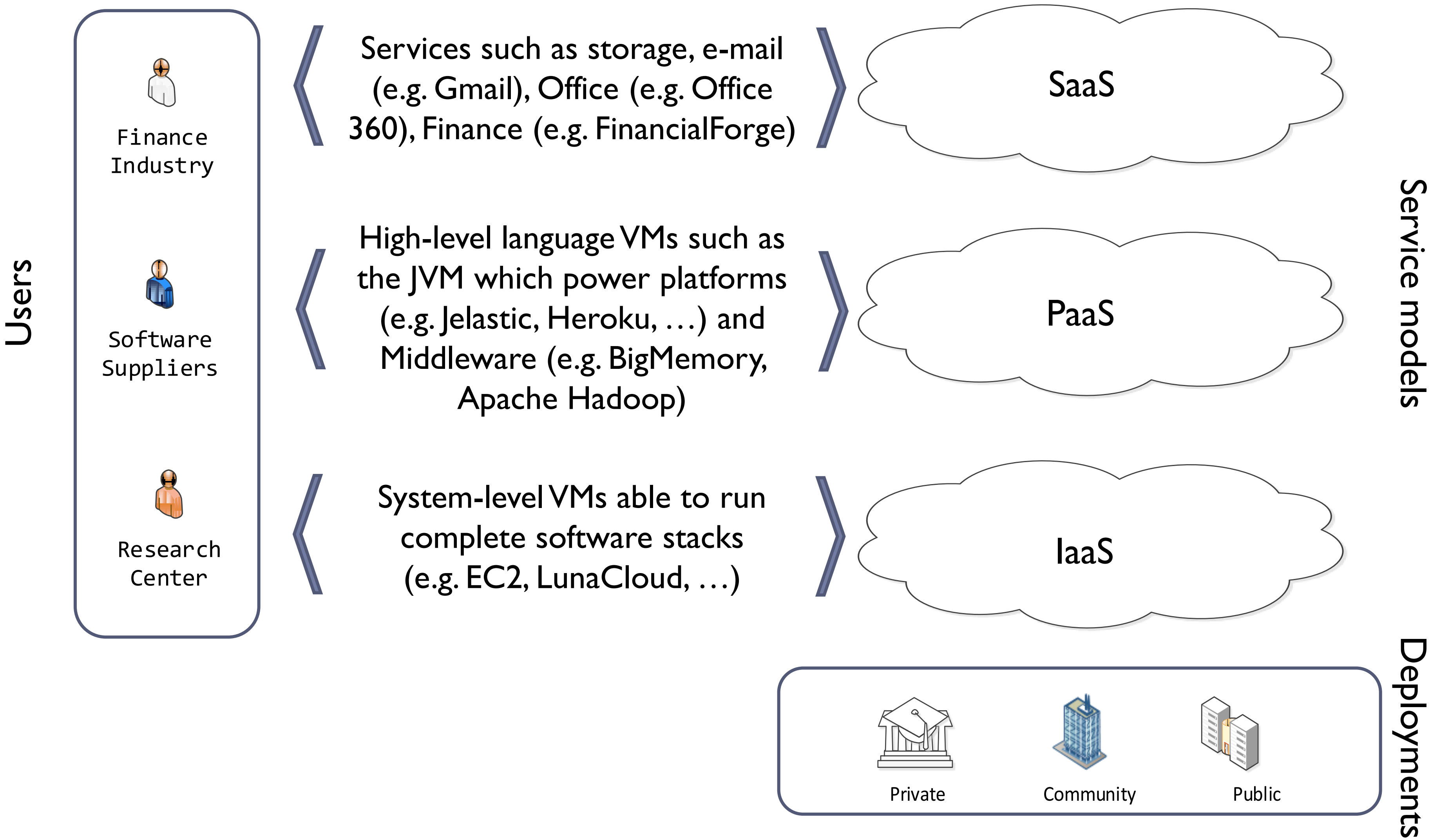
Keynote on ESaaSA – CLOSER 2015

Economics-inspired Resource and Energy Management for Cloud Environments

Luís Veiga
INESC-ID Lisboa
Instituto Superior Técnico
Universidade de Lisboa

May 2015

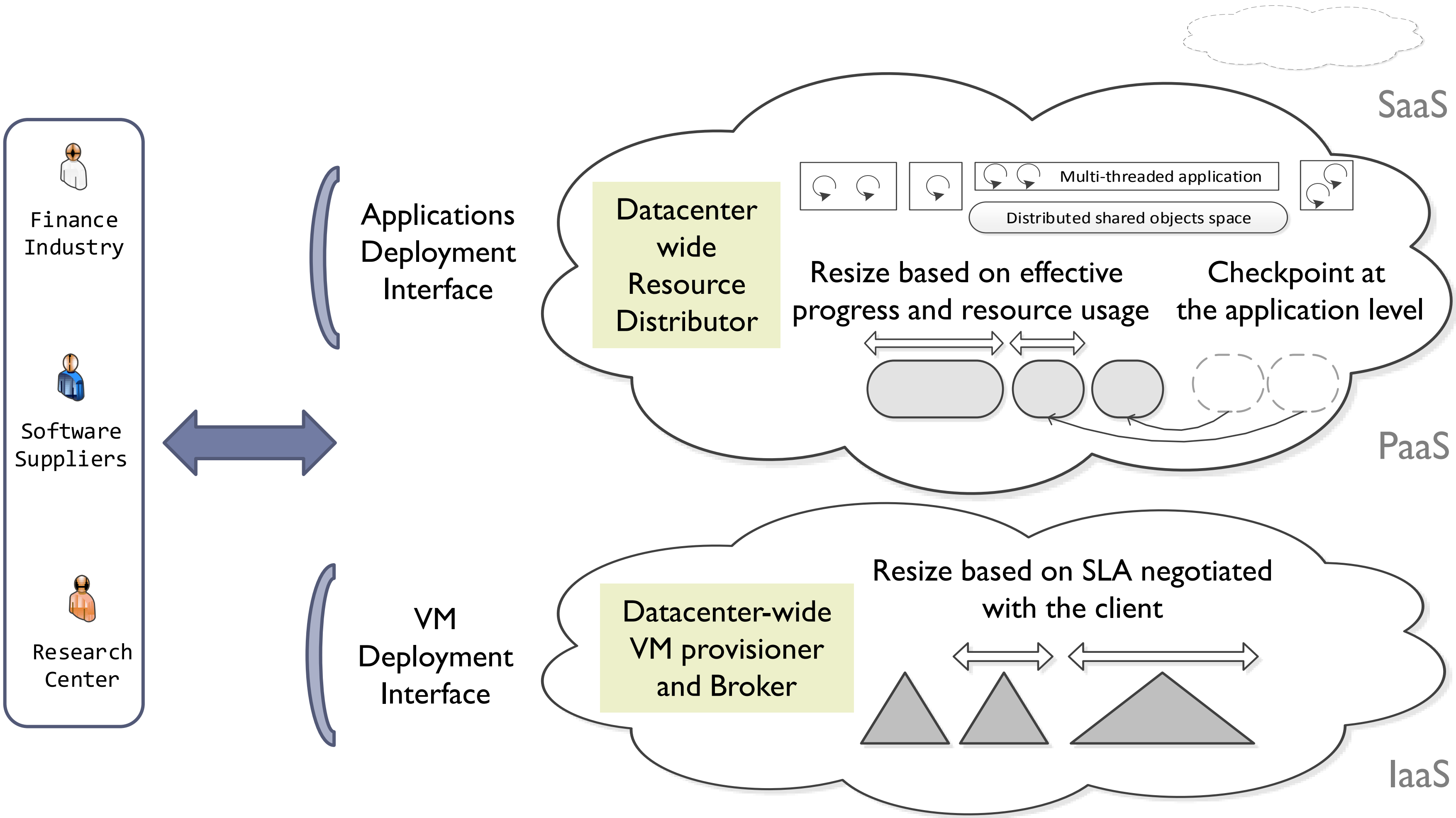
A *day* in the Clouds



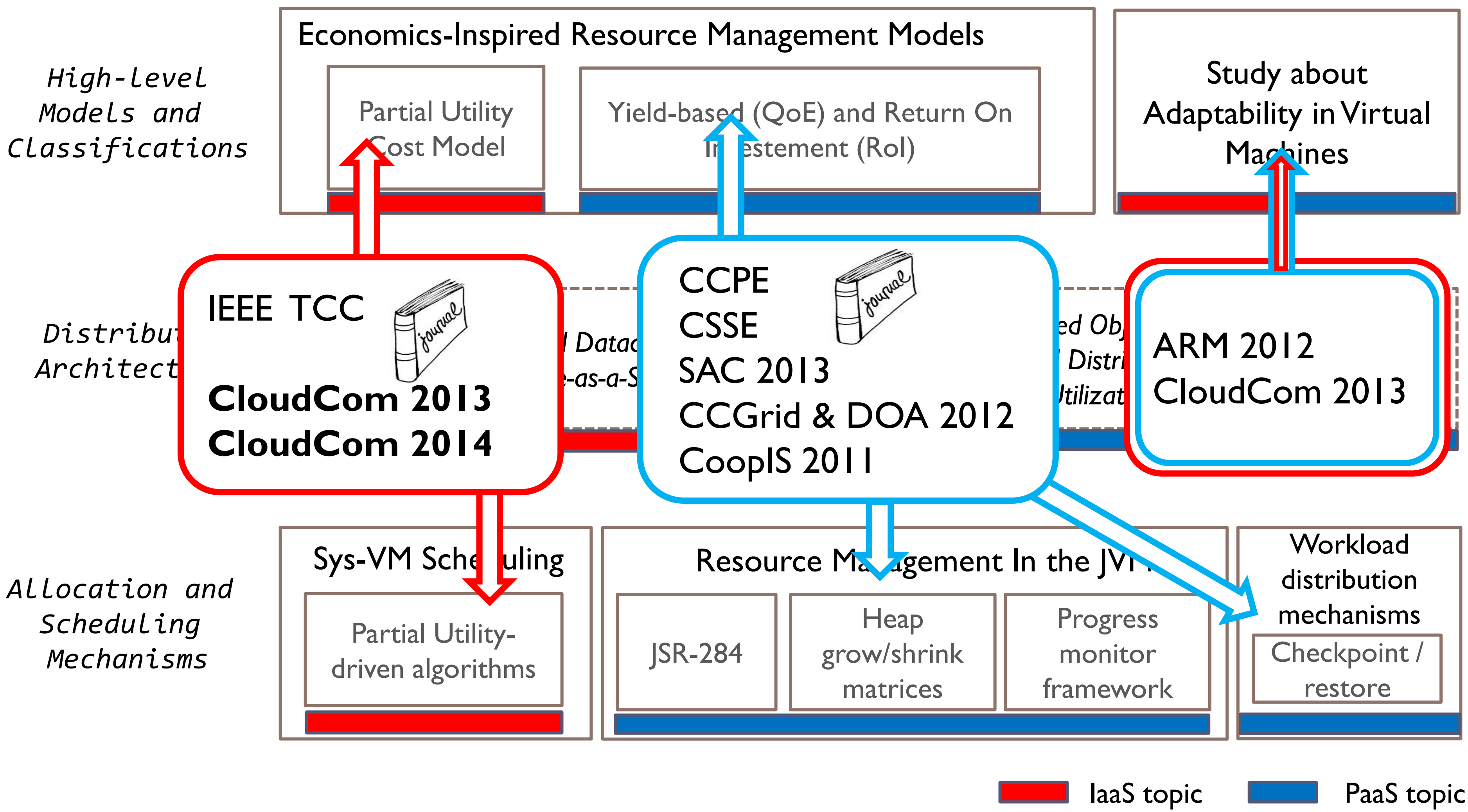
Main challenges

- In general…
  - Providers want to maximize *clients' satisfaction* while minimizing *operational expenditure*
  - But, some defend the infant cloud market is an *oligopoly* [1] and not fully passing the benefits to the client
- PaaS
  - Large-scale simulations, e-Science applications, increasingly depend on manage language runtimes (e.g. JVM, CLR)
  - Resource allocation tailored to the applications, taking into account the effective progress of the workload
- IaaS
  - In public, but mostly in community an private clouds, all-or-nothing resource allocation is not flexible enough
  - A multi-level SLA agreement could foster competition and enlarge the market
- Energy and environmental footprint become prime concerns

A *glimpse* into recent work



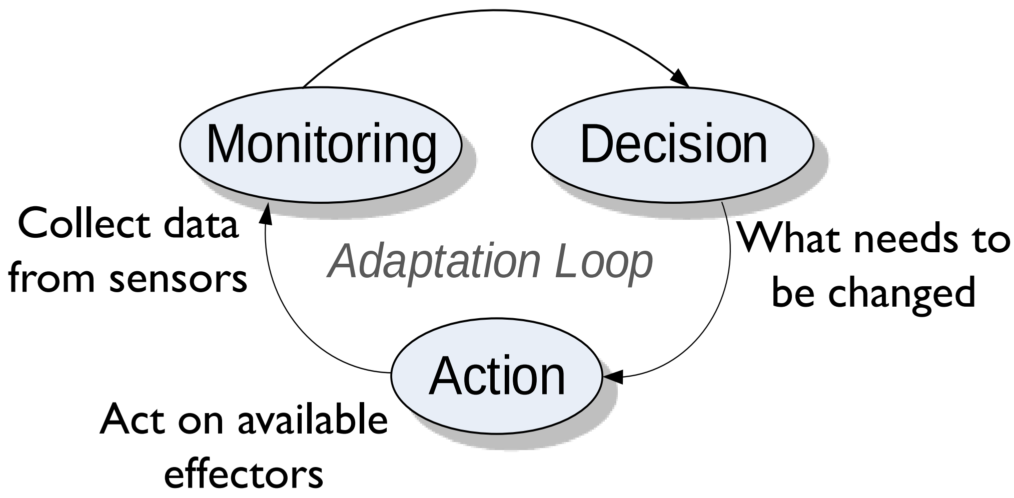
Layered view of the researched topics



Outline

- Introduction
- A study about «adaptability in virtual machines»
- PaaS
  - Models, Mechanisms, Evaluation
- IaaS
  - Models, Mechanisms, Evaluation
- Energy and Community Clouds
  - Models, Mechanisms, Evaluation
- Publications, Conclusions, Ongoing and Future Work

