vs. Scalability



What is the relationship between the term **elasticity** (E) and the more classical term **scalability** (S) ?

A: E is a modern buzzword for S

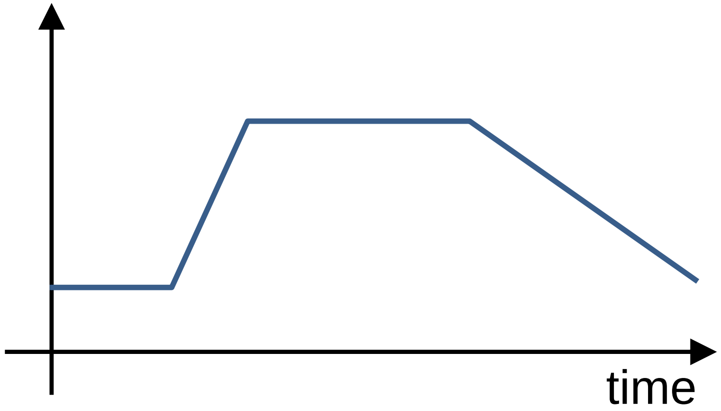
B: E is a prerequisite for S

[✓ C: S is a prerequisite for E]

D: The terms are orthogonal

Elasticity

Workload intensity (e.g., # requests / sec)



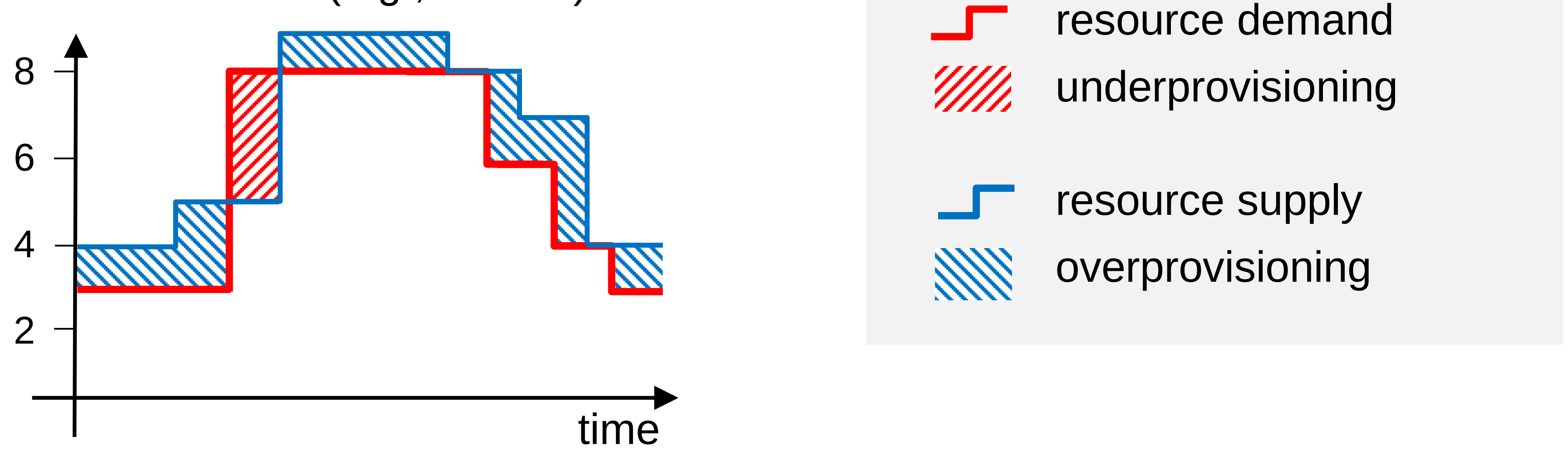
Service Level Objective (SLO)

(e.g., resp. time ≤ 2 sec, 95%)

Resource Demand

Minimal amount of resources required to ensure SLOs.

Amount of resources (e.g., # VMs)



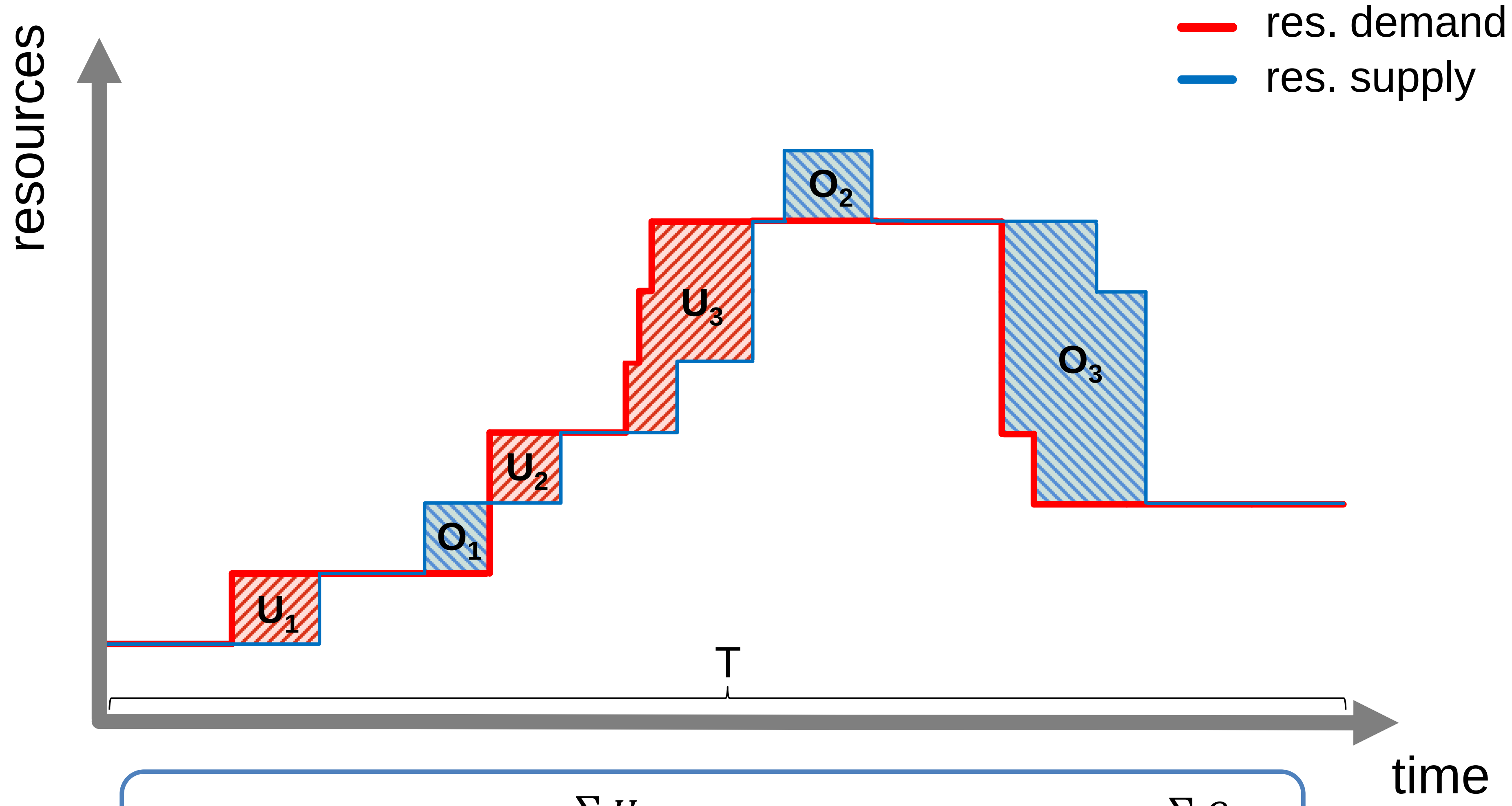
Elasticity

Def: The degree to which a system is able to **adapt** to **workload changes** by **provisioning and deprovisioning** resources in an **autonomic manner**, such that at each point in time the **available resources match** the **current demand** as closely as possible.

N. Herbst, S. Kounev and R. Reussner
Elasticity: What it is, and What it is Not.
in Proceedings of the 10th International Conference on Autonomic Computing (ICAC 2013), San Jose, CA, June 24-28, 2013.

http://en.wikipedia.org/wiki/Elasticity_(cloud_computing)