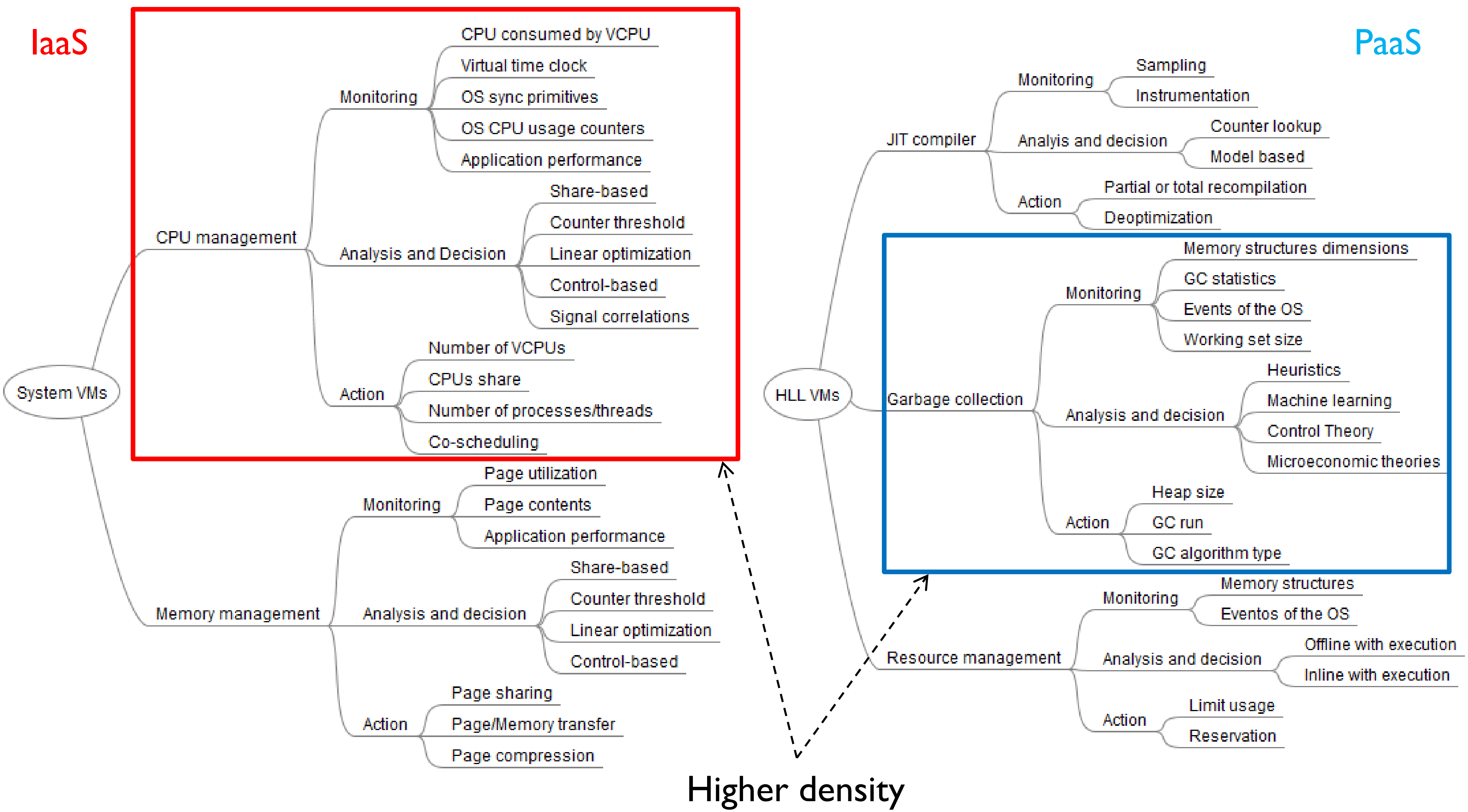
Adaptability in virtual machines



- How to analyze?
- **R**esponsiveness
  - *how fast can the system adapt?*
- **C**omprehensiveness
  - *which is the breadth and scope of the adaptation process?*
- **I**ntricateness
  - *which is the depth/complexity of the adaption process?*

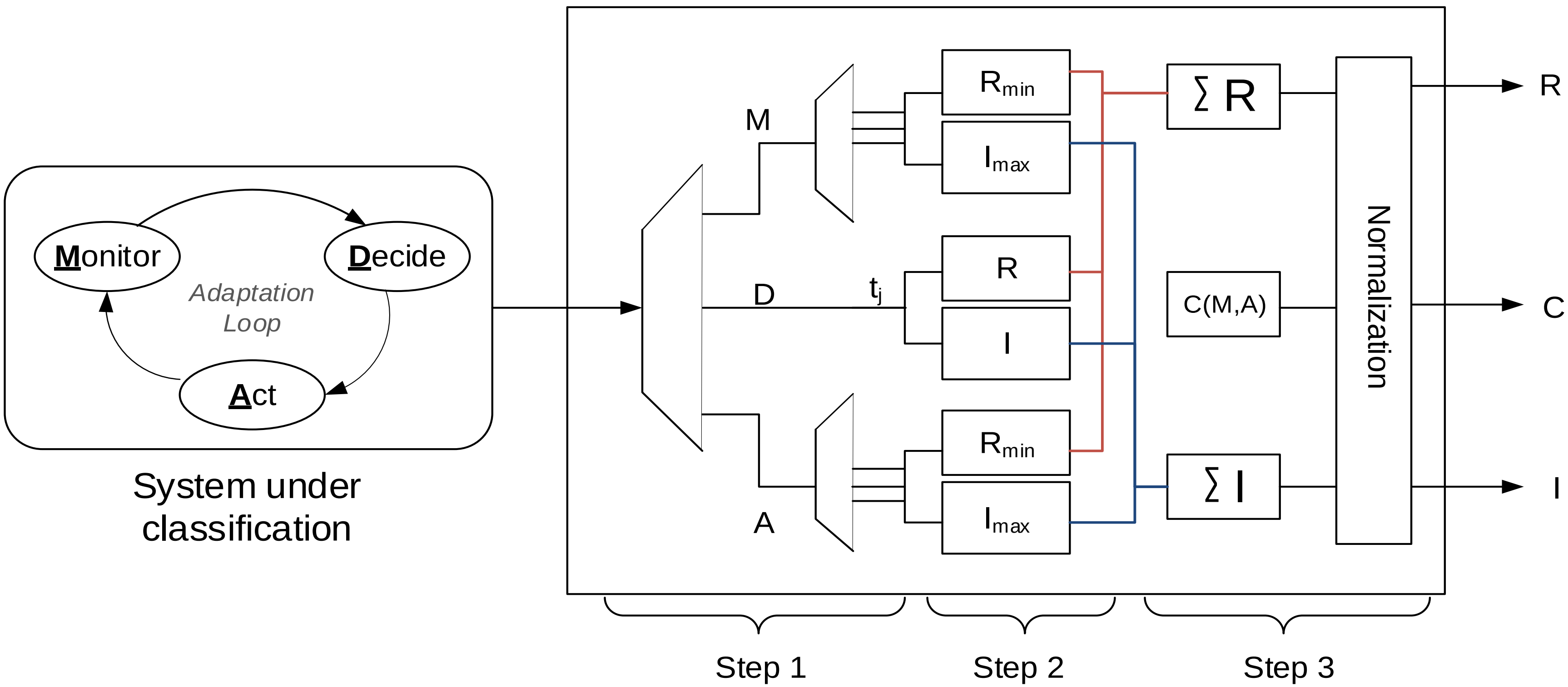
- *Conjecture:* A given adaptation technique aiming at achieving improvements on two of these aspects (**R**esponsiveness, **C**omprehensiveness, **I**ntricateness) can only do so at the cost of the remaining one.
  - *Distributed system in general: **C**onsistency, **A**vailability and tolerance to **P**artitions [5]*
  - *P2P: **H**igh availability, **S**calability and support for **D**ynamic Populations [6]*

Adaptability techniques



JS, LV @ ARM workshop 2012 (IaaS)
JS, LV @ IEEE CloudCom 2013 (PaaS)

RCI framework internals

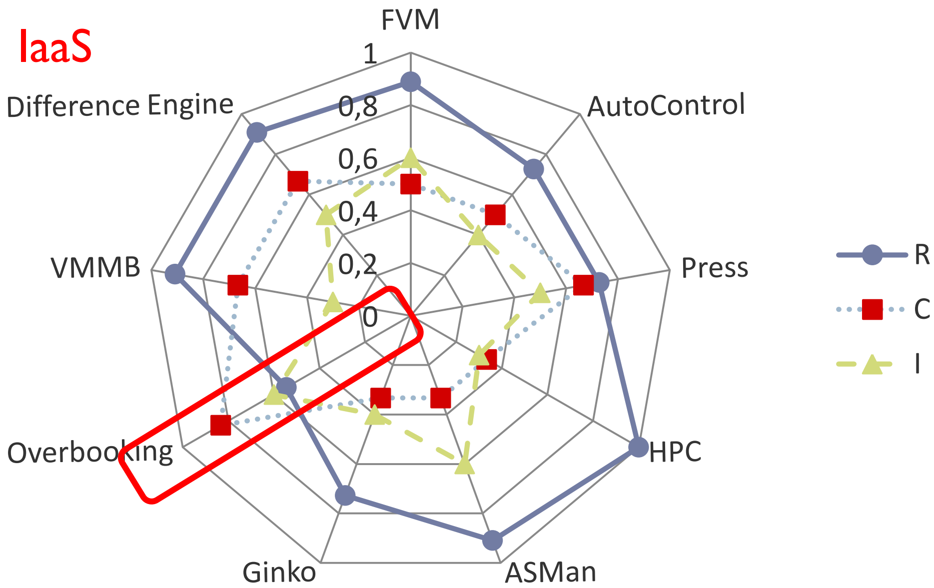


Decomposition of techniques

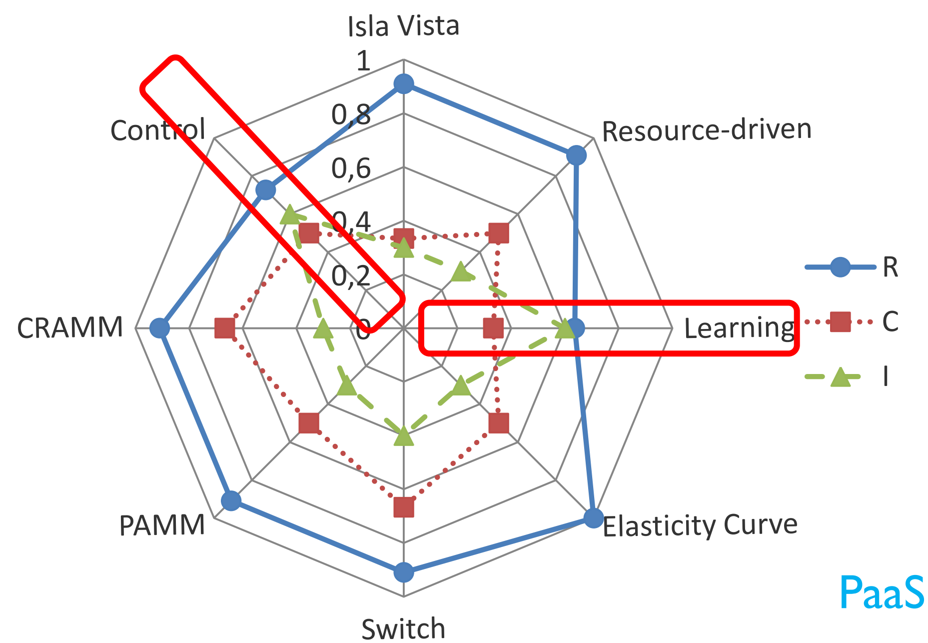
Mapping to a qualitative value

Aggregation and normalization

RCI conjecture in practice



- Currently, 17 influential systems were analyzed in depth, assessed and classified.
  - New systems and techniques can be added without changing the classification framework



- In both types of VMs *R* is dominant
- *Overbooking* exchanges *R* by *C*
- In *Control* and *Learning*, a higher *I* lead to a reduced *R*

Outline

PaaS-level motivation and goals

- How to influence an application behavior, ***effectively*** (wide range and impact), ***efficiently*** (low overhead) and ***flexibly*** (with no or little intrusive coding)?
- *Line of work*: Extend managed runtimes (e.g., Java VMs such as Jikes RVM [3] and OpenJDK [4]) to operate efficiently in multi-tenancy scenarios such as those of cloud computing infrastructures
  - ***Accurately monitor resource usage***
  - ***Monitor application progress***
  - ***Resource management***
  - ***Elasticity and horizontal scaling***

Economic *yield*

- ***yield*** is a return/reward from applying a given allocation strategy (S) to some resource (r)

$$Yield_r(S_a, S_b) = \frac{\mathrm{Savings}_r(S_a, S_b)}{\mathrm{Degradation}(S_a, S_b)}$$