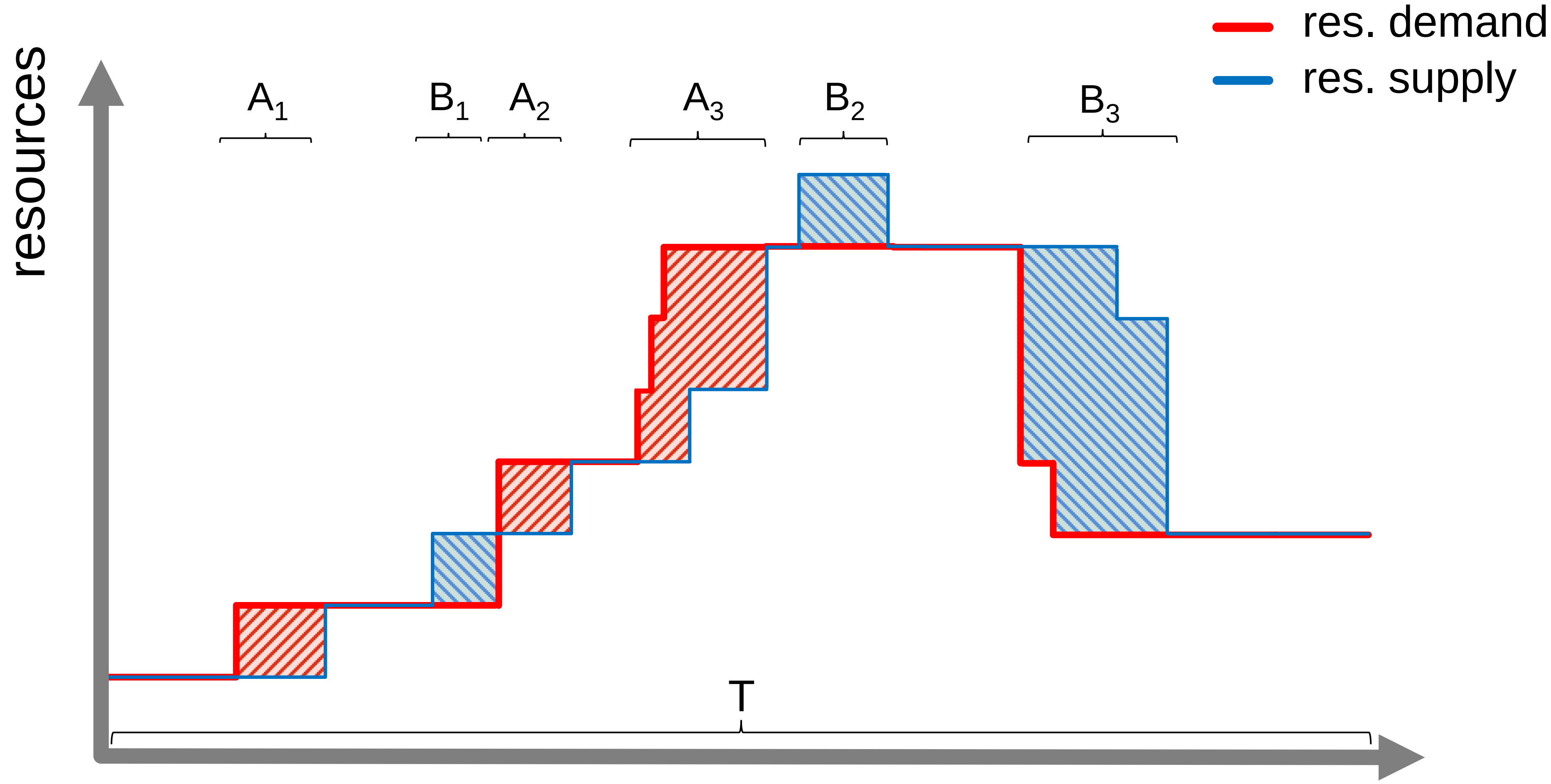
Metrics: Accuracy



(1) accuracy$_U$: $\frac{\sum U}{T}$

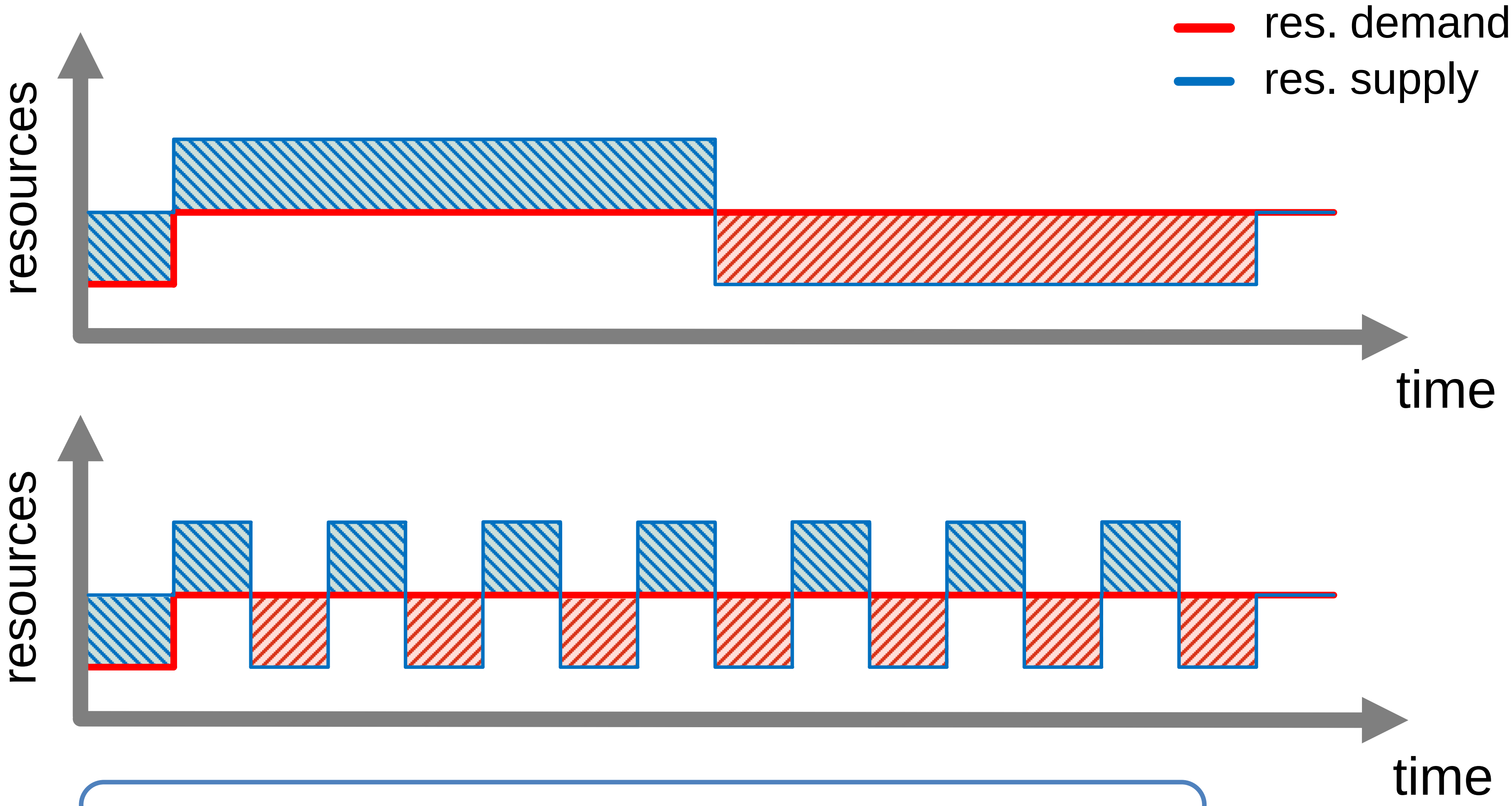
(2) accuracy$_O$: $\frac{\sum O}{T}$

Metrics: Timeshare



(3) timeshare$_U$: $\frac{\sum A}{T}$ (4) timeshare$_O$: $\frac{\sum B}{T}$

Metrics: Jitter

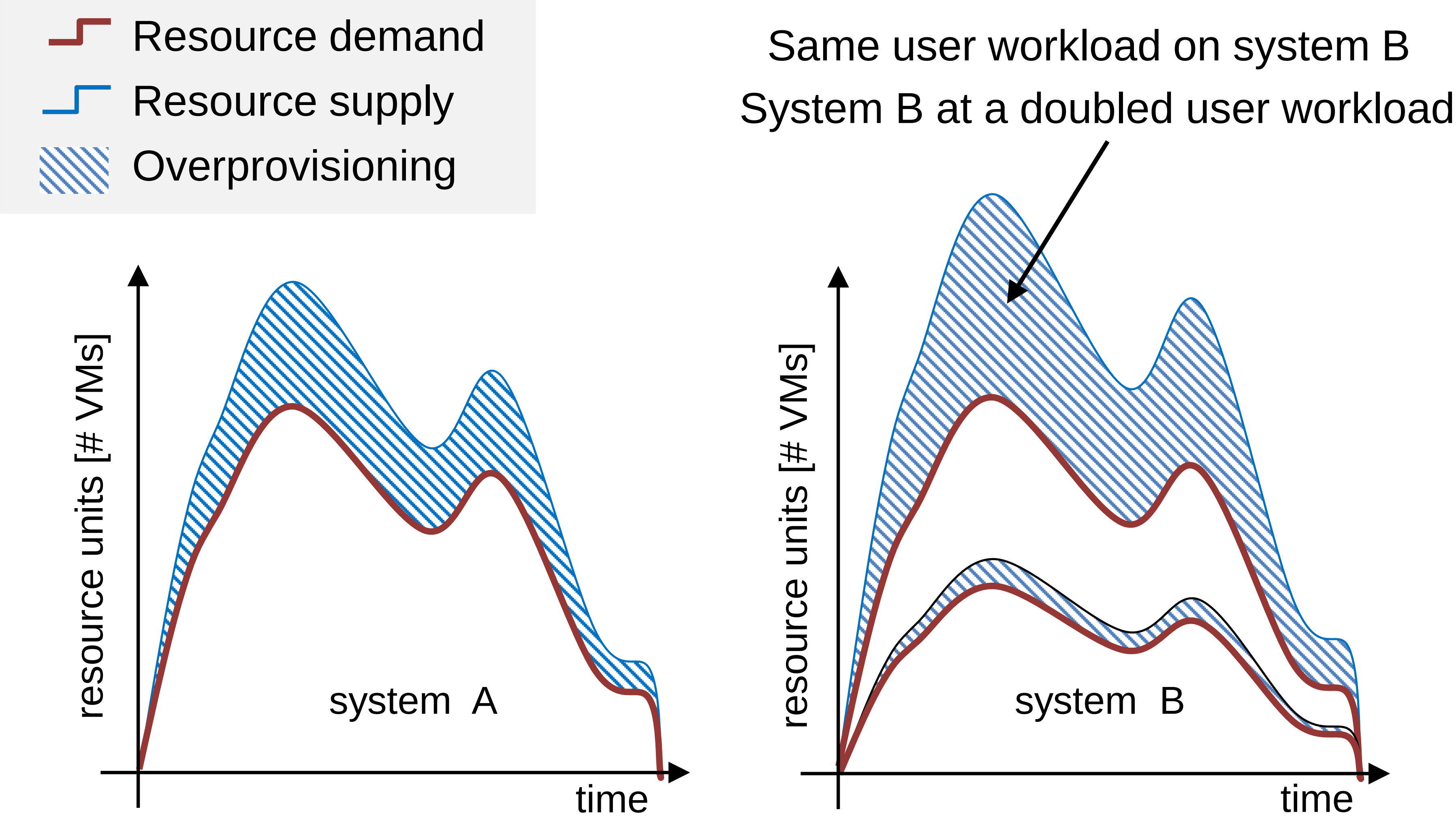


(5) jitter: $$\frac{E_S - E_D}{T}$$

E_D: # demand changes

E_S: # supply changes

Elasticity Benchmarking



▬ Resource demand

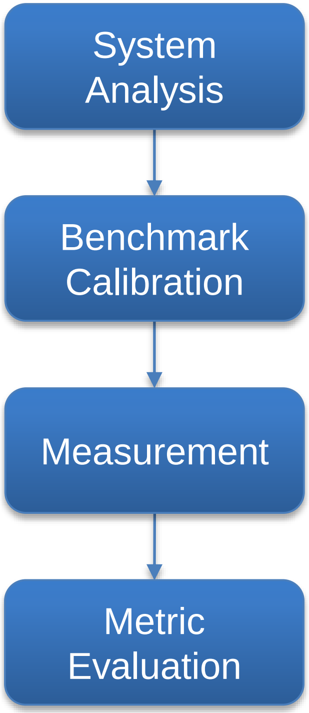
▬ Resource supply

▨ Overprovisioning

Same user workload on system B

System B at a doubled user workload

Elasticity Benchmarking Approach



Analyze efficiency & scaling behavior of underlying resources

Adjust load profile

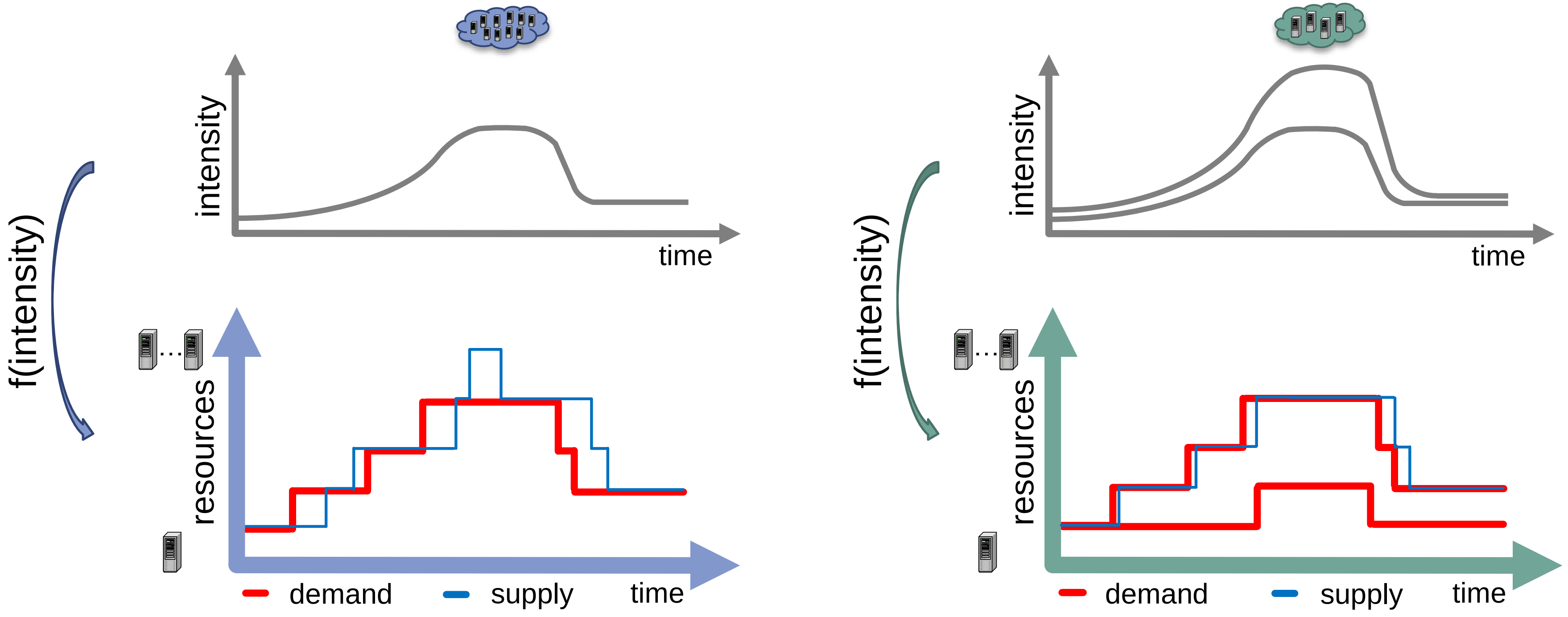
Expose SUT to varying load
&
monitor resource supply & demand

Compute elasticity metrics
(accuracy & timing)

N. Herbst, A. Weber, H. Groenda and S. Kounev. **BUNGEE: Benchmarking Resource Elasticity of Cloud Environments**. *Submitted to SEAMS 2015*.