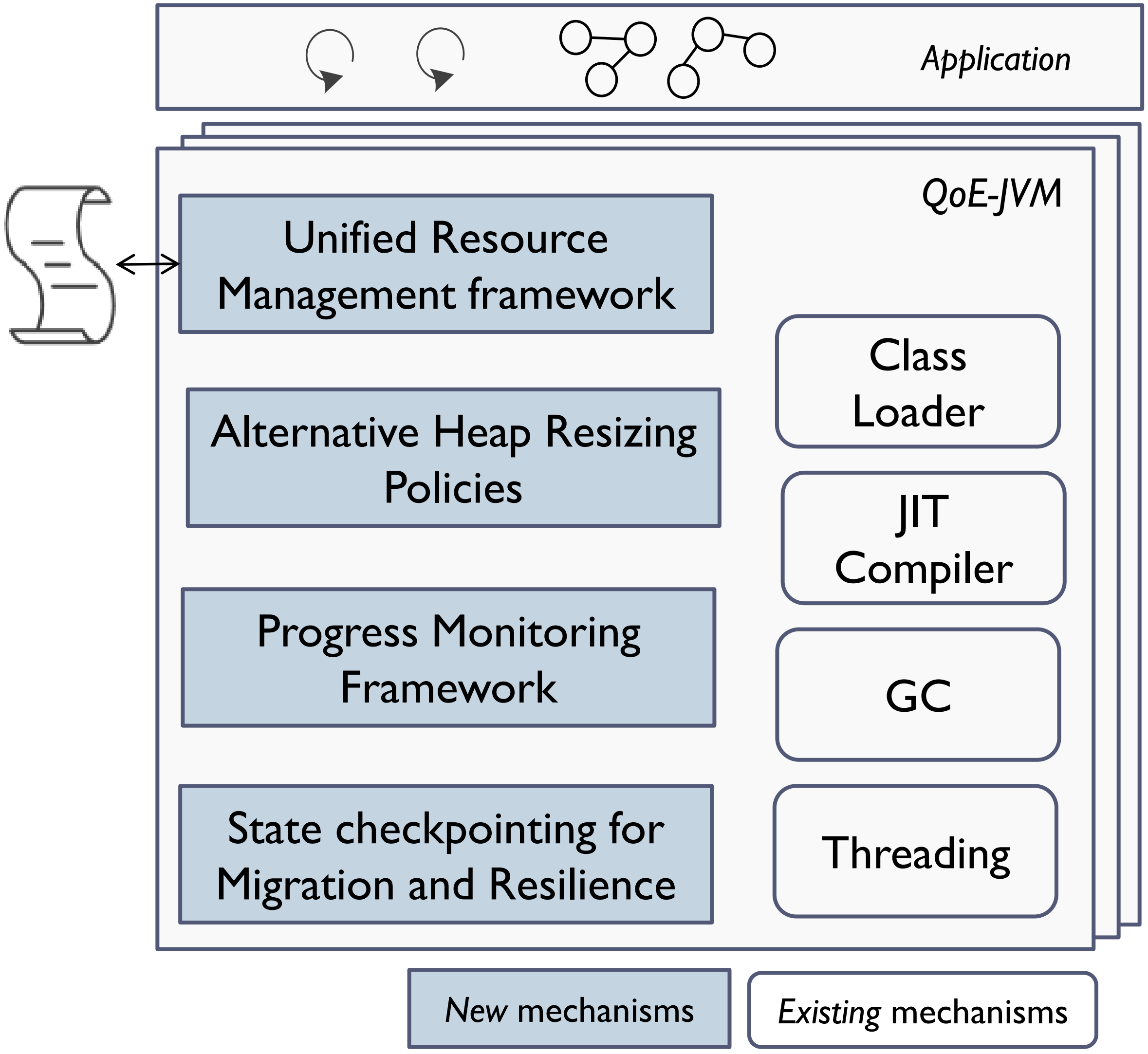
$$\mathrm{Savings}_r(S_a, S_b) = \frac{U_r(S_a) - U_r(S_b)}{U_r(S_a)}$$

- ***Savings*** represents how much of a given resource *(r) is saved* when two management strategies are compared.
- It relates the usage (*U*) of a resource with the old and the new configuration

$$\mathrm{Degradation}(S_a, S_b) = \frac{P(S_b) - P(S_a)}{P(S_a)}$$

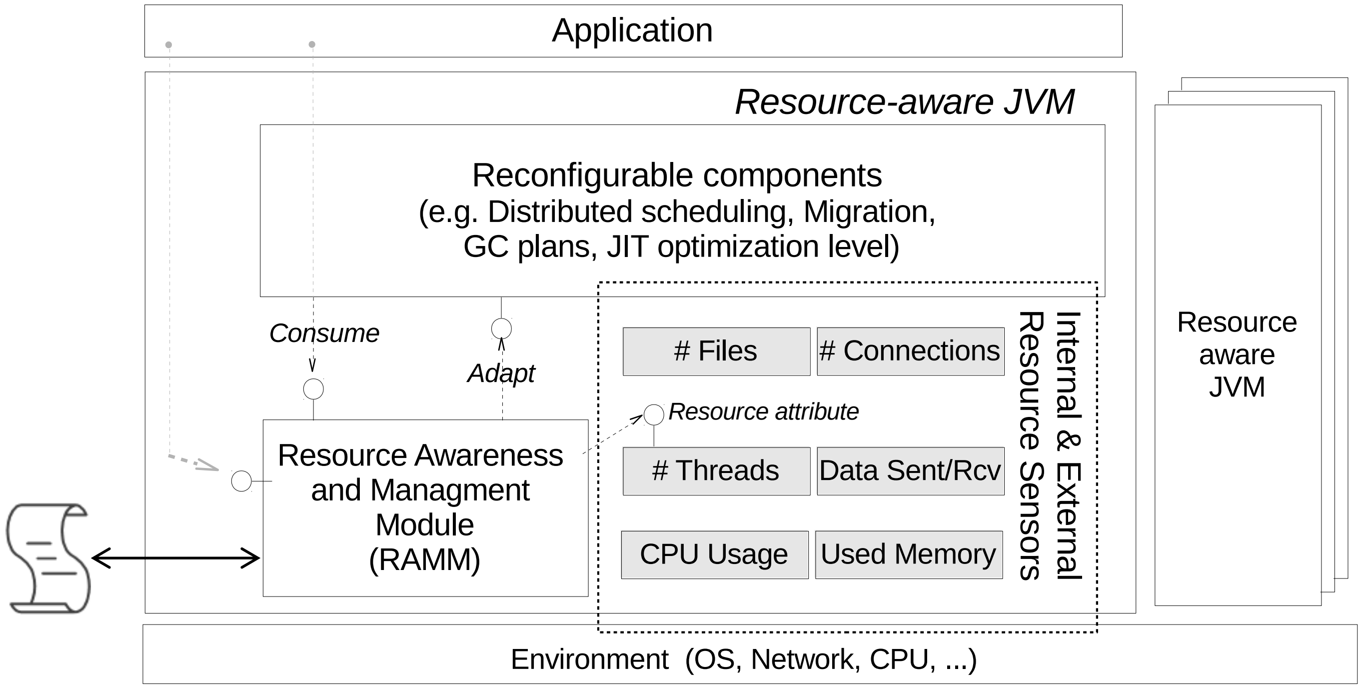
- ***Degradation*** represents the impact of the savings, given a specific performance or progress metric (e.g. execution time).
- It relates the progress (*P*) made with the old and the new configuration

Mechanisms



- Mechanisms incorporated in Jikes RVM, «winner of the *ACM SIGPLAN Software award*, cited for its "high quality and modular design"» in http://en.wikipedia.org/wiki/Jikes_RVM
- Progress monitor supported on Java instrumentation agent infrastructure

Unified Resource Management Framework

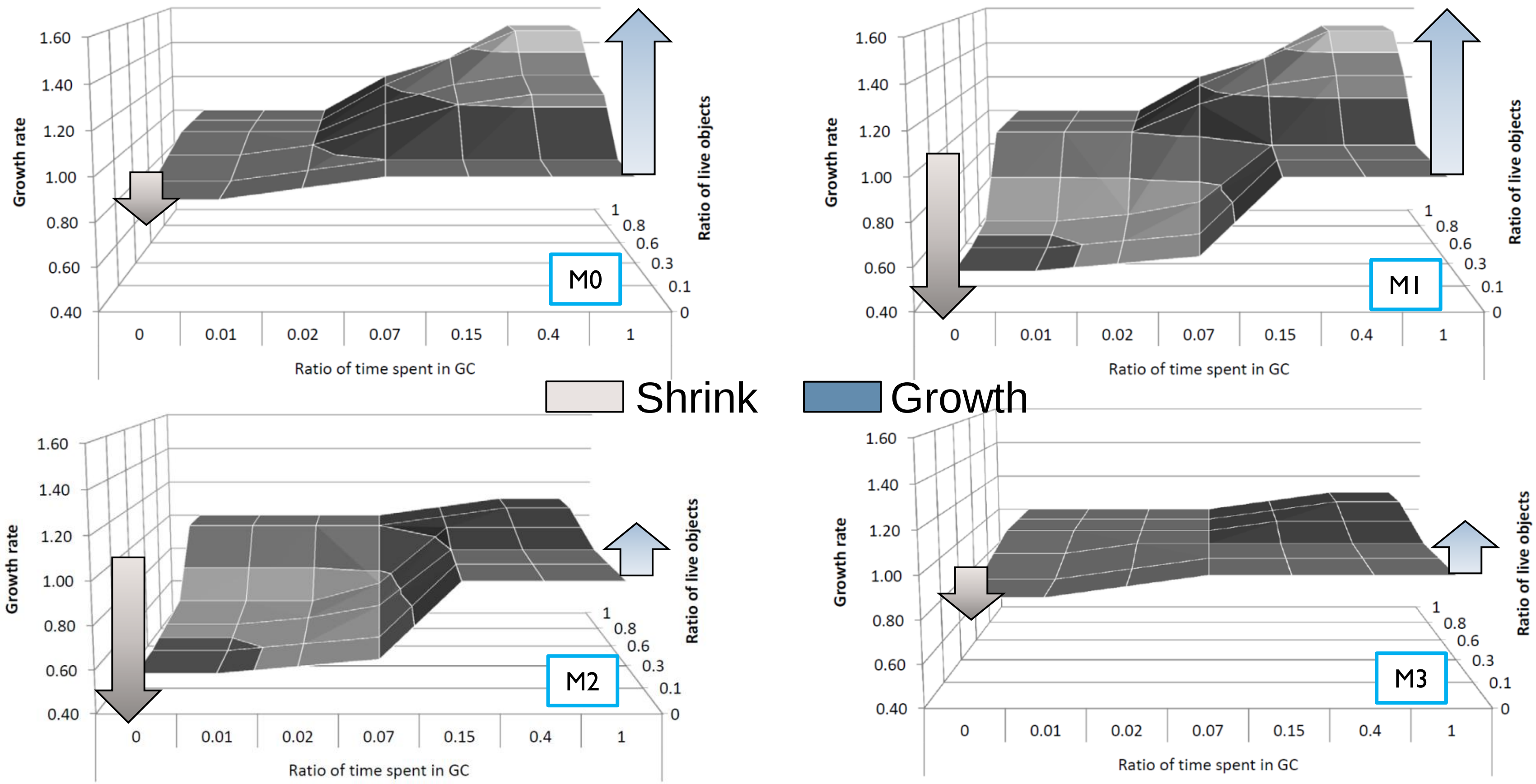


- Extension of Jikes RVM [3], and the GNU classpath, with JSR 284 – The resource management API
- Monitoring and enforcement points include
  - Memory allocation (heap growth rate), CPU usage, Thread creation

JS, JL, LV @ CoopIS 2011, LNCS
JS, LV @ DOA-SVI 2012, LNCS

Heap Policies: Base and alternatives

- GC-Economics in Jikes RVM
  - *heap growth rate driven by wasted CPU on GC*



JS, LV @ CSSE, CRL Publishing, 2013
JS, LV @ DOA-SVI 2012, LNCS

Progress Monitoting framework

```
@Retention(RetentionPolicy.RUNTIME)
@Target({ElementType.METHOD,
         ElementType.FIELD,
         ElementType.PARAMETER})
public @interface Progress {
  double relevance() default 1.0;
}
```

(a)
Definition

Call rates updater

update

```
public class AClass {
  @Progress(relevance=0.
  public void m1() { ... }

  @Progress(relevance=0.2)
  public void m2() { ... }
}
```

update

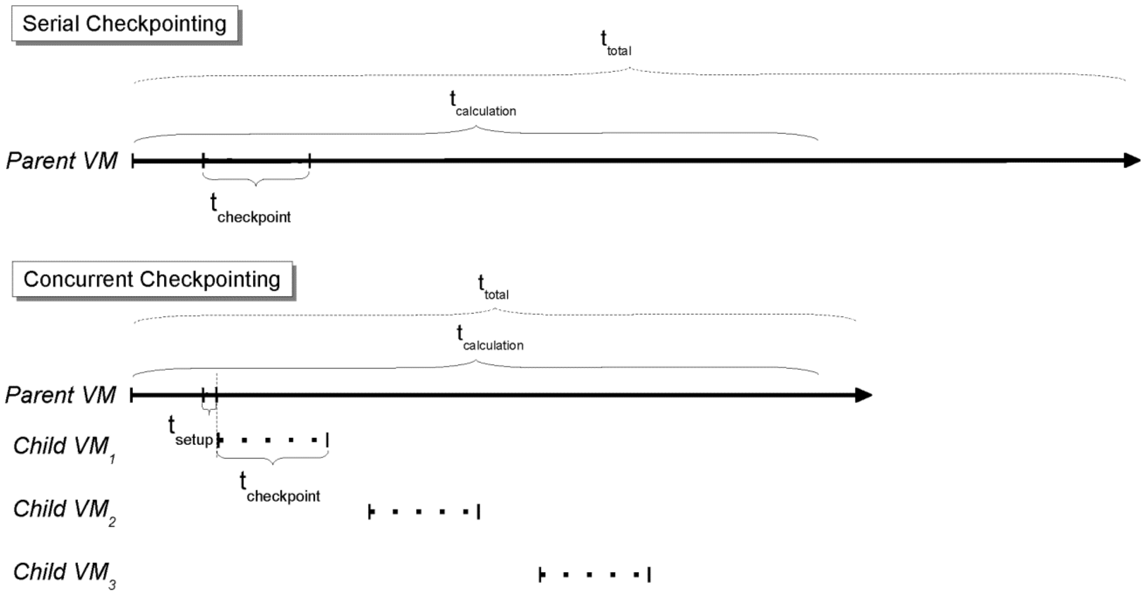
(b)
Usage

Method	MethodStats
m1	*Counters, Call rates*
m2	*Counters, Call rates*
...	

- Annotations are used at load time to insert measurement code (by an instrumentation agent)
- Measurements: overall call rate, window call rate (last *n* ms.)

Checkpointing for application-level migration

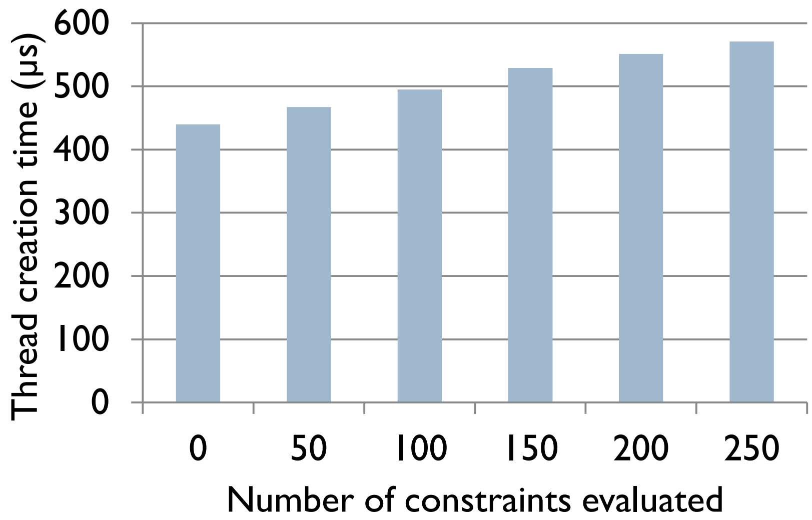
- Serial checkpoint needs to:
  - 1. Stop all running threads, 2. Build method descriptors, 3. Save execution state (i.e. stack frames), 4. Save graph of reachable objects
- Concurrent checkpoint makes the two final steps in parallel with the application
- Relies on on-stack-replacement, serialization and fork technologies
- Limitations
  - JNI code that touches heap managed objects

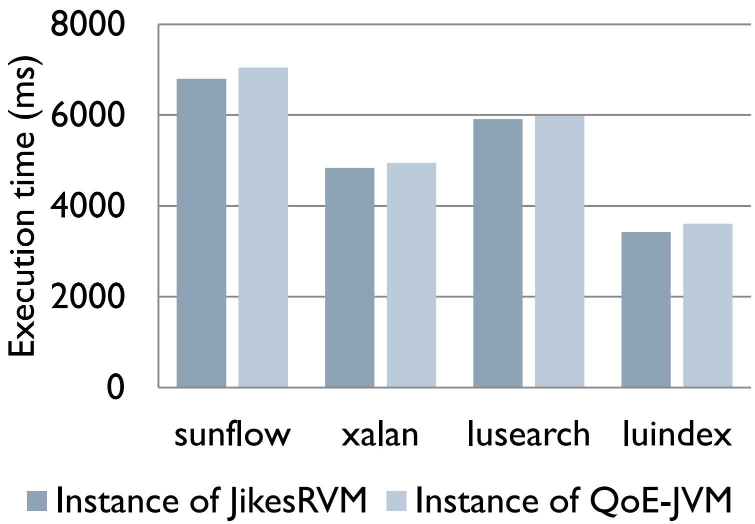


Evaluation

- Questions regarding these extensions
  - Q1: How costly is to account *resource usage* and *execution progress*?
  - Q2: What are the benefits of applying application-tailored policies (e.g. heap policies)?
  - Q3: Which are the costs and benefits of concurrent checkpoint?
- Evaluated with Dacapo benchmarks
  - Each benchmark explores a different aspect of a Java VM, as shown with a principal components analysis using metrics that architecture, code, and memory behavior [18]

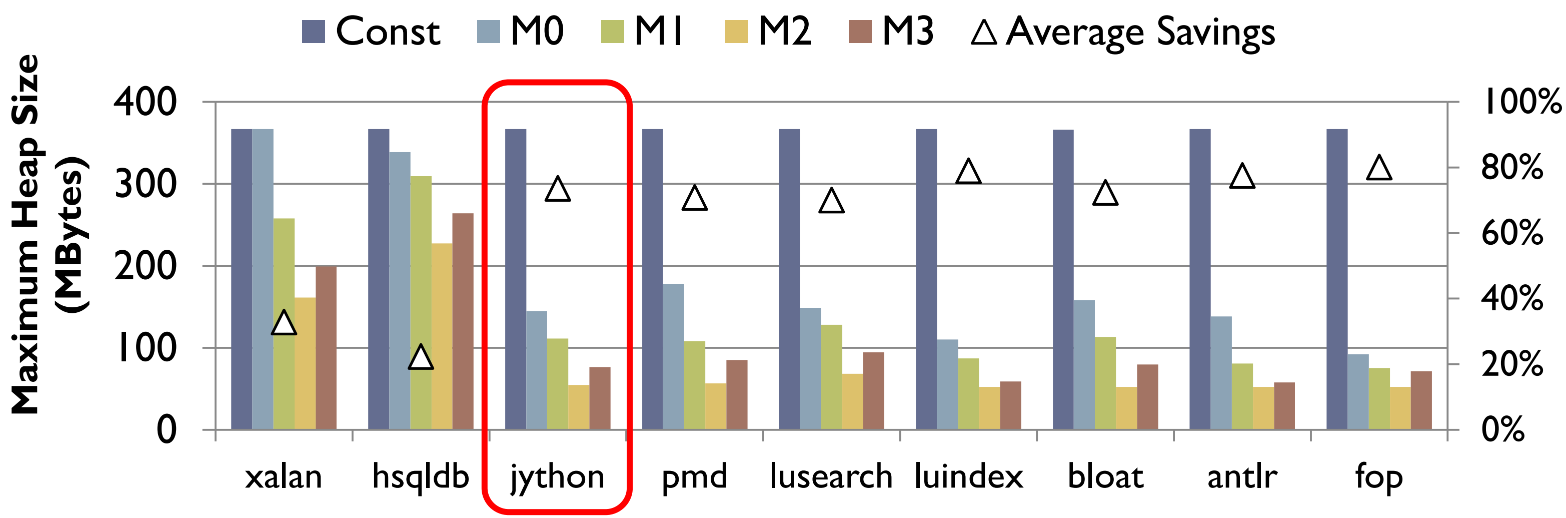
Q1: Accounting *resource usage* and *execution progress*?





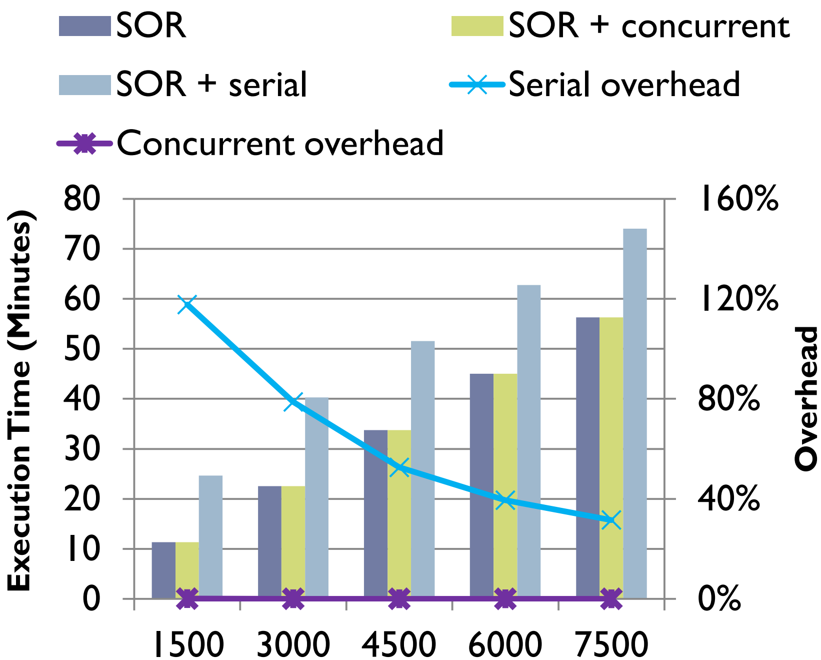
- **Policy evaluation overheads** (for *resource domain* thread creation):
  - +6% to the baseline using a (complex) policy with 50 constraints
  - +3% (average) overhead in real multi-threaded applications
  - The accounting of other resources (mem, cpu) also shows very small overhead
- **Progress monitoring related overheads** (using complete version of *Sunflow*)
  - At load time: +105 ms
  - At run time: +0.5%

Q2: *Yield* applied to heap management

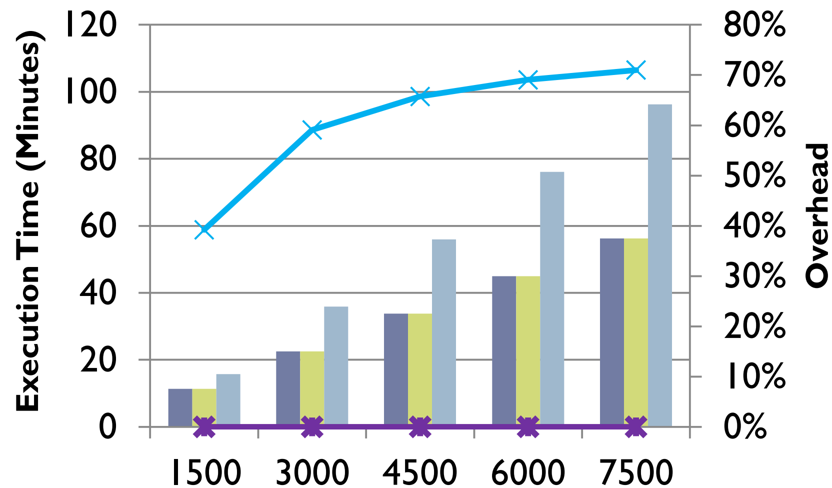


		xalan	hsqldb	jython	pmd	lusearch	luindex	bloat	antlr	fop
Degradation in Execution time	M0	-1.5%	-1.2%	17.6%	8.6%	20.3%	9.1%	14.5%	3.9%	-2.1%
	M1	7.4%	-3.5%	2.5%	7.7%	26.1%	14.7%	12.4%	24.8%	-3.2%
	M2	11.2%	50.9%	80.5%	43.7%	225.5%	26.9%	17.7%	31.0%	18.7%
	M3	7.6%	17.0%	13.1%	23.2%	66.2%	25.0%	-4.9%	39.4%	10.8%
Yield	M0	0.0	**-6.6**	3.4	6.0	**2.9**	**7.7**	3.9	**15.8**	**-36.3**
	M1	4.0	-4.5	**28.4**	**9.2**	2.5	5.2	5.6	3.1	-24.7
	M2	5.0	0.7	1.1	1.9	0.4	3.2	4.8	2.8	4.6
	M3	**6.0**	1.7	6.1	3.3	1.1	3.4	**-15.8**	2.1	7.5

Q3: Checkpointing mechanisms



- 1500 – 7500 linear equations to solve



- Checkpoint at 20%, 40%, 60% and 80% of progress
  - Serial overhead is amortized
- Checkpoint at approximately every 5 minutes
  - Serial overhead increasingly stretches
- The overhead of concurrent checkpoint is negligible - less than 1% in all configurations

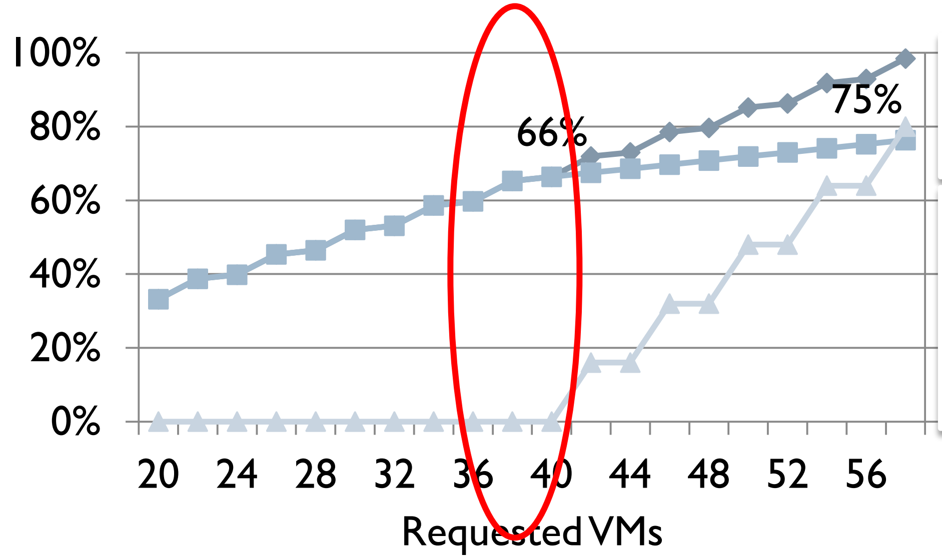
Outline

- Introduction
- Adaptability in virtual machines
- PaaS
  - Models, Mechanisms, Evaluation
- IaaS
  - Models
  - Mechanisms
  - Evaluation
- Energy and Community Clouds
  - Models, Mechanisms, Evaluation
- Publications, Conclusions, Ongoing and Future Work