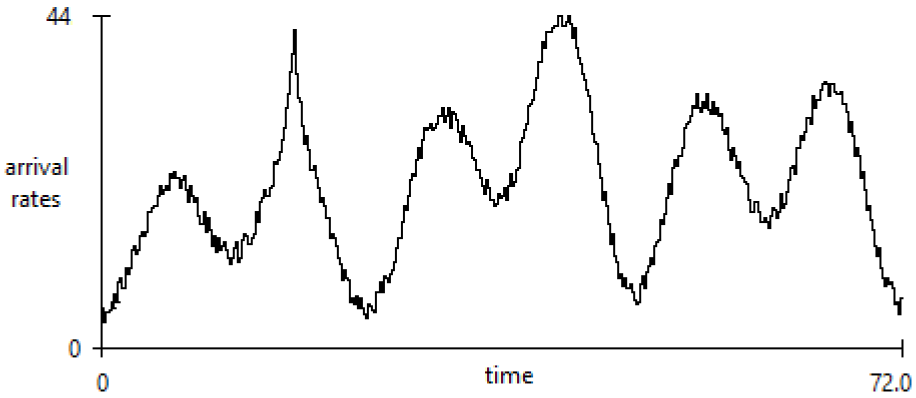
Step 2: Benchmark Calibration

- Goal: Induce same resource demand on all systems



- Approach: Adjust load intensity profile to overcome
  - Different efficiency of underlying resources
  - Different scalability

LIMBO: A Tool For Modeling Variable Load Intensities

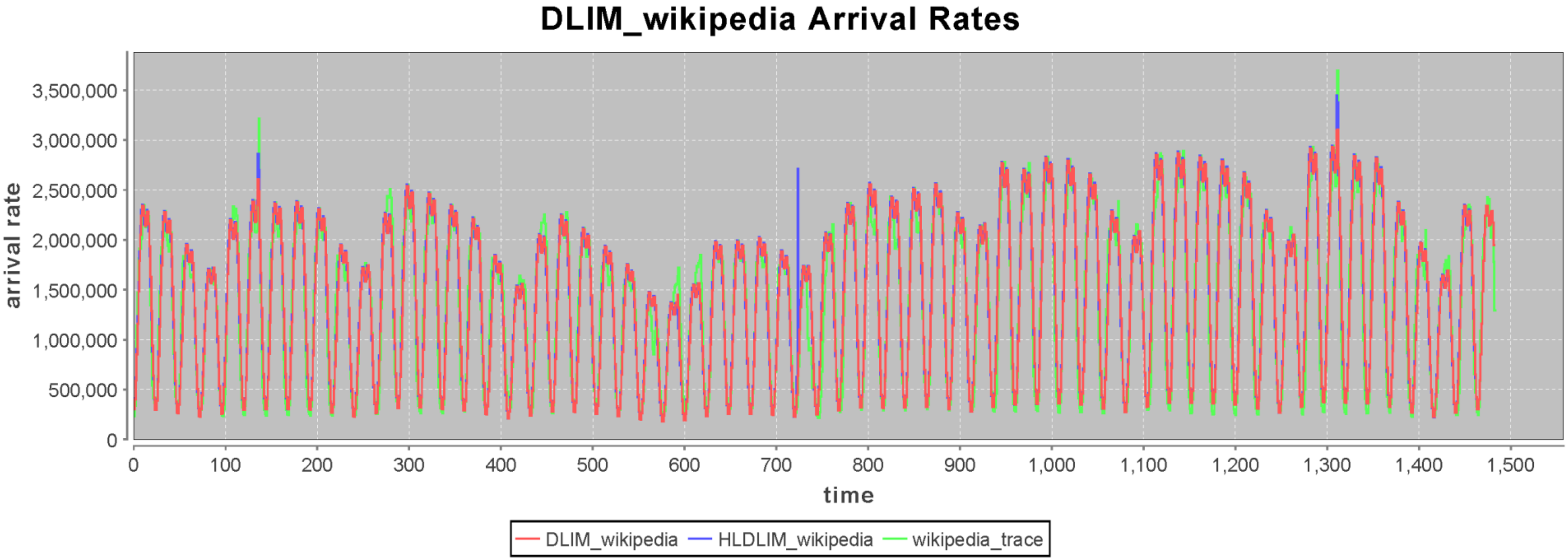


http://descartes.tools/limbo

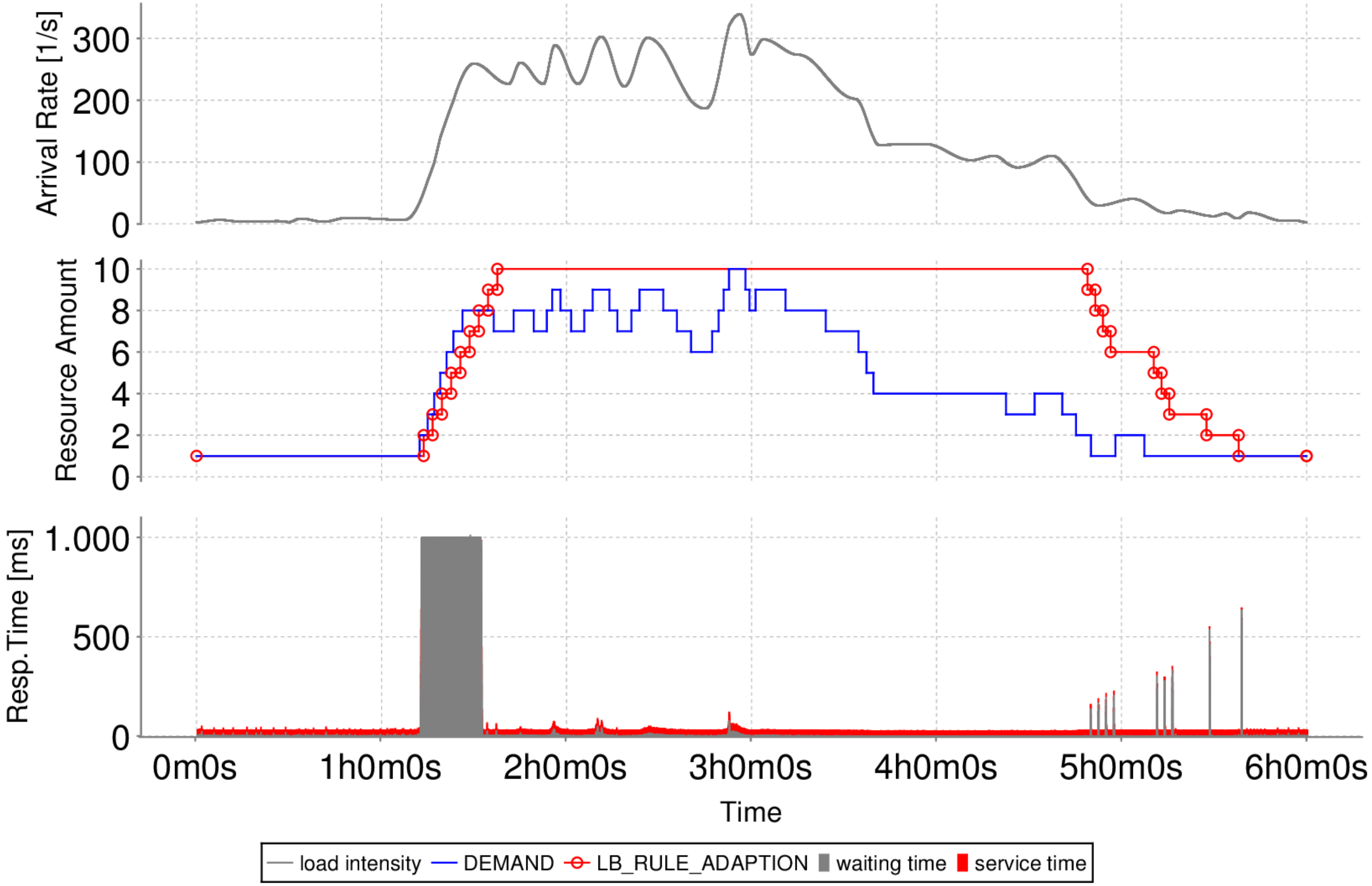
J. von Kistowski, N. Herbst and S. Kounev. **LIMBO: A Tool For Modeling Variable Load Intensities** (Demonstration Paper). In *Proc. of the 5th ACM/SPEC Intl. Conf. on Performance Engineering (ICPE 2014)*, Dublin, Ireland, March 22-26, 2014. ACM. [DOI | slides | http | .pdf]

J. von Kistowski, N. Herbst and S. Kounev. **Modeling Variations in Load Intensity over Time**. In *Proc. of the 3rd Intl. Workshop on Large-Scale Testing (LT 2014)*, Dublin, Ireland, March 22, 2014. ACM. [DOI | slides | http | .pdf]

Example: Wikipedia Workload



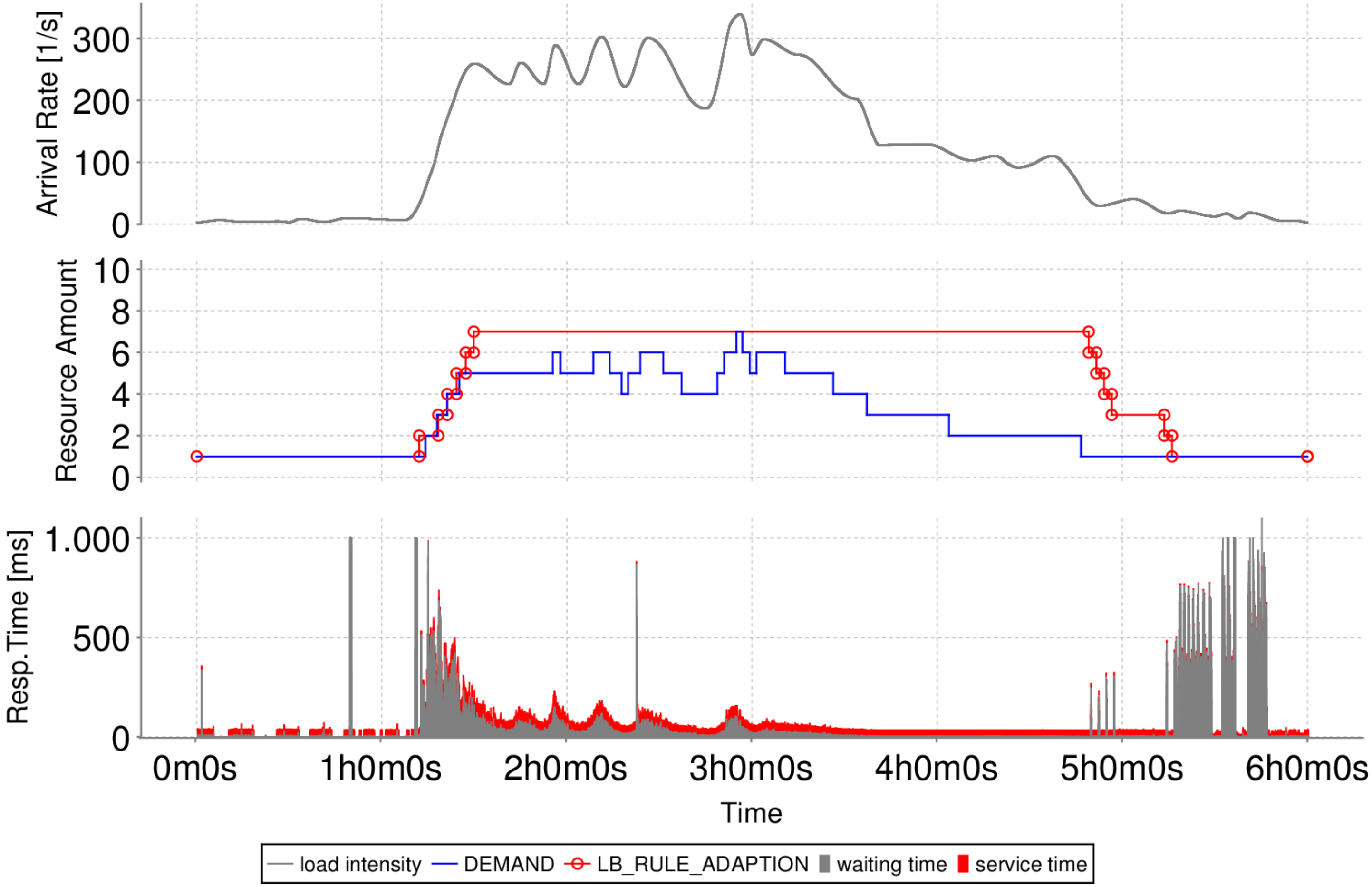
Case Study: CloudStack (CS) - 1Core



CloudStack Settings
quietTime
120s
condTrueDur
30s
threshUp
65%
threshDown
10%

Configuration	accuarcy$_O$ [res. units]	accuracy$_U$ [res. units]	timeshare$_O$ [%]	timeshare$_U$ [%]	jitter [adap/min.]	elastic speedup	violations [%]
CS – 1Core	2.423	0.067	66.1	4.8	-0.067	**1.046**	7.6

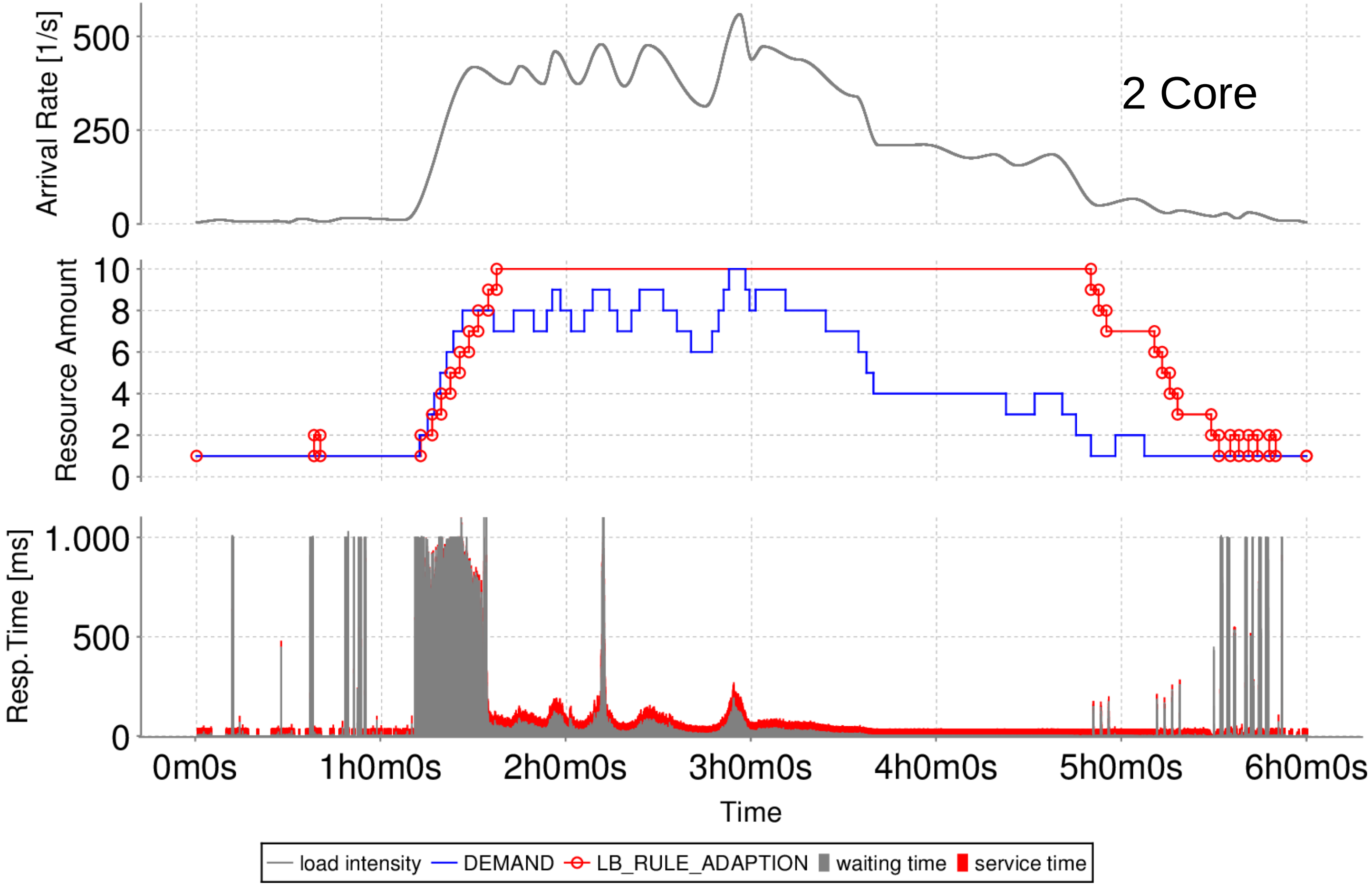
CloudStack (CS) – 2 Core – no adjustment



CloudStack Settings
quietTime
120s
condTrueDur
30s
threshUp
65%
threshDown
10%

Configuration	accuarcy$_{O}$ [res. units]	accuracy$_{U}$ [res. units]	timeshare$_{O}$ [%]	timeshare$_{U}$ [%]	jitter [adap/min.]	elastic speedup	violations [%]
CS – 1Core	2.423	0.067	66.1	4.8	-0.067	**1.046**	7.6
CS – 2Core no adjustment	1.811	0.001	63.8	0.1	-0.033	**1.291**	2.1

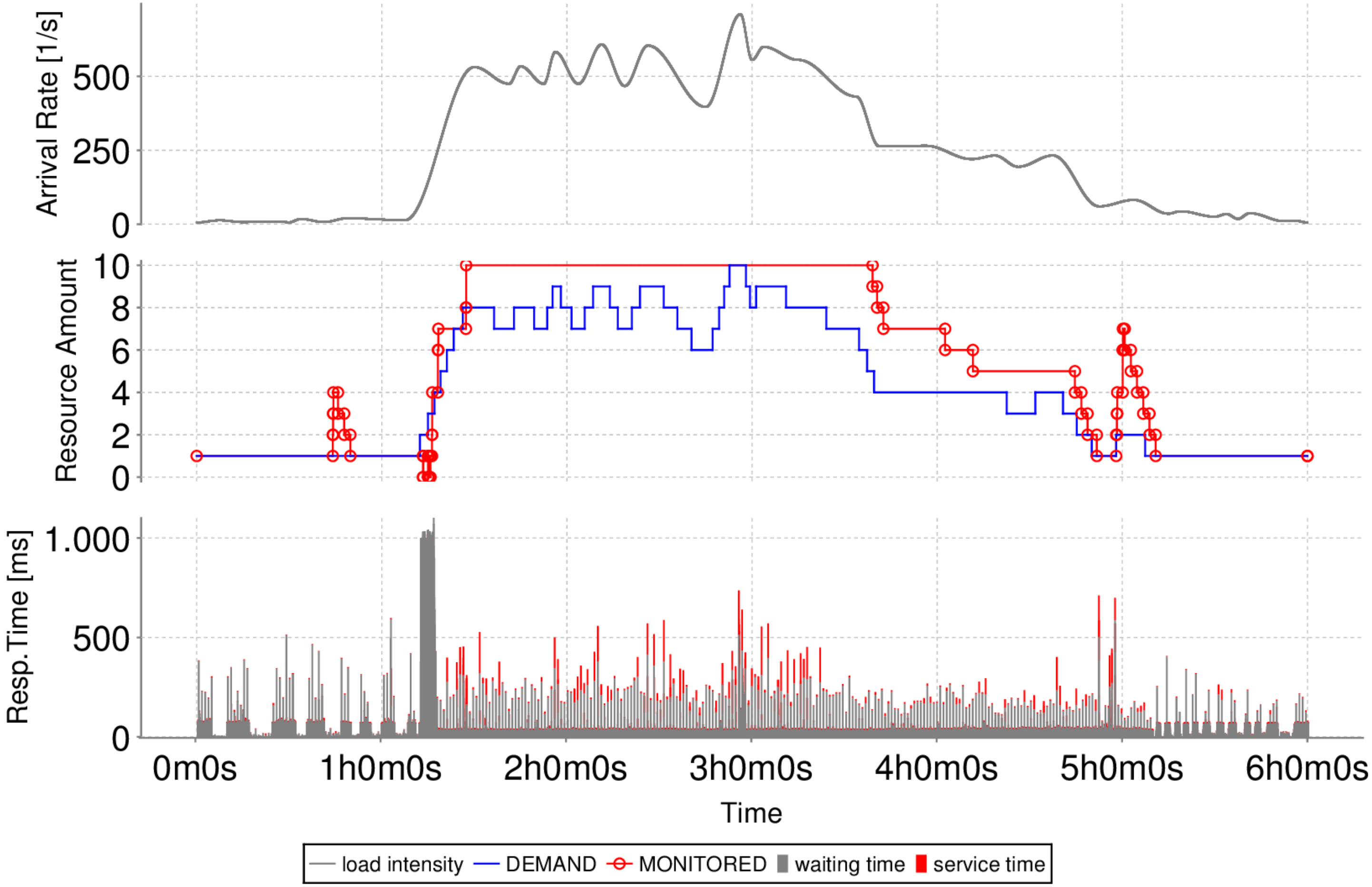
CloudStack (CS) – 2 Core – adjusted



CloudStack Settings
quietTime
120s
condTrueDur
30s
threshUp
65%
threshDown
10%

Configuration	accuarcy$_O$ [res. units]	accuracy$_U$ [res. units]	timeshare$_O$ [%]	timeshare$_U$ [%]	jitter [adap/min.]	elastic speedup	violations [%]
CS – 1Core	2.423	0.067	66.1	4.8	-0.067	**1.046**	7.6
CS – 2Core no adjustment	1.811	0.001	63.8	0.1	-0.033	**1.291**	2.1
CS – 2Core adjusted	2.508	0.061	67.1	4.5	-0.044	**1.025**	8.2

Amazon Web Services (AWS) - m1.small



CloudStack Settings
quietTime
60s
condTrueDur
60s
threshUp
80%
threshDown
50%
instUp/Down
3/1

Configuration	accuarcy$_O$ [res. units]	accuracy$_U$ [res. units]	timeshare$_O$ [%]	timeshare$_U$ [%]	jitter [adap/min.]	elastic speedup	violations [%]
CS – 1Core	2.423	0.067	66.1	4.8	-0.067	**1.046**	7.6
CS – 2Core adjusted	2.508	0.061	67.1	4.5	-0.044	**1.025**	8.2
AWS - m1.small	1.340	0.019	61.6	1.4	0.000	**1.502**	2.5

Part II: Performance Isolation

Main references

R. Krebs, C. Momm and S. Kounev. **Metrics and Techniques for Quantifying Performance Isolation in Cloud Environments**. *Elsevier Science of Computer Programming Journal (SciCo)*, Vol. 90, Part B:116-134, 2014, Elsevier B.V. [bib | .pdf]

R. Krebs, A. Wert and S. Kounev. **Multi-Tenancy Performance Benchmark for Web Application Platforms**. In *Proc. of the 13th Intl. Conf. on Web Engineering (ICWE 2013)*, Aalborg, Denmark, July 8-12, 2013. Springer-Verlag. [.pdf]

R. Krebs, C. Momm and S. Kounev. **Metrics and Techniques for Quantifying Performance Isolation in Cloud Environments.** In *Proc. of the 8th ACM SIGSOFT Intl. Conf. on the Quality of Software Architectures (QoSA 2012),* Bertinoro, Italy, June 25-28, 2012. ACM. [http | .pdf]

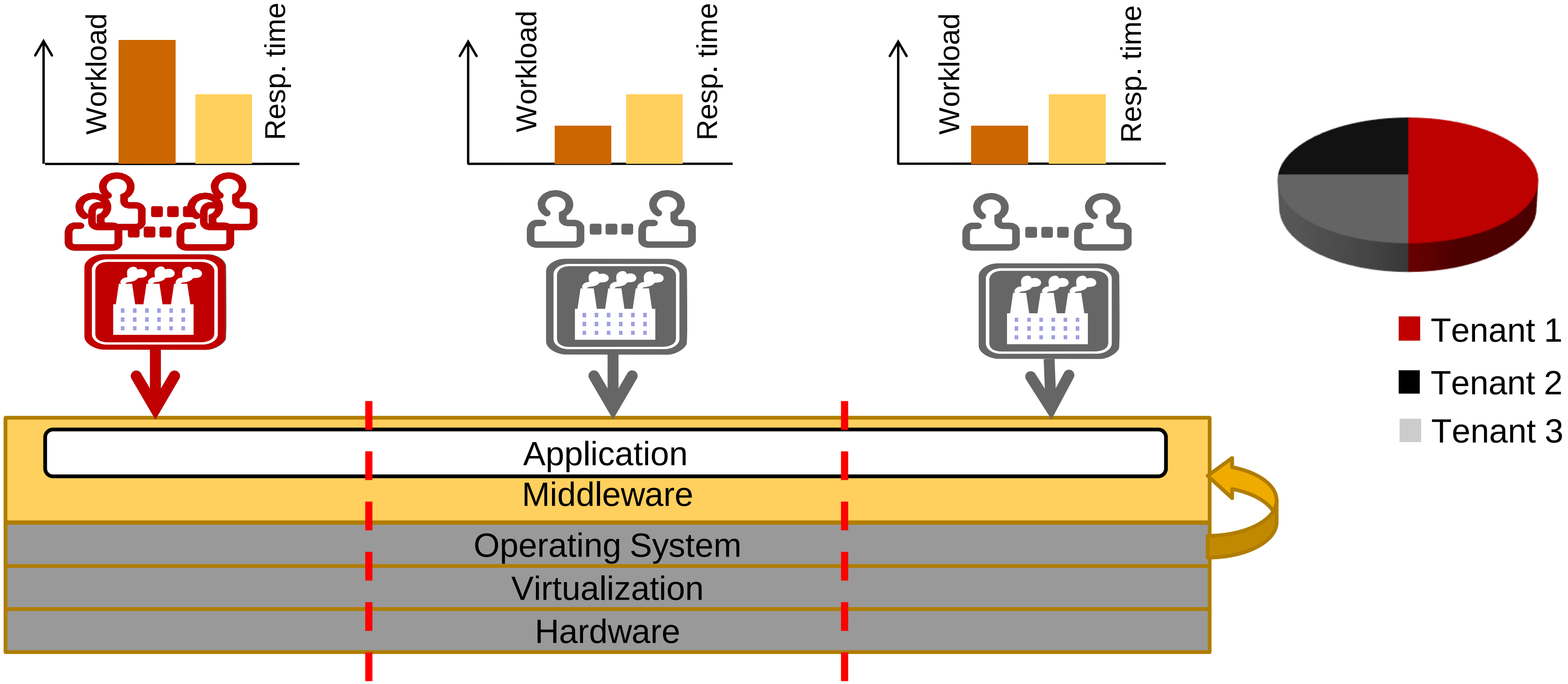
Further references

R. Krebs, S. Spinner, N. Ahmed and S. Kounev. **Resource Usage Control In Multi-Tenant Applications**. In *Proc. of the 14th IEEE/ACM Intl. Symp. on Cluster, Cloud and Grid Computing (CCGrid 2014)*, Chicago, IL, USA, May 26, 2014. IEEE/ACM. [.pdf]

R. Krebs, M. Loesch and S. Kounev. **Platform-as-a-Service Architecture for Performance Isolated Multi-Tenant Applications**. In *Proc. of the 7th IEEE Intl. Conf. on Cloud Computing*, Anchorage, USA, July 2, 2014. IEEE.