Life in a small (classic) datacenter

H_{type}	A	B
Cores	2	2
Hz	1860	2660
Mem (Gb)	4	4
# Hosts	10	10

VM_{type}	$x10^3$ MIPS
Small	0.5
Medium	1
Regula	2
Extra	2.5



25% of resources are idle (wasted)

31% of VM requests are rejected

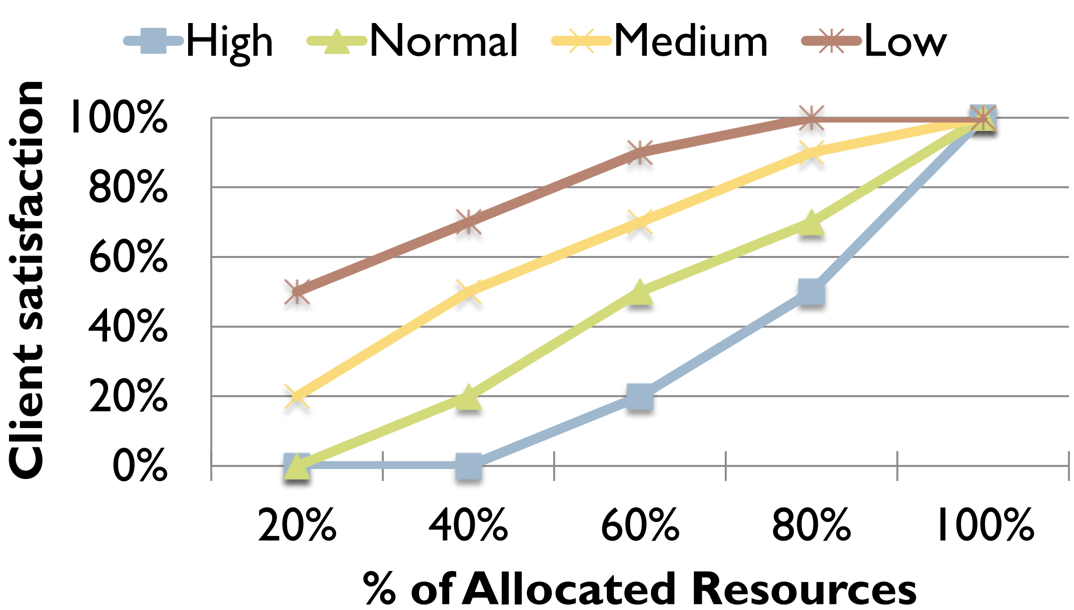
- In summary, clients are not satisfied but datacenters are not fully utilized
  - Idle machines consume ~70% of peak power [19]

Research at the IaaS level - overview

- An **architectural extension** to the current relation between cloud users and providers, particularly useful for private and community cloud deployments;
- A **cost model** which takes into account the clients' partial utility of having their VMs depreciated when in overcommit;
- **Strategies** to determine, in a overcommitted scenario, the best distribution of workloads (from different classes of users) among VMs with different execution capacities, aiming to maximize the utility of the allocation.

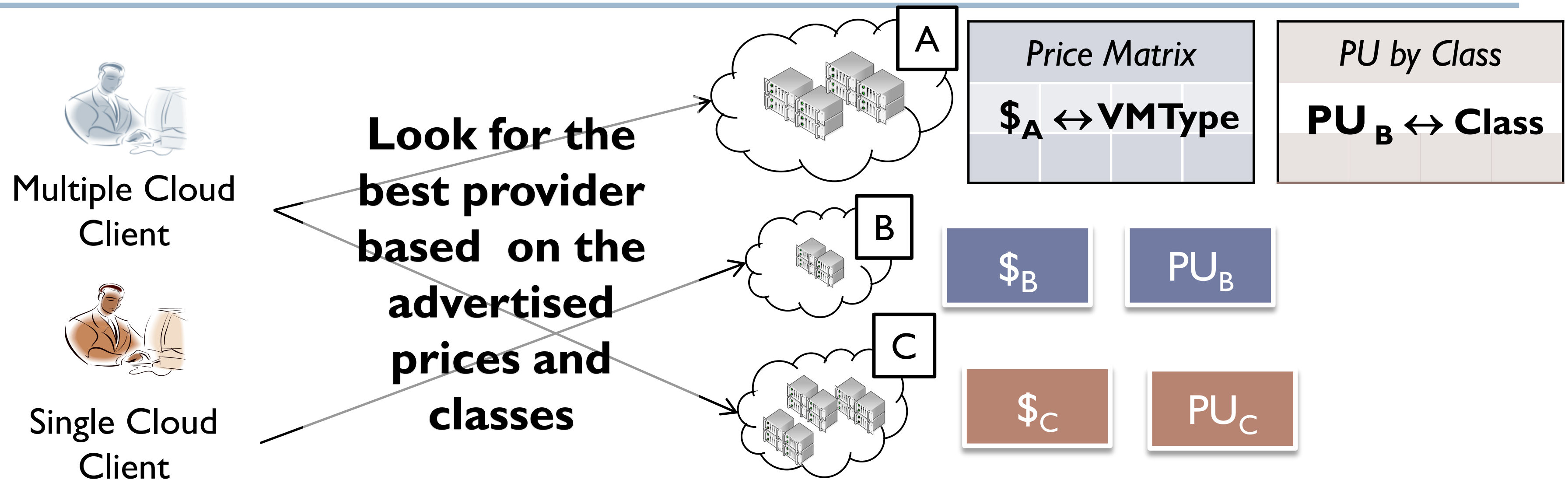
Exploring the remainder "25%"

- Base scenario: A new VM is requested but no space is available without some kind of degradation – results in a *VM rejection*
- Our proposal: Use the user's *partial utility specification*, to explore a degradation factor for each allocated VM



- Provider wants to *maximize VM allocations* while *maximizing clients' satisfaction*

A new cost model



$$VMCost(vm) = Pr(VMType(vm)) * (1 - Df(vm)) * Pu(Df(vm))$$

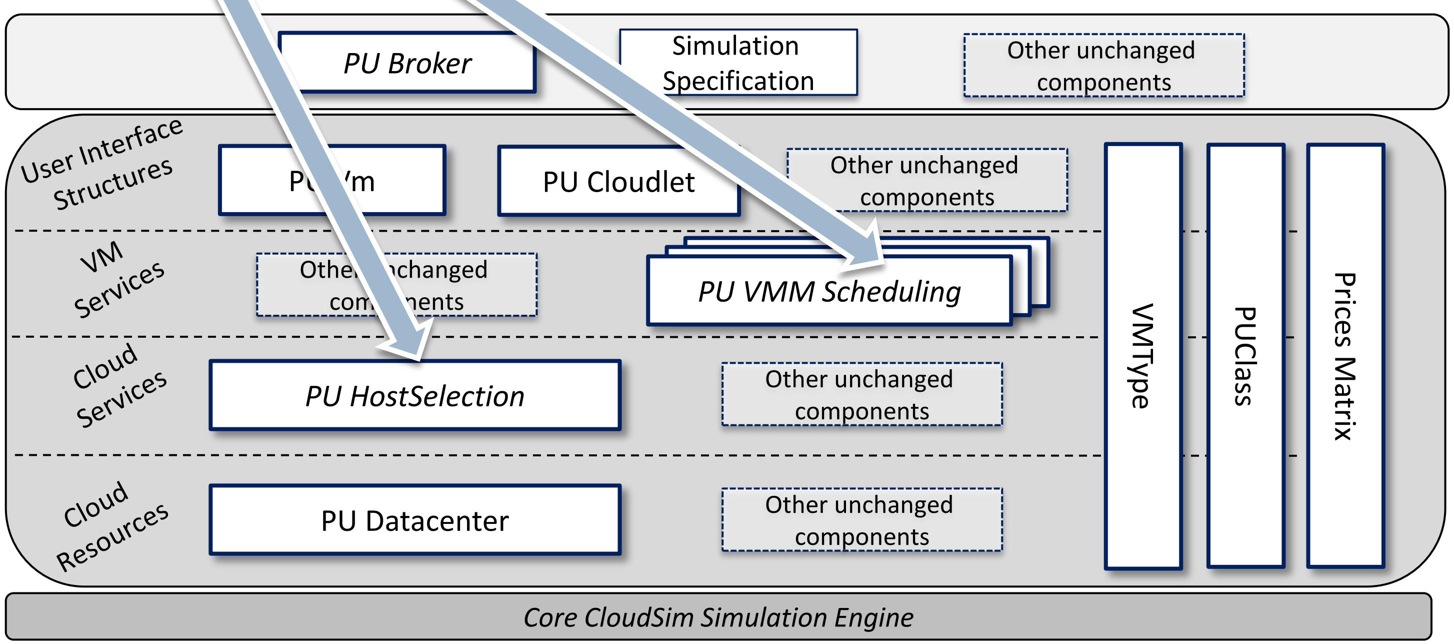
- Price of *vm* based on computational capacity
- VMs are sorted by computational power

- Depreciation factor of *vm*
- *Df(vm)=0* if provider can assign maximum resources

- Partial-utility of client based on the depreciation
- It varies based on the client class

IaaS Scheduling Algorithms

- Resources of requested VMs are changed according to multi-level partial-utility negotiation between the client and the provider
- Heuristics used by the provider
  - Sort hosts by computational power and increasingly take from allocated VMs
  - Asymptotic cost bellow quadratic: *O(nr_hots · nr_vms · lg(nr_vms))*
- Extension to CloudSim [19-21], a highly cited/used cloud simulation framework

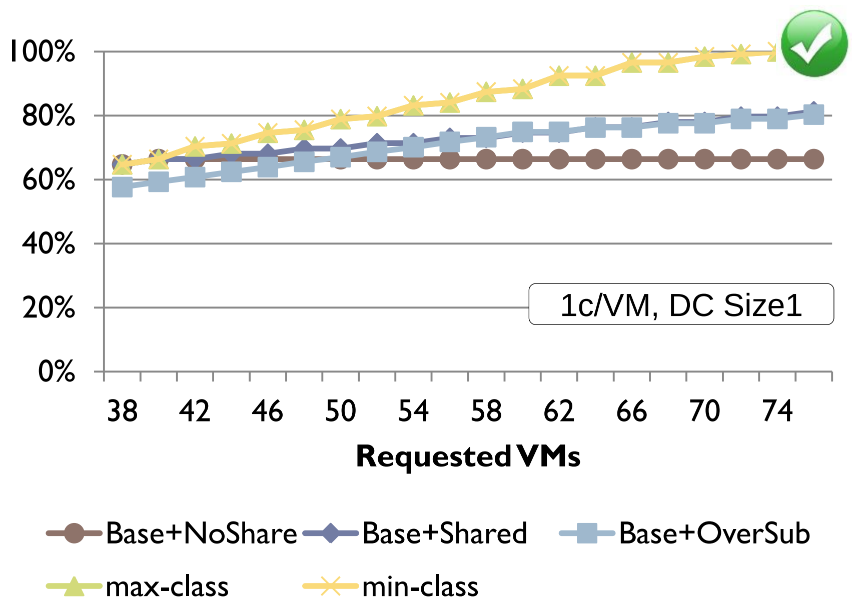


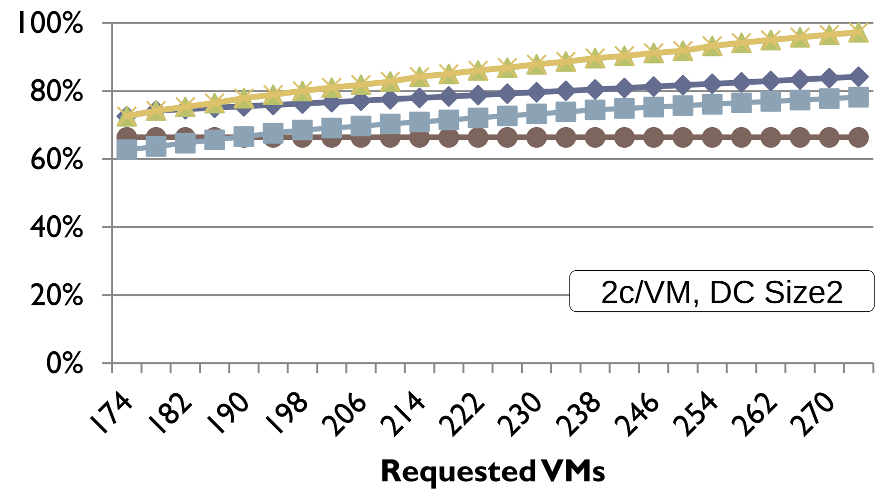
JS, LV @ Transactions on Cloud Computing, IEEE, 2014
JS, LV @ IEEE, CloudCom, 2013, Best Paper runner-up