R. Krebs, C. Momm and S. Kounev. **Architectural Concerns in Multi-Tenant SaaS Applications**. In *Proc. of 2nd Intl. Conf. on Cloud Computing and Services Science (CLOSER 2012)*, Setubal, Portugal, April 18-21, 2012. [.pdf]

Example Scenario: Multi-Tenant Environments

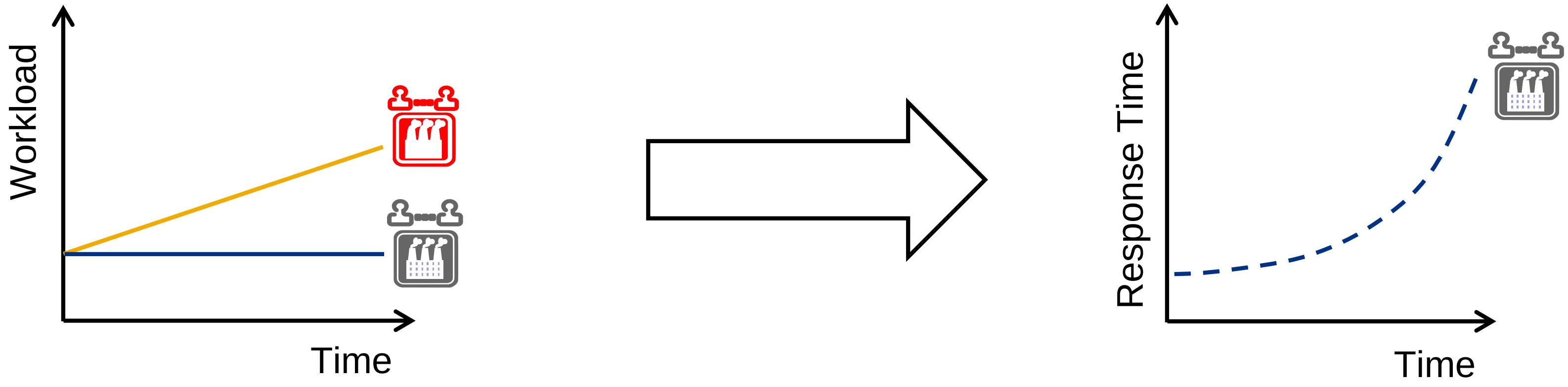


Tenants working within their assigned quota (e.g., # users) should not suffer from tenants exceeding their quotas.

Performance Isolation Metrics

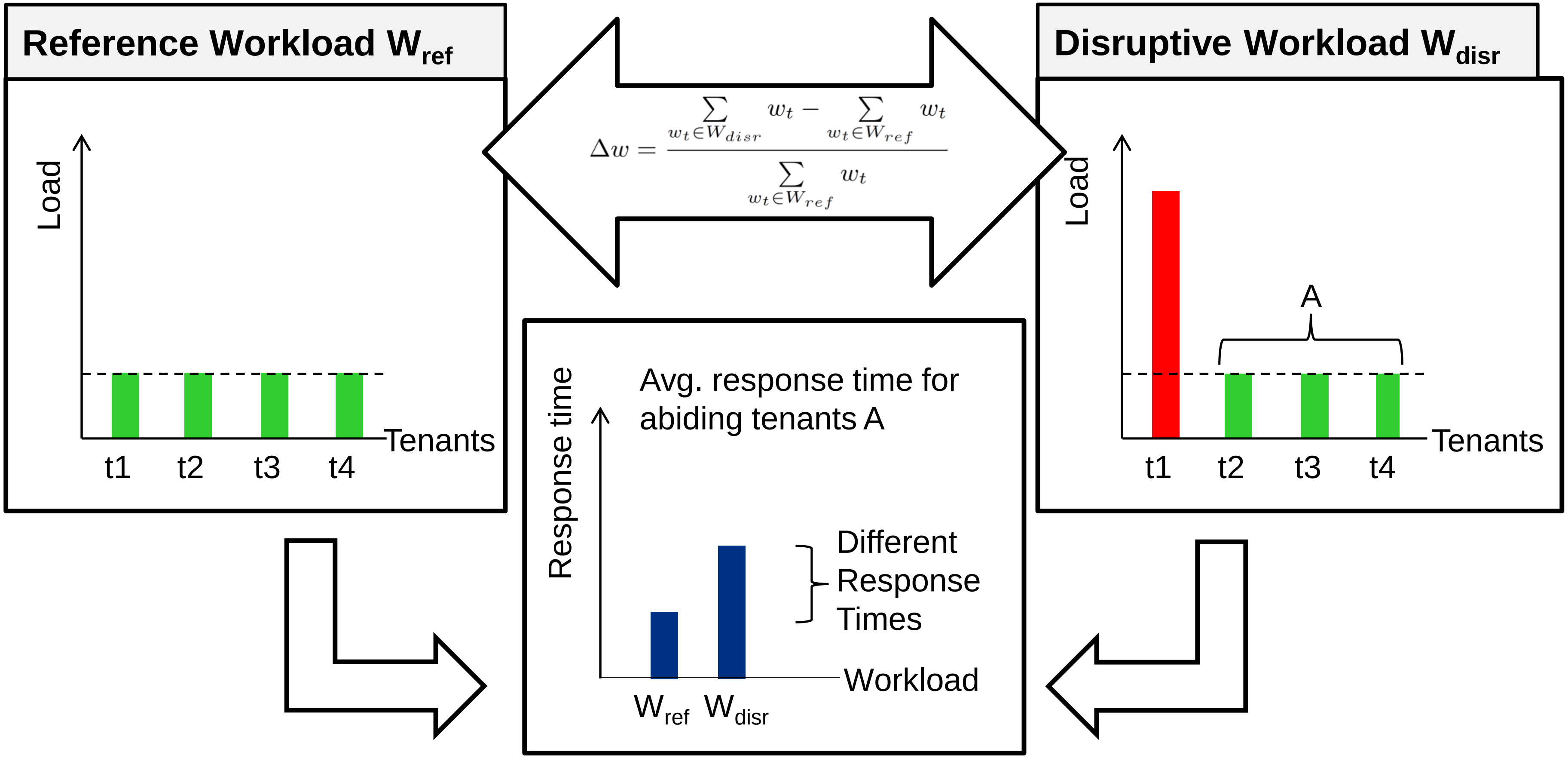
D is a set of **disruptive tenants** exceeding their quotas.

A is a set of **abiding tenants** not exceeding their quotas.



Approach: Quantify impact of increasing workload of the disruptive tenants on the performance of the abiding ones.

Metrics Based on QoS Impact



$$\Delta w = \frac{\sum\limits_{w_t \in W_{disr}} w_t - \sum\limits_{w_t \in W_{ref}} w_t}{\sum\limits_{w_t \in W_{ref}} w_t}$$

$$\Delta z_A = \frac{\sum\limits_{t \in A} [z_t(W_{disr}) - z_t(W_{ref})]}{\sum\limits_{t \in A} z_t(W_{ref})}$$

Example Metric

$$I_{QoS} = \frac{\Delta z_A}{\Delta w}$$

Difference in response time

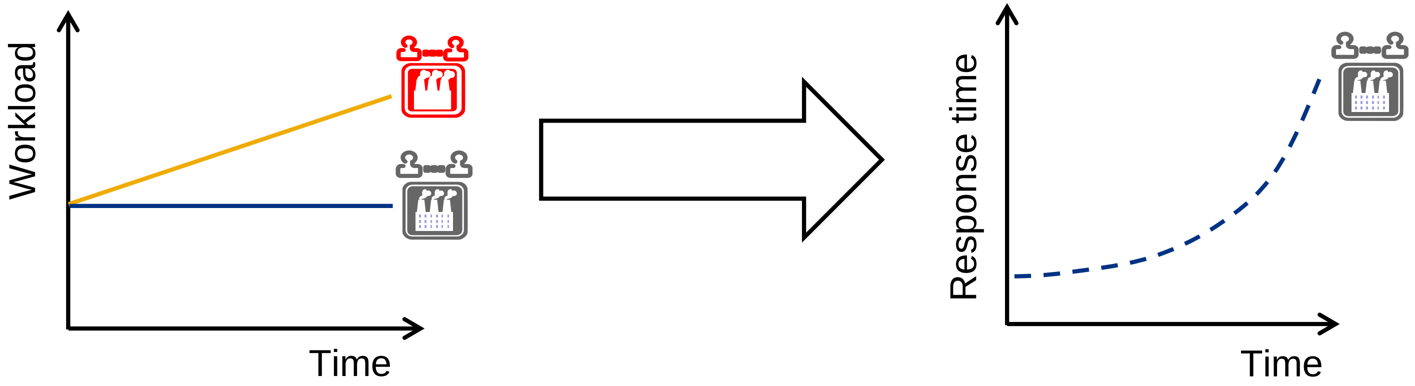
Difference in workload

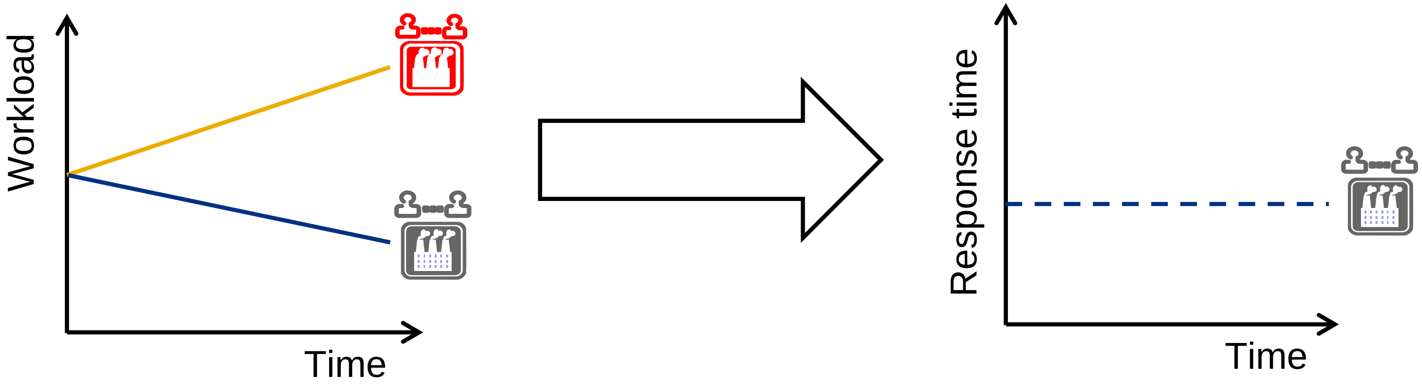
Perfectly Isolated = 0

Non-Isolated = ?

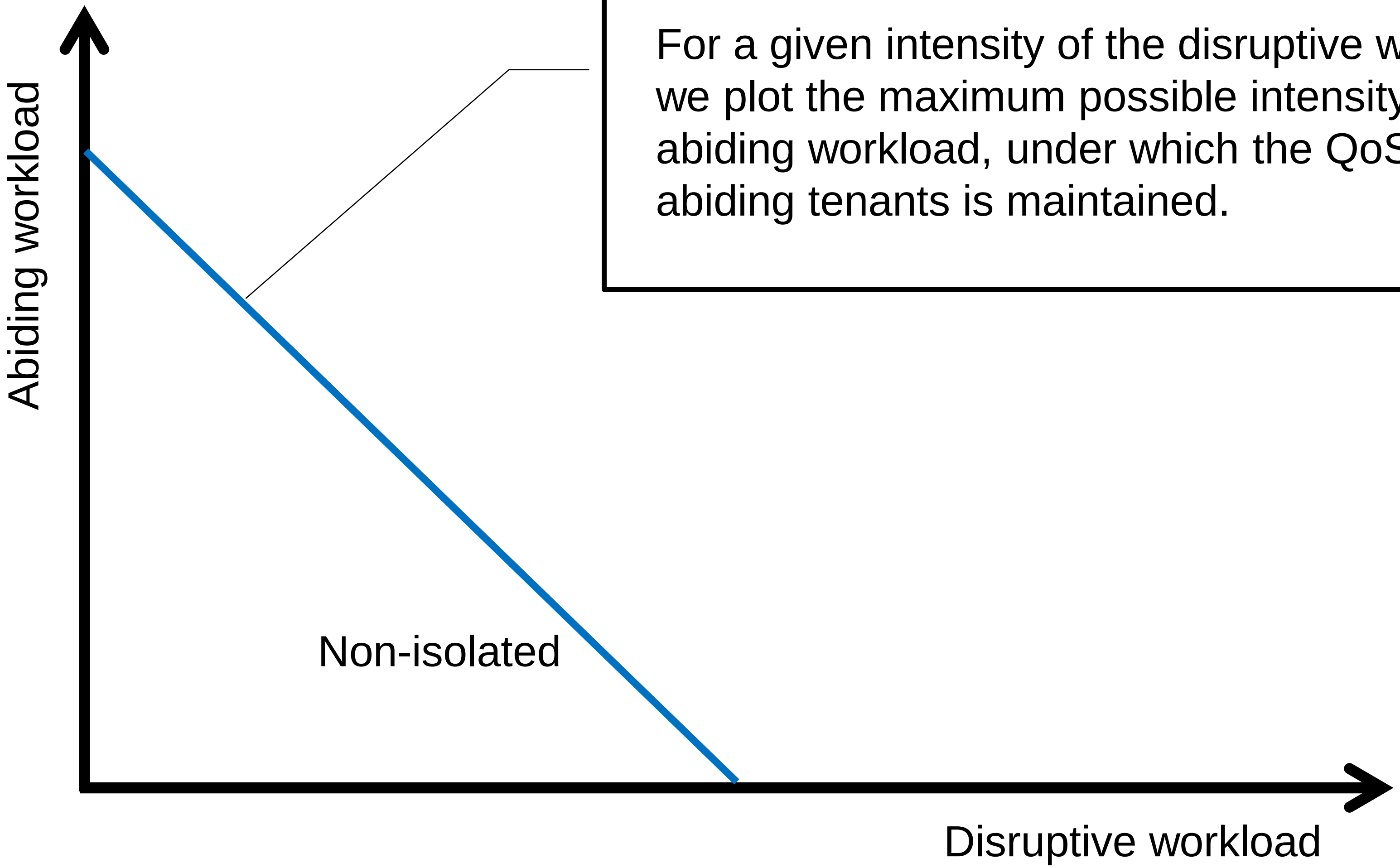
Answers: How strong is a tenant's influence on the others?

Metrics Based on Workload Ratio





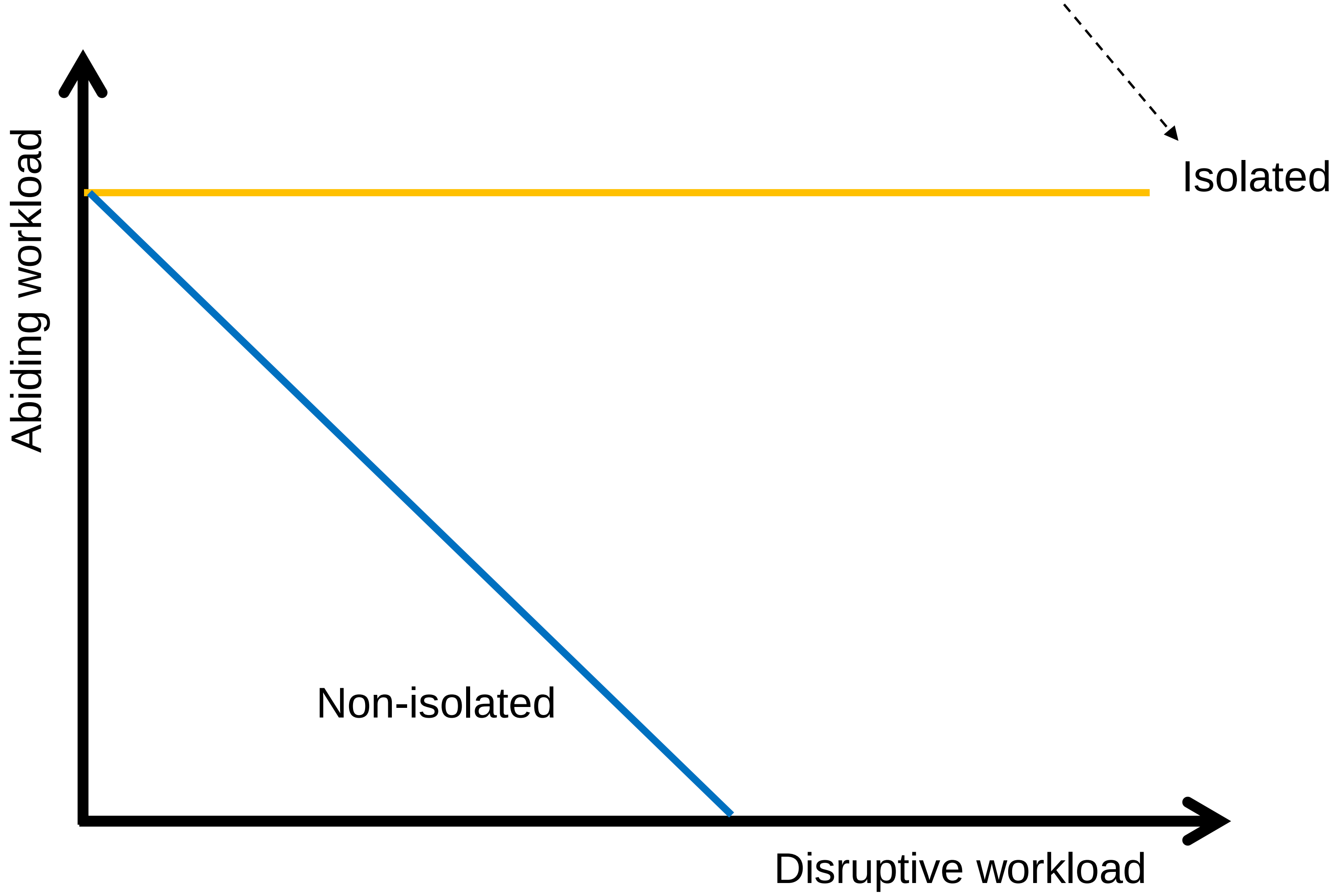
Metrics Based on Workload Ratio



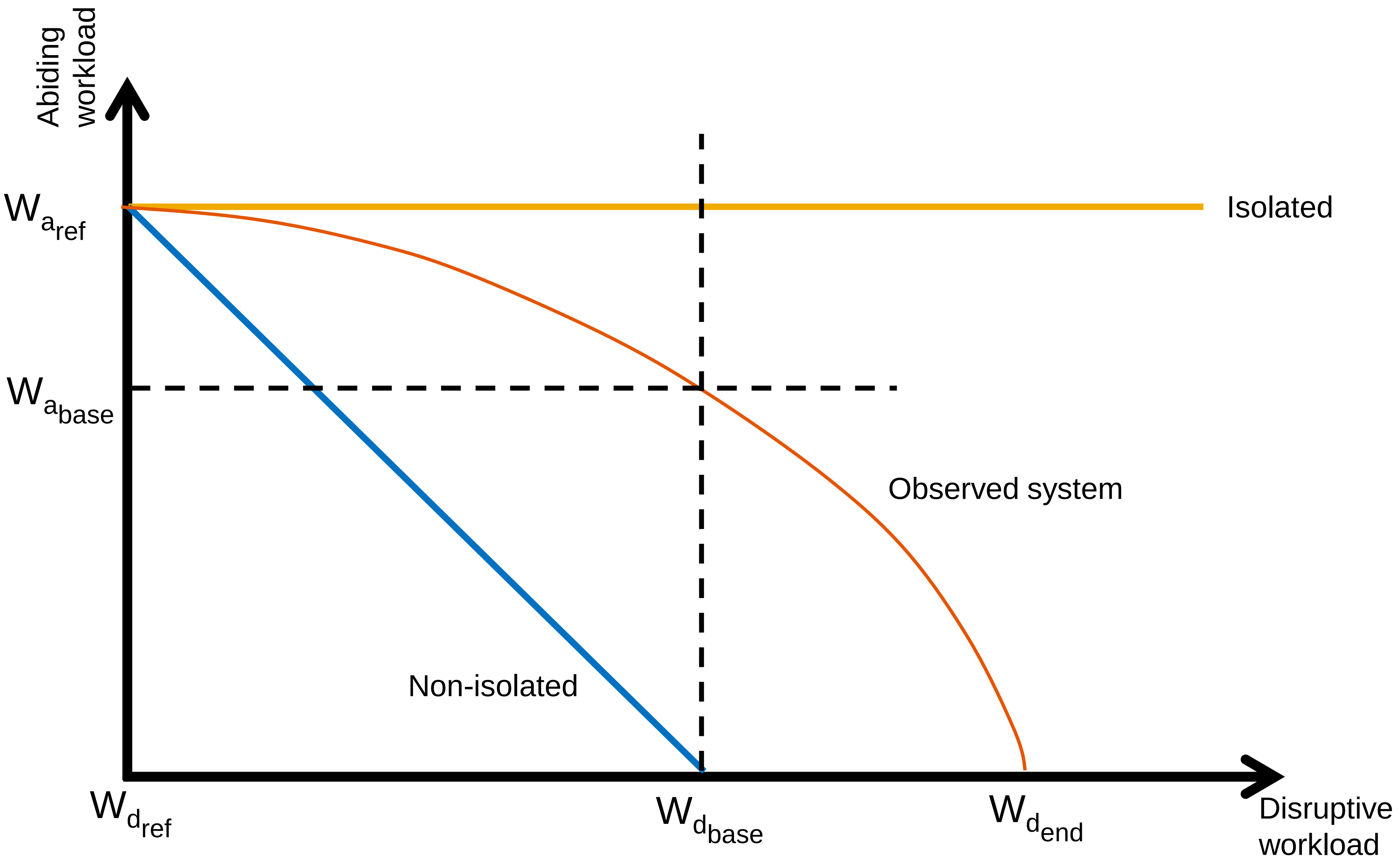
For a given intensity of the disruptive workload, we plot the maximum possible intensity of the abiding workload, under which the QoS of the abiding tenants is maintained.

Metrics Based on Workload Ratio

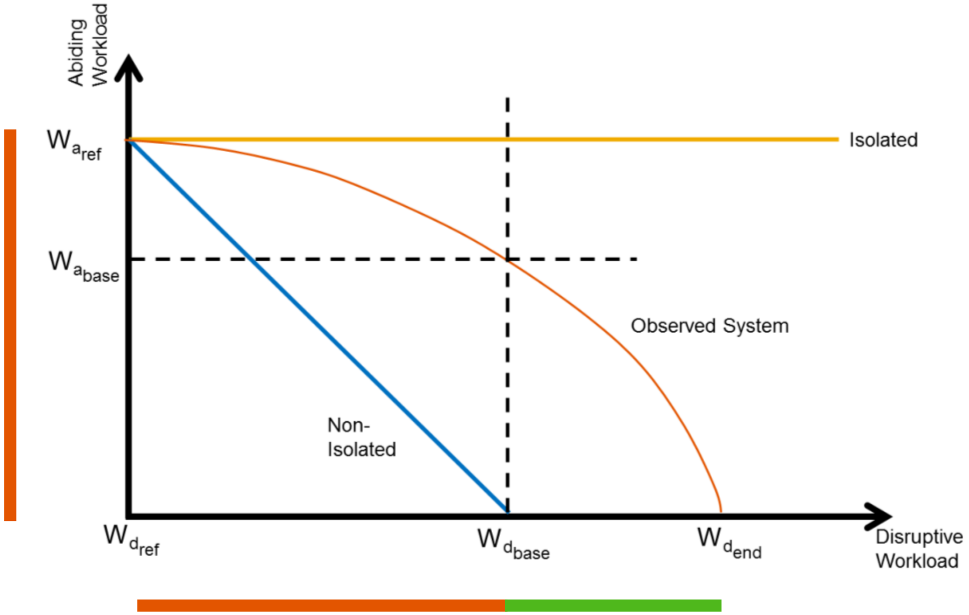
We can maintain the QoS for the abiding tenant without decreasing his workload.