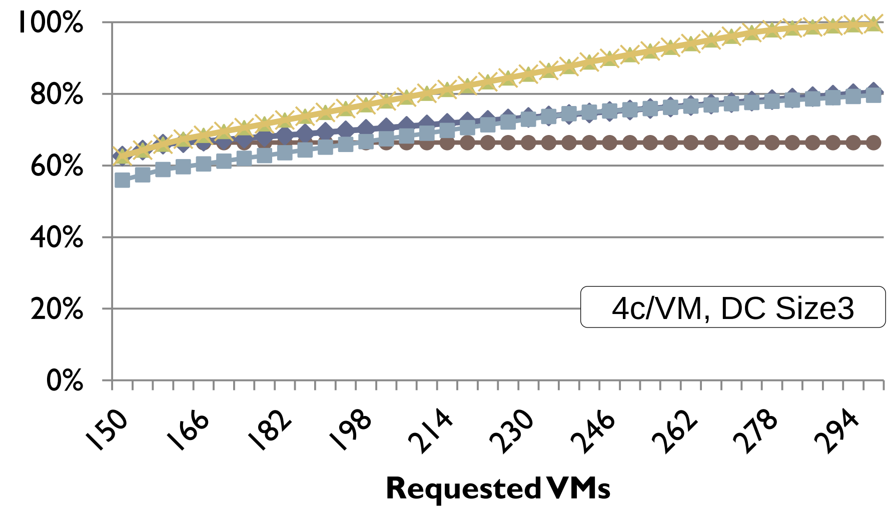
Evaluation

- Questions regarding this cost model and algorithms
  - Q1: Resource usage increases? (provider interest)
  - Q2: Revenue increases? (provider interest)
  - Q3: Impact on the workload execution time (client interest)
  - Transversal: How does this approach scale?
    - DC_1 (2 Cores) DC_2 (4 Cores) and DC_3 (4 Cores+HT)
    - VMs requesting 2 Cores and 4 Cores
- Evaluated with traces from VMs running in PlanetLab [21] collected in the context of the CoMon project [22]
  - A trace from a PlanetLab VMs is assigned to each VM in the simulation

Q1: Resource Usage

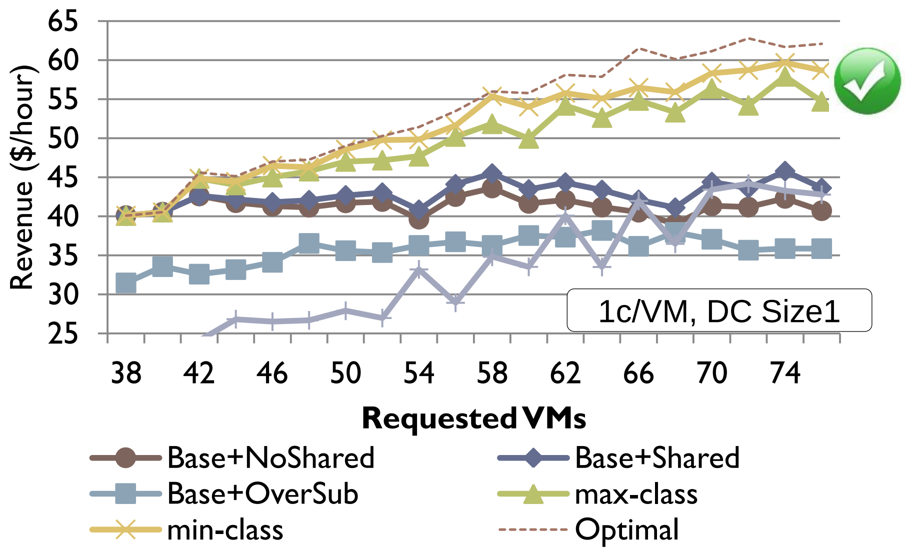


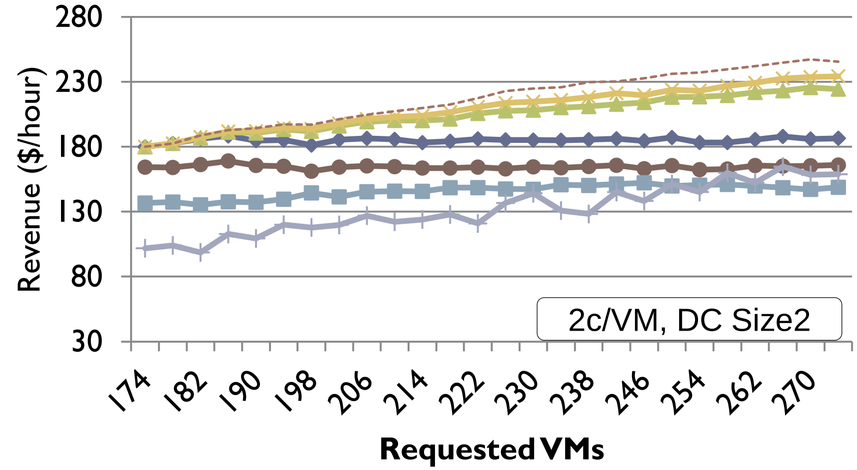




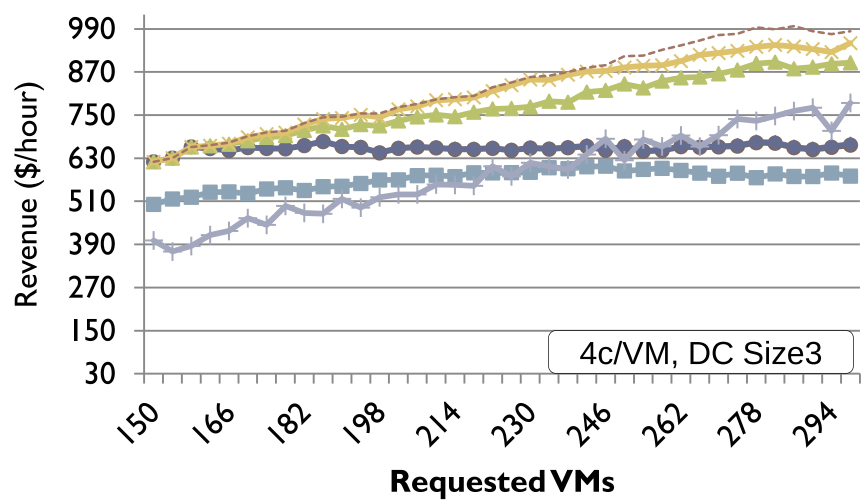
- Utility-driven approaches
  - achieves better resource utilization, while allocating all VMs
  - reach the peak in a similar fashion, across all sizes of datacenters.

Q2: Revenue

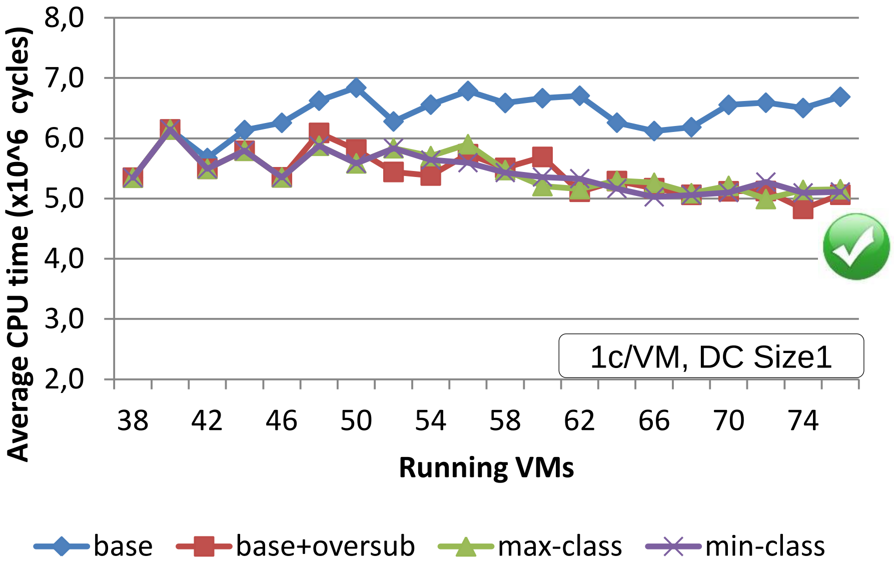


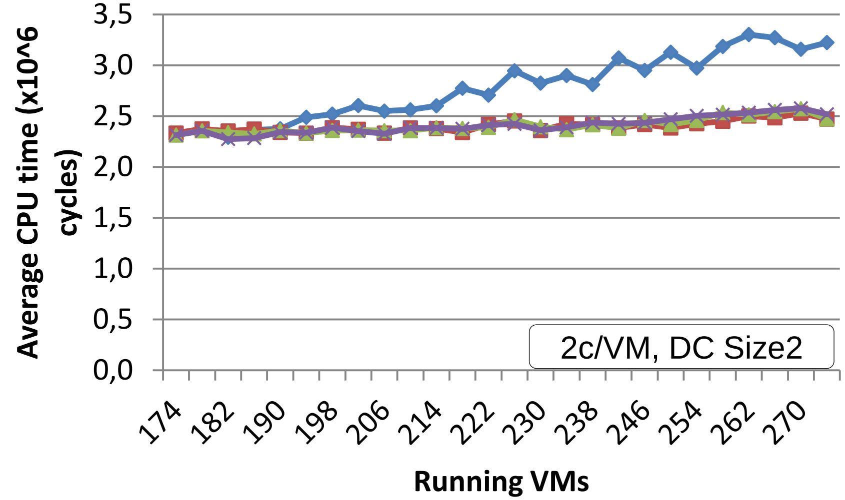


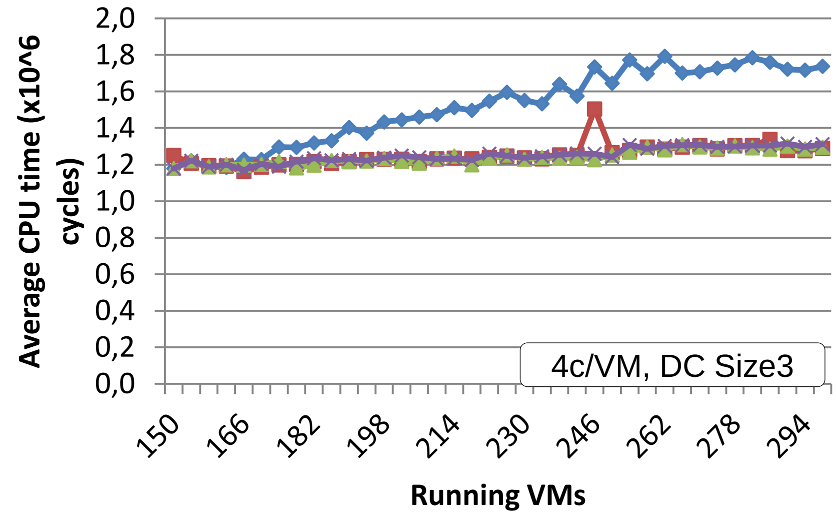
- Revenue increases with more VMs allocated
- What would be rejected VMs are accepted with a partial utility-driven allocation



Q3: Impact in workloads' execution time





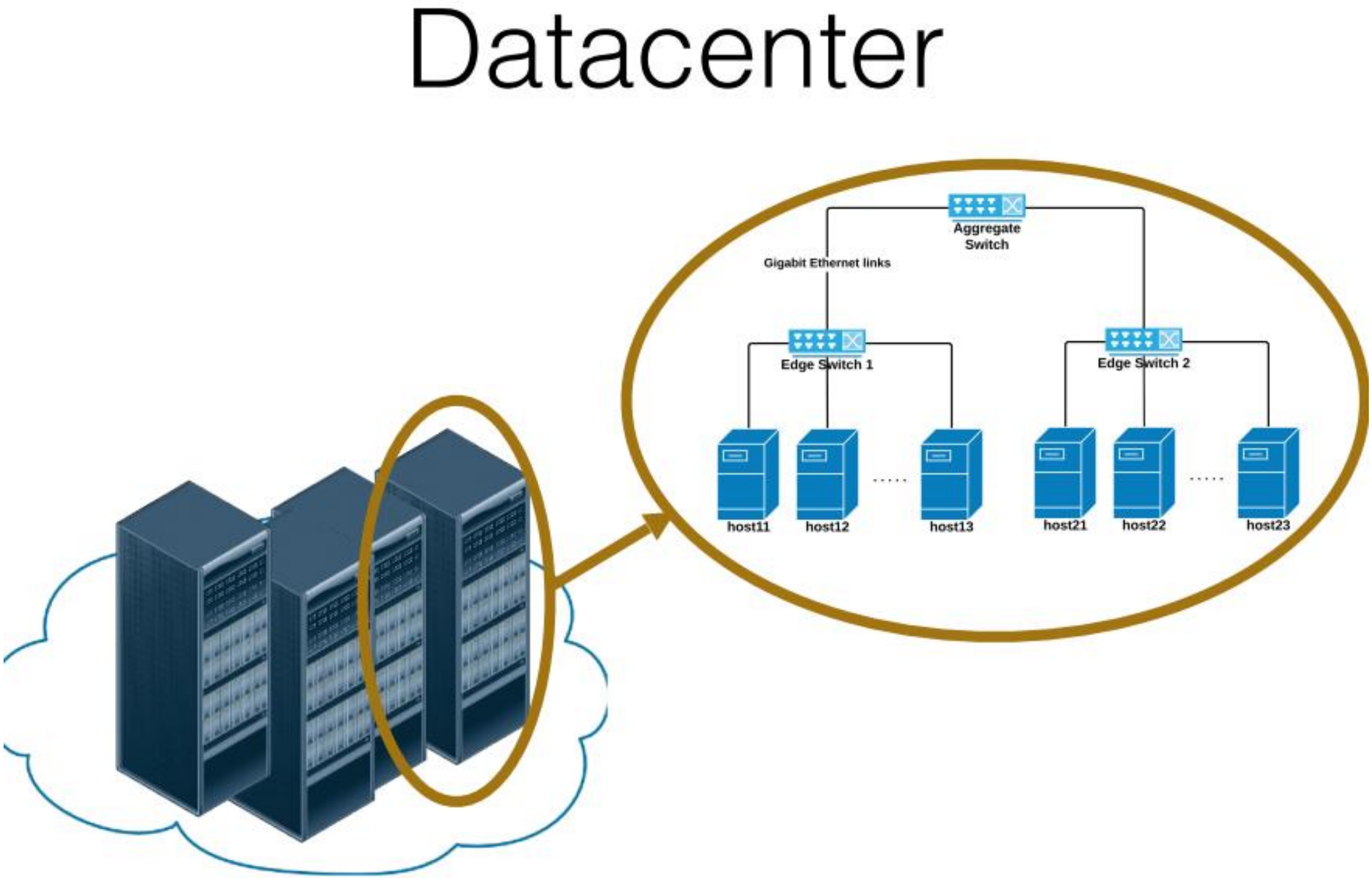


- *With more VMs allocated*, even if with less allocated resources than the ones requested, as it is the case, *average execution time is below the execution times achieved with the base strategies.*