Metrics Based on Workload Ratio



Example Metric: I_{end}



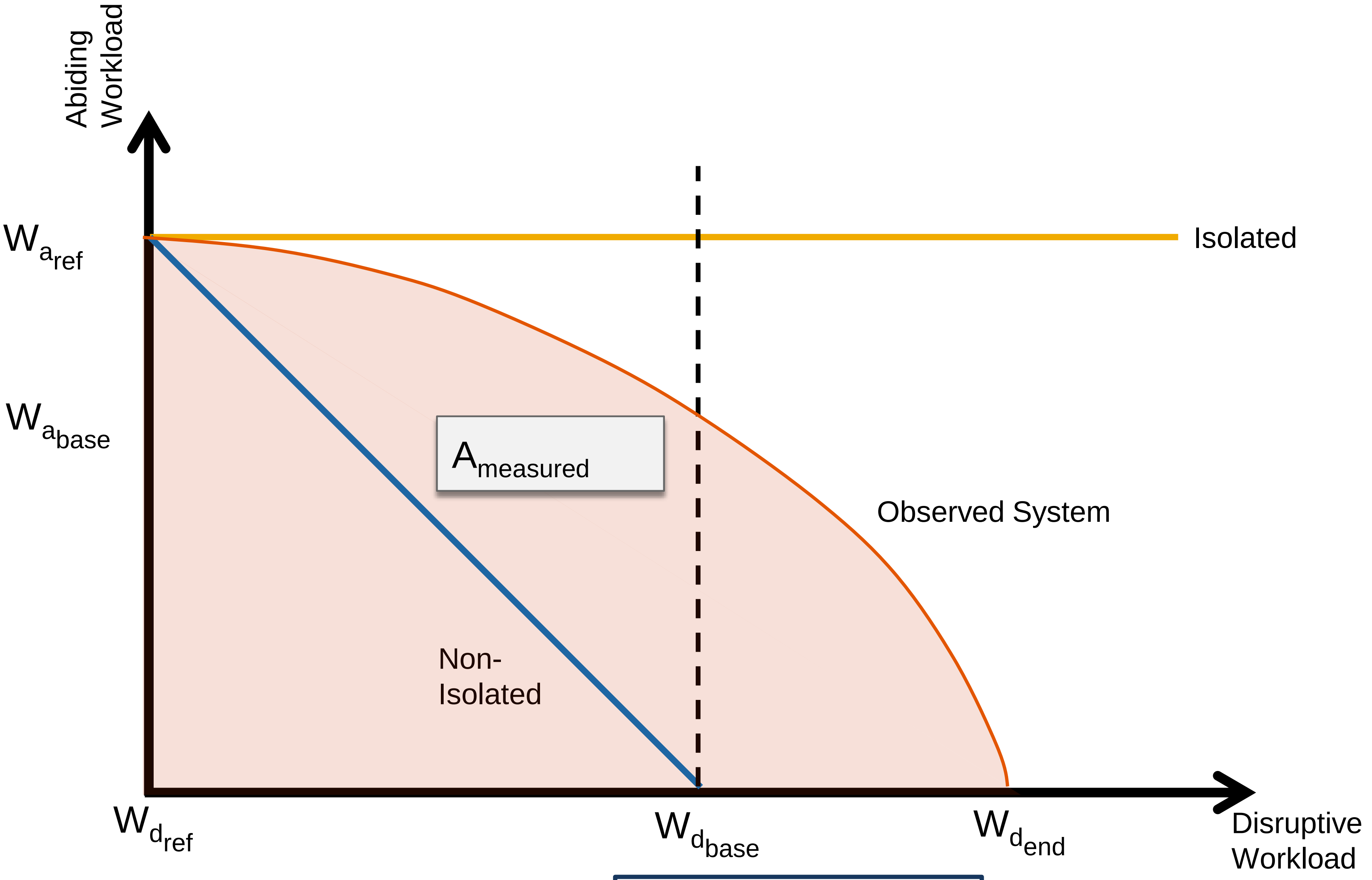
$$I_{end} = \frac{\boxed{W_{d_{end}} - W_{d_{base}}}}{\boxed{W_{a_{ref}}}}$$

Perfectly Isolated = ?

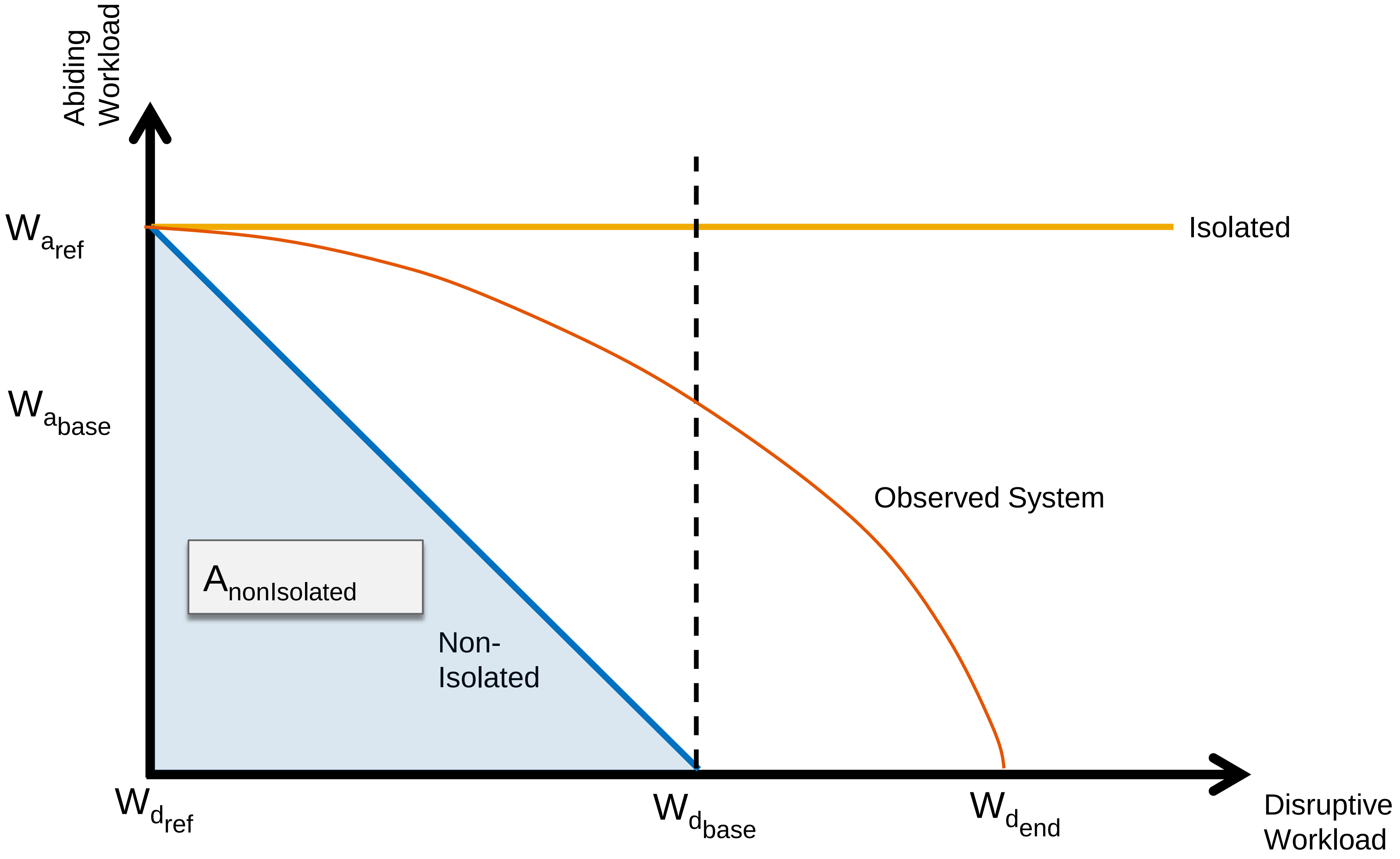
Non-Isolated = 0

Answers: How isolated is the system compared to a non-isolated system?

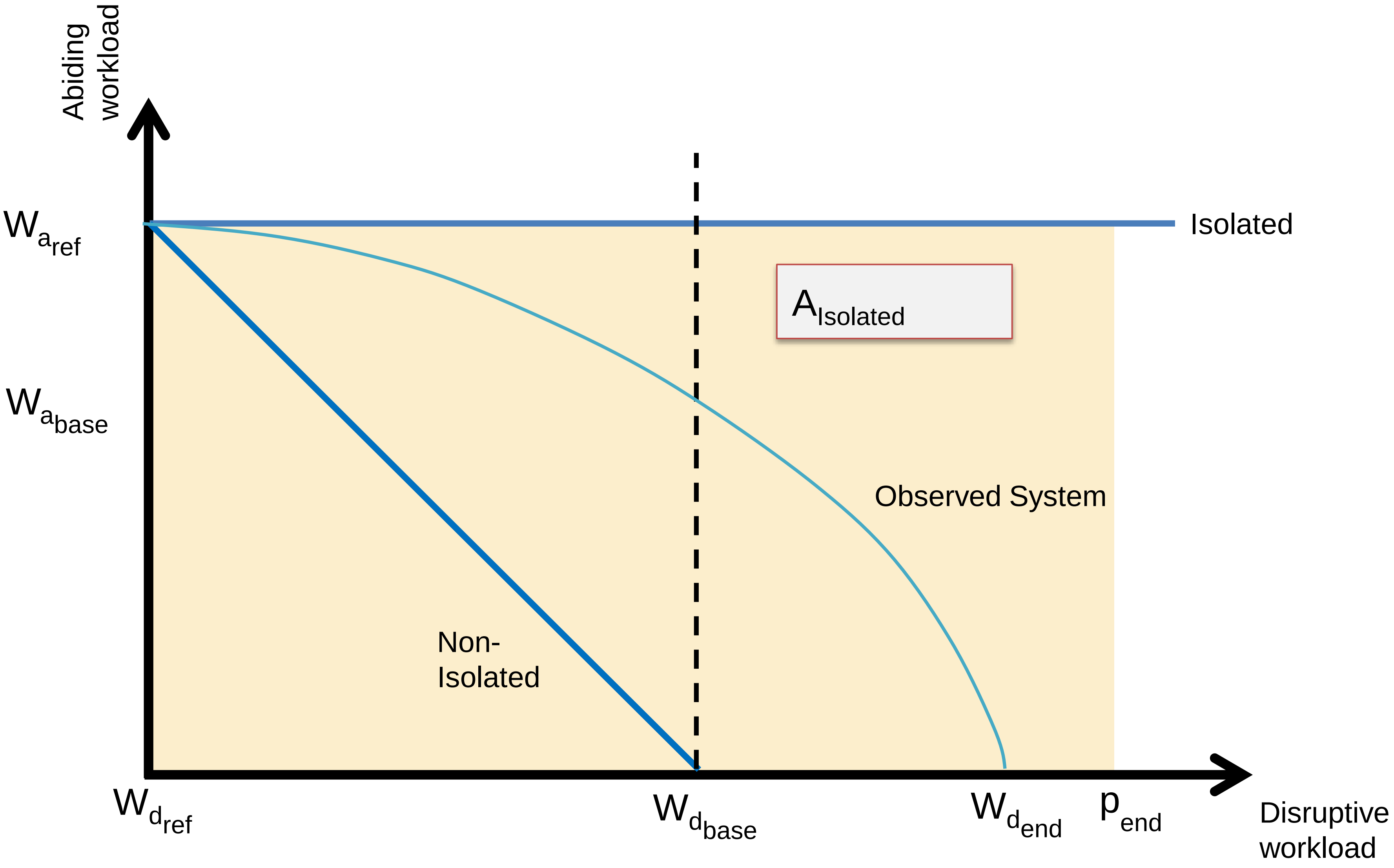
Metrics Based on Workload Ratio Integrals



Metrics Based on Workload Ratio Integrals

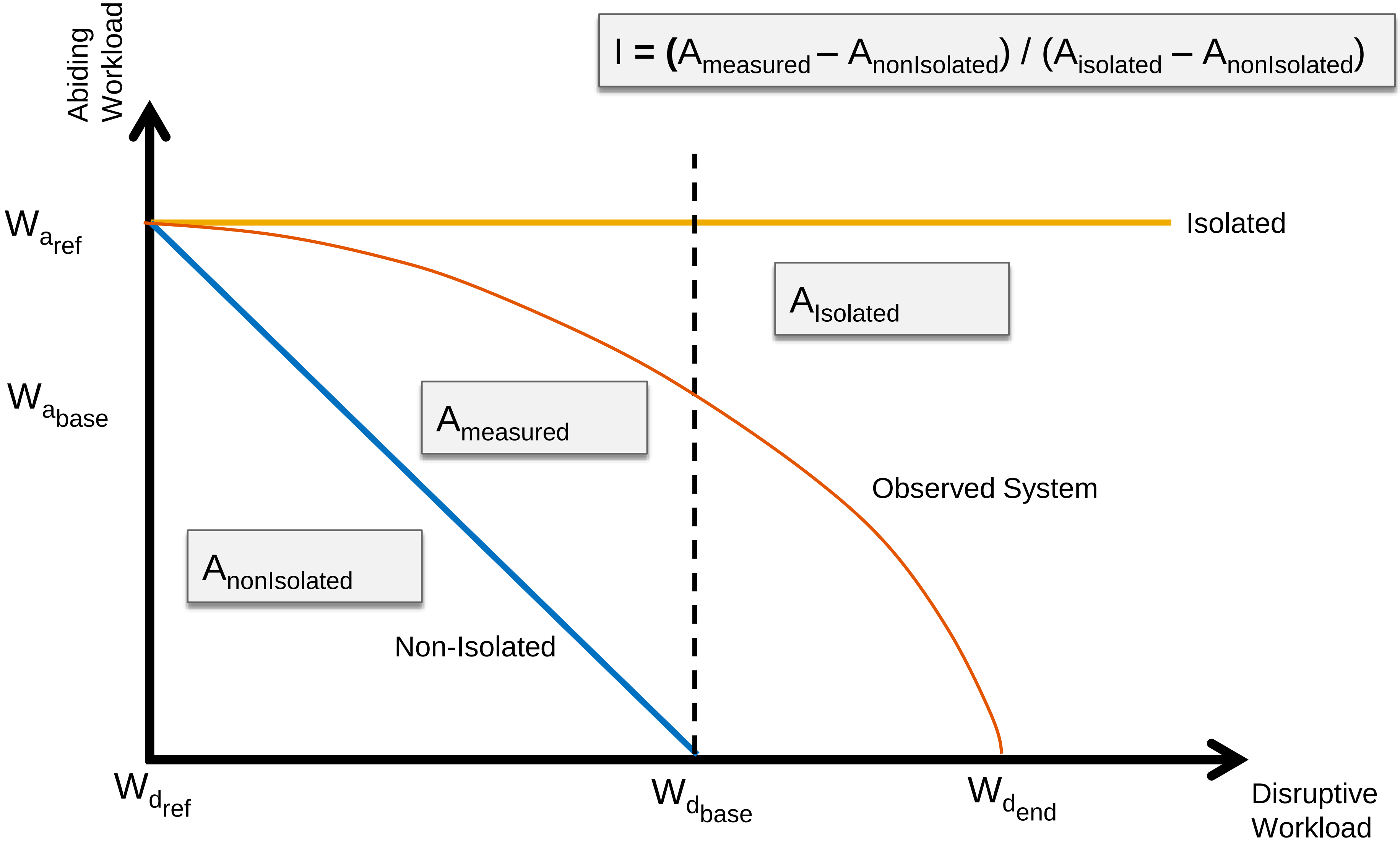


Metrics Based on Workload Ratio Integrals

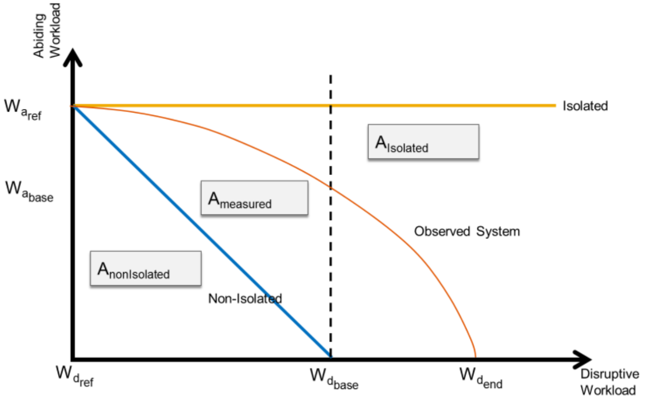


Metrics Based on Workload Ratio Integrals

$I = (A_{measured} - A_{nonIsolated}) / (A_{isolated} - A_{nonIsolated})$



Example Metrics: $I_{intBase}$ and $I_{intFree}$



$$I_{intBase} = \frac{\left(\int_{W_{d_{ref}}}^{W_{d_{base}}} f_m(W_d)\,dW_d\right) - W_{a_{ref}}^2/2}{W_{a_{ref}}^2/2}$$

Areas within W_{dref} and W_{dbase}

$$I_{intFree} = \frac{\left(\int_{W_{d_{ref}}}^{p_{end}} f_m(W_d)\,dW_d\right) - W_{a_{ref}}^2/2}{W_{a_{ref}} \cdot (p_{end} - W_{d_{ref}}) - W_{a_{ref}}^2/2}$$

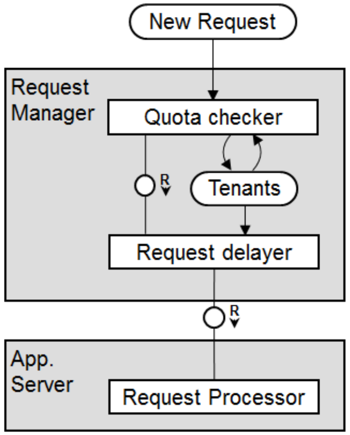
Areas within W_{dref} and predefined bound.

Perfectly Isolated = 1

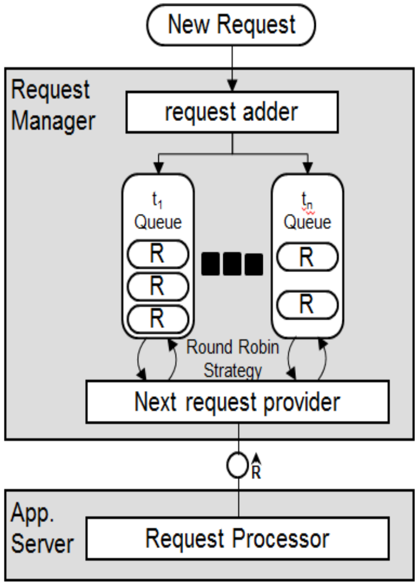
Non-Isolated = 0

Answers: How much potential has the isolation method to improve?

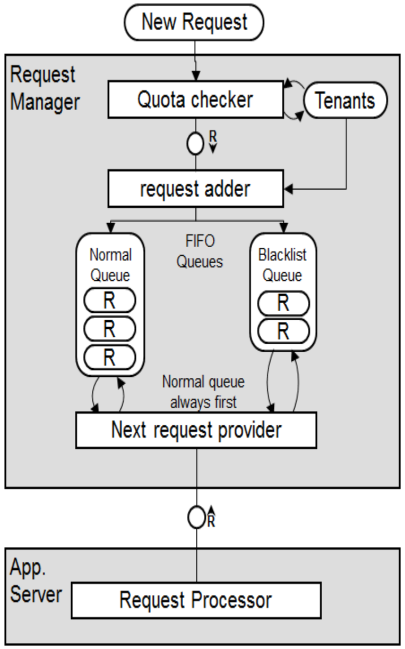
Case Study



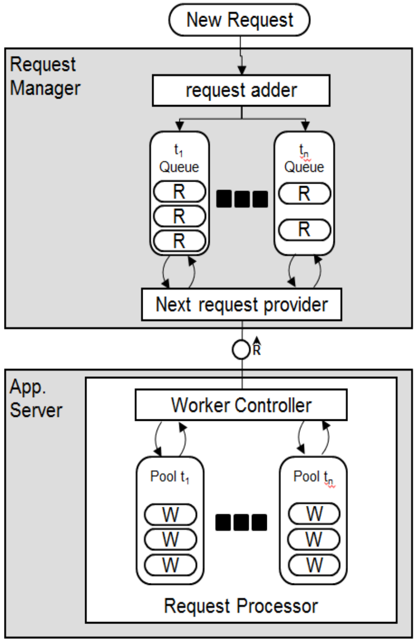
Add Delay



Round Robin



Blacklist



Separate Thread Pools

R. Krebs, C. Momm and S. Kounev. **Metrics and Techniques for Quantifying Performance Isolation in Cloud Environments**. *Elsevier Science of Computer Programming Journal (SciCo)*, Vol. 90, Part B:116-134, 2014, Elsevier B.V. [bib | .pdf]

Three Components of Reliable Benchmarking

Reliable Metrics

- What exactly should be measured and computed?

Representative Workloads

- For which scenarios and under which conditions?

Sound Measurement Methodology

- How should measurements be conducted?

“*To **measure** is to **know**.*” -- Clerk Maxwell, 1831-1879

“*It is much easier to make **measurements** than to **know** exactly what you are measuring.*“ -- J.W.N.Sullivan (1928)

Conclusion

- Use of individual metrics in isolation can provide misleading impression
- To understand the overall system behavior, we need multiple metrics reflecting different aspects
- We also need representative workloads and a sound measurement methodology

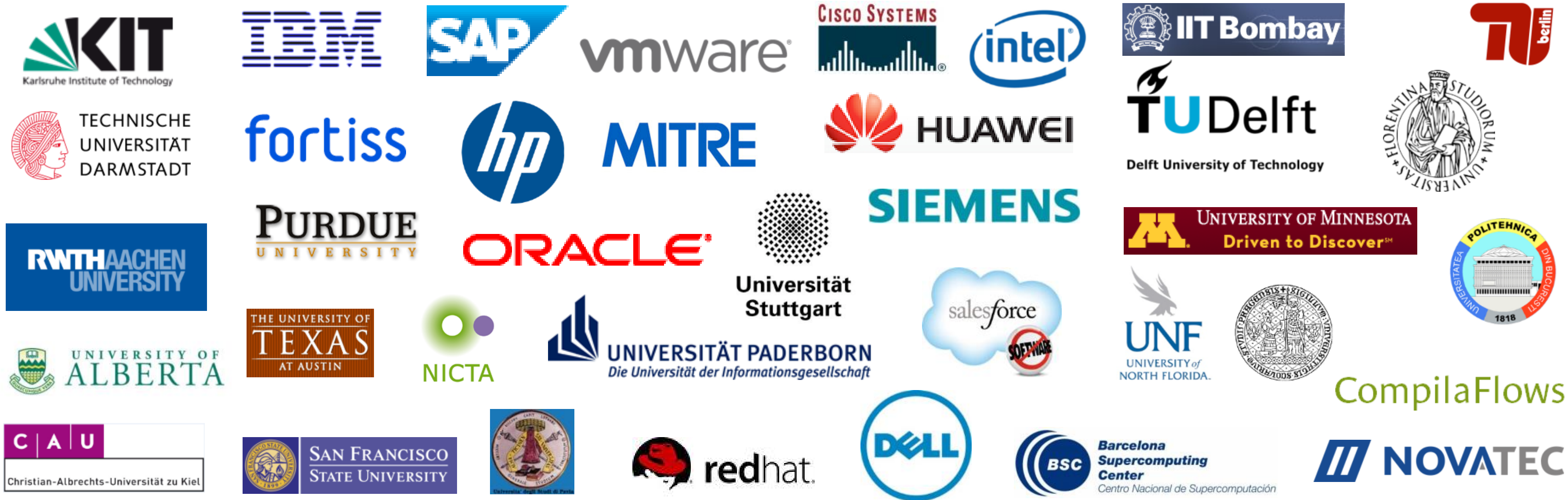
Standard-Performance-Evaluation-Corporation

SPEC Research Group

- **Open-Systems-Group (OSG)**
  - Processor and computer architectures
  - Virtualization platforms
  - Java (JVM, Java EE)
  - Message-based systems
  - Storage systems (SFS)
  - Web-, email- and file server
  - SIP server (VoIP)
  - Cloud computing
- **High-Performance-Group (HPG)**
  - Symmetric multiprocessor systems
  - Workstation clusters
  - Parallel and distributed systems
  - Vector (parallel) supercomputers
- **“Graphics and Workstation Performance Group” (GWPG)**
  - CAD/CAM, visualization
  - OpenGL

SPEC Research Group (RG)

- Founded in March 2011
  - Transfer of knowledge btw. academia and industry
- Activities
  - Methods and techniques for experimental system analysis
  - Standard metrics and measurement methodologies
  - Benchmarking and certification
  - Evaluation of academic research results
- Member organizations (Feb 2014)


KIT
Karlsruhe Institute of Technology
IBM
SAP
vmware
Cisco Systems
intel
IIT Bombay
TU berlin
TECHNISCHE UNIVERSITÄT DARMSTADT
fortiss
hp
MITRE
HUAWEI
TU Delft
Delft University of Technology
FLORENTINA STUDIORUM UNIVERSITAS
RWTH AACHEN UNIVERSITY
PURDUE UNIVERSITY
ORACLE
Universität Stuttgart
SIEMENS
UNIVERSITY OF MINNESOTA
Driven to Discover
UNIVERSITATEA POLITEHNICA DIN BUCUREŞTI 1818
UNIVERSITY OF ALBERTA
THE UNIVERSITY OF TEXAS AT AUSTIN
NICTA
UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft
salesforce
SOFTWARE
UNF
UNIVERSITY of NORTH FLORIDA
CompilaFlows
C A U
Christian-Albrechts-Universität zu Kiel
SAN FRANCISCO STATE UNIVERSITY
Universita' degli Studi di Pavia
redhat
DELL
BSC
Barcelona Supercomputing Center
Centro Nacional de Supercomputación
NOVATEC

Thank You!

skounev@acm.org

http://se.informatik.uni-wuerzburg.de