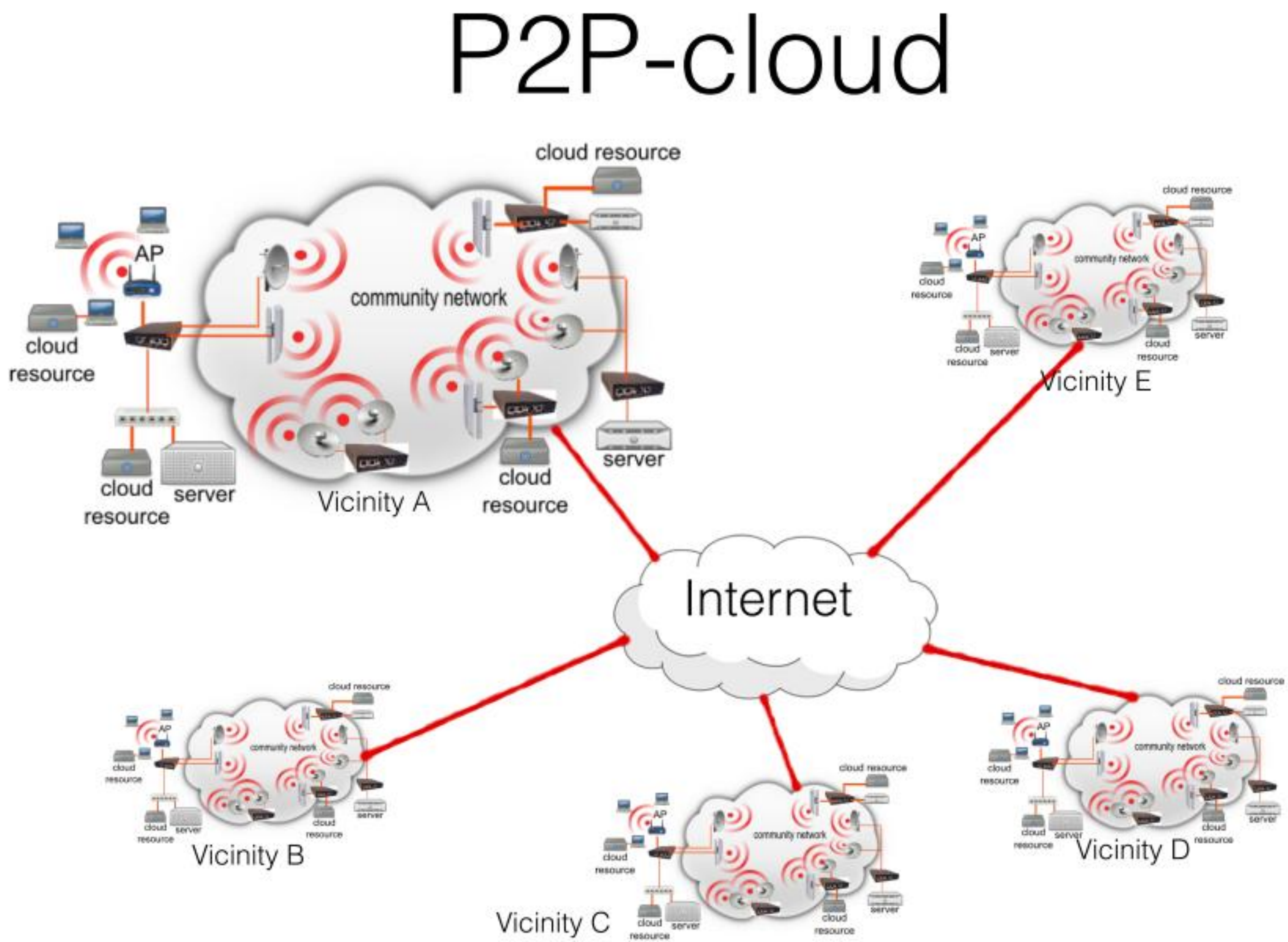
Outline

- Introduction
- Adaptability in virtual machines
- PaaS
  - Models, Mechanisms, Evaluation
- IaaS
  - Models, Mechanisms, Evaluation
- Energy and Community Clouds
  - Models
  - Mechanisms
  - Evaluation
- Publications, Conclusions, Ongoing and Future Work

Life in the **Corporate Clouds**

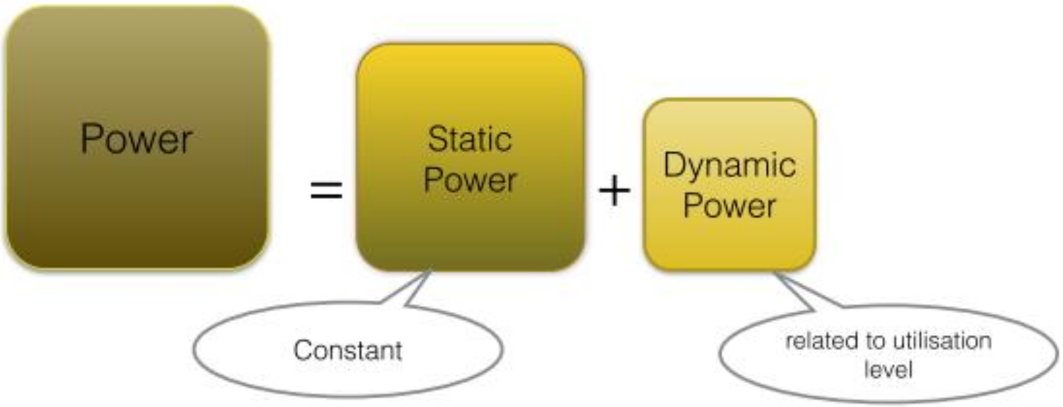


Life in Peer-to-Peer **Community Clouds**



TÉCNICO LISBOA

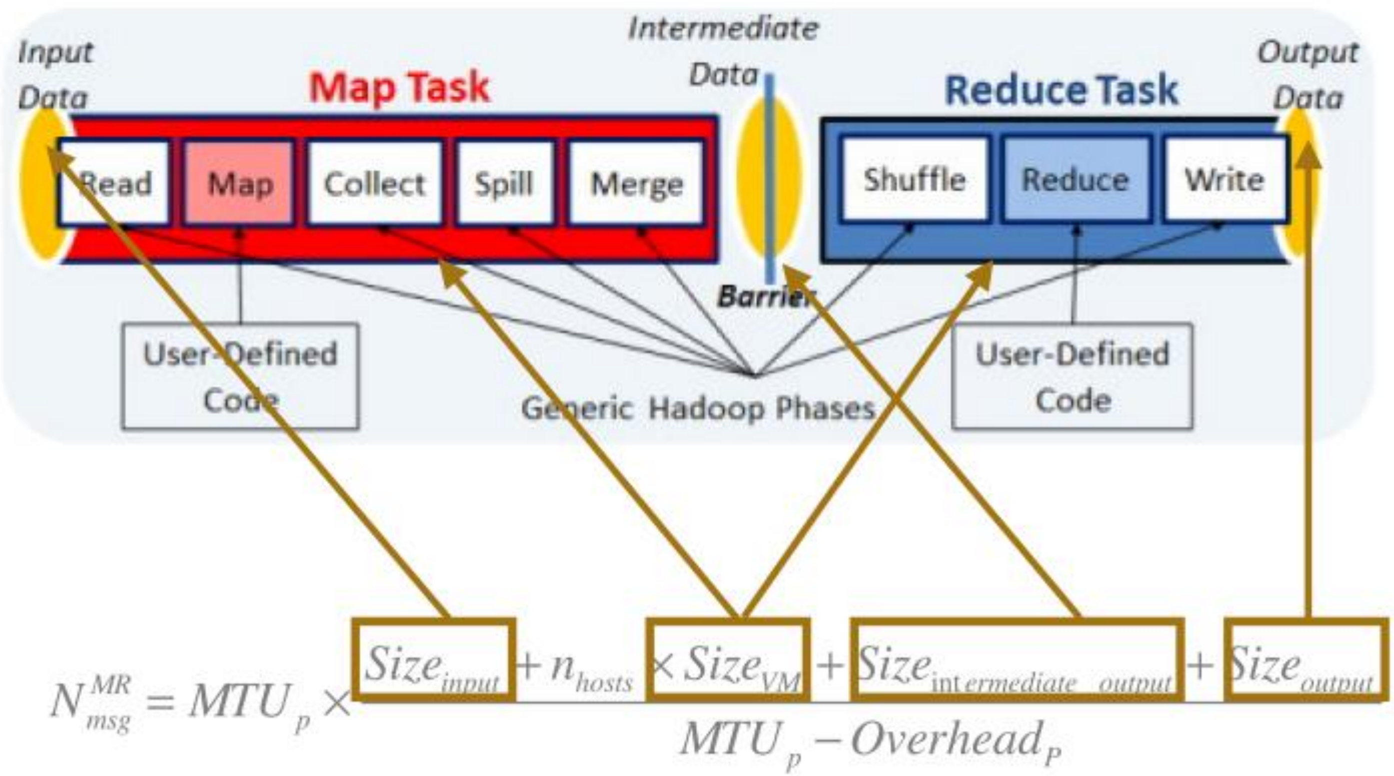
General Power Model



Linear Power Model: $P_{Linear} = P_{static} + (P_{\max} - P_{static}) \times U$

Real world workloads resource consumption

MapReduce case study



$$N_{msg}^{MR} = MTU_p \times \frac{Size_{input} + n_{hosts} \times Size_{VM} + Size_{intermediate\_output} + Size_{output}}{MTU_p - Overhead_p}$$

Model real world workloads energy usage

MapReduce Energy Model

intra-datacenter

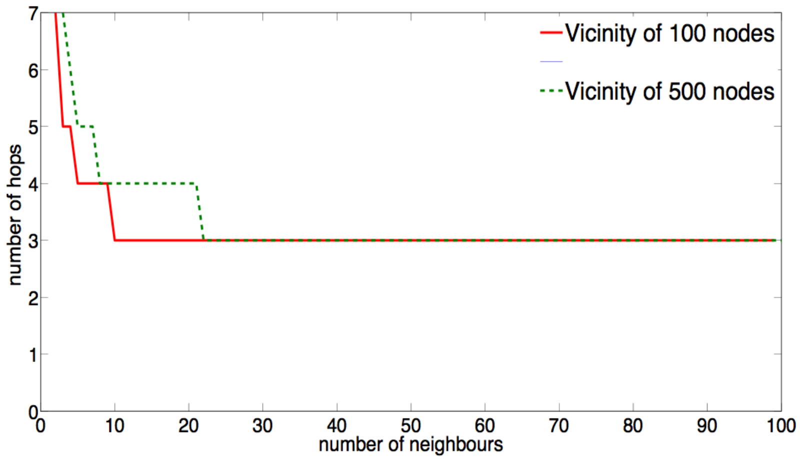
$$E^{MR}_{intra\_DC} = P^{Com}_{intra\_DC}(Size_{msg}) \times t_{send} \times N_{msgs} + \sum_{i=1}^{phases} (E[task]_i \times \sum_{nt} P_{host})$$

Communication — Process

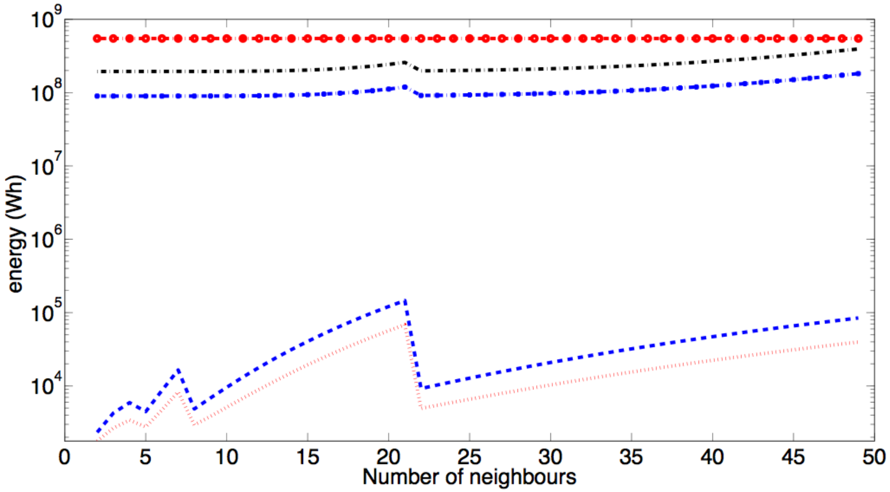
$$E^{MR}_{intra\_P2P} = P^{Com}_{WN}(Size_{msg}) \times t_{send} \times N_{msgs} + \sum_{i=1}^{phases} (E[task]_i \times \sum_{nt} P_{host})$$

intra-vicinity P2P-cloud

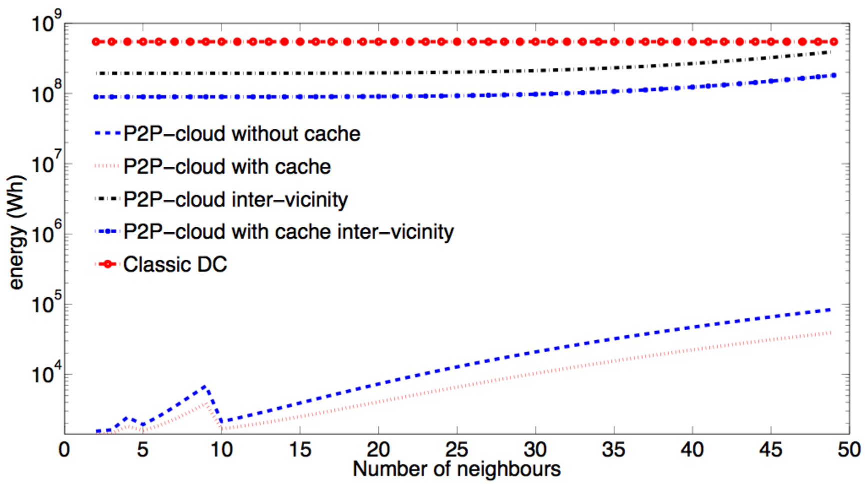
Vicinity Density effect



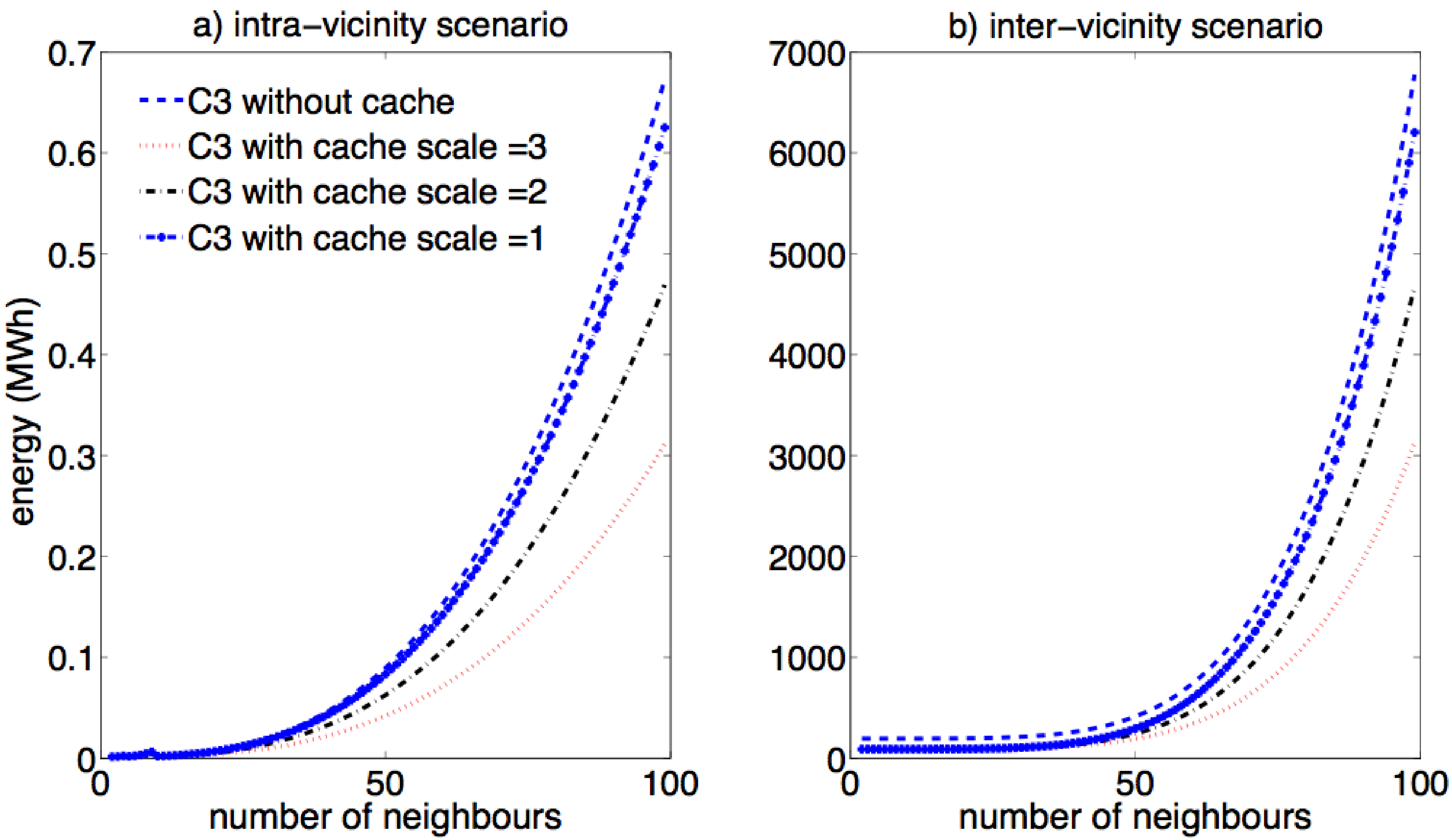
Vicinity of 500 nodes



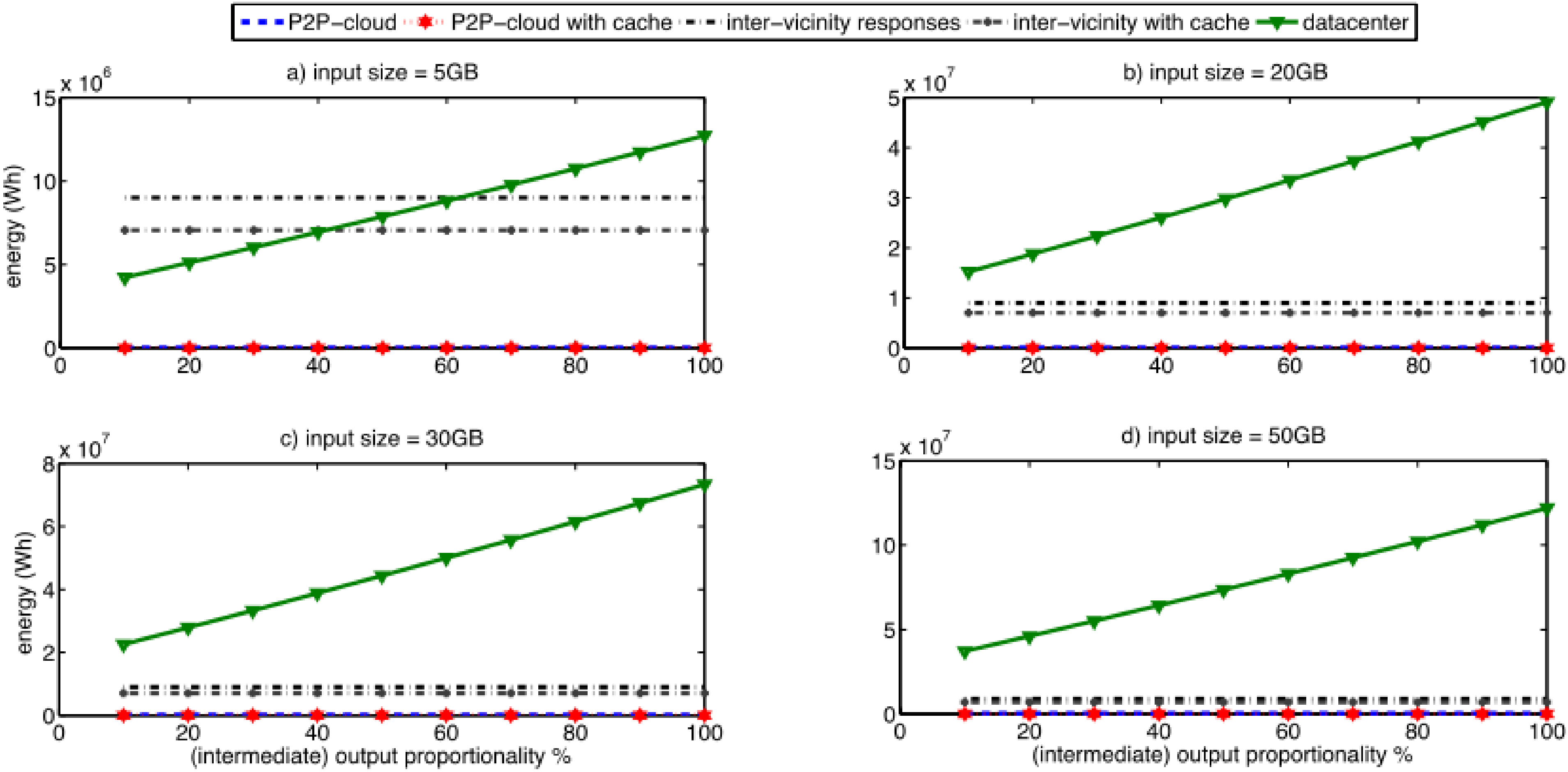
Vicinity of 100 nodes



Impact of cache scale



Input-(Intermediate) Output proportionality



Energy Take Aways

- P2P-cloud can provide an ecosystem for energy efficient decentralised clouds
- Intra-vicinity supply is the most energy efficient.
- Trade-off between energy efficiency and resource availability- Cache mechanism
- However:
  - Not easy to support large VM
  - Performance degradation
- Looking forward:
  - There is room to improve the energy efficiency of P2P-cloud
  - A decision support system for energy aware resource provisioning

Outline

- Introduction
- Adaptability in virtual machines
- PaaS
  - Models, Mechanisms, Evaluation
- IaaS
  - Models, Mechanisms, Evaluation
- Energy and Community Clouds
  - Models, Mechanisms, Evaluation

- Publications, Conclusions, Ongoing and Future Work

Wrapping up – Summary of publications

- LS, NS,FF, and LV, *IEEE 5th Conference on Cloud Computing Technology and Science (CloudCom 2014)* - ***Best-Paper Award Candidate (TCC submission under revision)***
- JS and LV, *IEEE Transactions on Cloud Computing*, Nov. 2014, IEEE, *online first.*
- JS and LV, *International Journal of Computer Systems Science and Engineering*, Nov. 2013, CRL publishing.
- JS et al., *Concurrency and Computation: Practice and Experience*, Sep. 2012, Wiley
- JS and LV, *IEEE 5th Conference on Cloud Computing Technology and Science (CloudCom 2013)* - **Best-Paper Award Runner-up**
- JS et.al., *19th International Conference on Cooperative Information Systems (CoopIS 2011)*, Springer
- JS and LV, *2nd International Symposium on Secure Virtual Infrastructures (DOA-SVI 2012)*
- JS and LV, *28th ACM Symposium On Applied Computing (SAC 2013)*
- JS and . Singer and Luís Veiga, *IEEE 5th International Conference on Cloud Computing Technology and Science (CloudCom 2013).*
- J. P. Silva and JS and LV, ACM/IFIP/Usenix Middleware 2013
- JS and LV, *11th International Workshop on Adaptive and Reflective Middleware (ARM 2012), In conjuntion with Middleware 2012.*
- JS and LV, *12th IEEE/ACM CCGrid 2012 - Doctoral Symposium: Cloud Scheduling, Clusters and Data Centers.*

Top cited

CCGrid

Top cited

Wrapping up – Digital agenda for Europe

- In "A Roadmap for Advanced Cloud Technologies under H2020" [23]
  - «Europe is characterized by a heterogeneity of culture and business practices. It also has an agile SME sector, with companies that often are world leaders in their specialties and are willing to take risks. »
  - *«This must be considered an opportunity, rather than a disadvantage as it forces the European industry to think beyond homogeneous infrastructures with a sufficient amount of resources.* »
  - «Therefore, Europe faces an historic opportunity to 'leapfrog' other world regions […] to play a key role, in the international CLOUD computing market.»
  - «Main immediately relevant work includes:

  Managing the data deluge; intelligent networking; elastic applications; performance and portability; vulnerabilities; reducing lock-in; competition and collaboration; viable business models; »

Economics-inspired Resource and Energy Management for Cloud Environments

**** Thank you for your attention ****

Acknowledgements: PhD and MSc students

José Simão (recently graduated Ph.D.) and
Leila Sharifi (graduating Ph.D. 2016)

Wrapping up – Summary of publications

[J1] José Simão and L Veiga, **Partial Utility-driven Scheduling for Flexible SLA and Pricing Arbitration in Cloud**, *IEEE Transactions on Cloud Computing*, Nov. 2014, IEEE, *online first*

[J2] José Simão and LV, **Adaptability Driven by Quality Of Execution in High Level Virtual Machines for Shared Environments**, *International Journal of Computer Systems Science and Engineering,* Nov. 2013, CRL publishing.

[J3] José Simão et al. , **A Checkpointing-enabled and Resource-Aware Java VM for Efficient and Robust e-Science Applications in Grid Environments**, *Concurrency and Computation: Practice and Experience,* Sep. 2012, Wiley

[C1] José Simão and LV, **Flexible SLAs in the Cloud with Partial Utility-driven Scheduling (Best-Paper Award Runner-up)**, *IEEE 5th International Conference on Cloud Computing Technology and Science (CloudCom 2013)*, Dec. 2013, IEEE.

[C2] José Simão et.al., **A^2-VM: A Cooperative Java VM with Support for Resource-Awareness and Cluster-Wide Thread Scheduling**, *19th International Conference on COOPERATIVE INFORMATION SYSTEMS (CoopIS 2011)*, Sep. 2011, Springer

[C3] José Simão and LV, **A Progress and Profile-driven Cloud-VM for Improved Resource-Efficiency and Fairness in e-Science Environments**, *28th ACM Symposium On Applied Computing (SAC 2013)*, Mar. 2013, ACM

[C4] João Pedro Silva and José Simão and Luís Veiga, **Ditto – Deterministic Execution Replayability-as-a-Service for Java VM on Multiprocessors**, ACM/IFIP/Usenix International Middleware Conference (Middleware 2013), Dec. 2013, Springer.

[C5] Leila Sharifi, Navaneeth Rameshan, Felix Freitag, Luis Veiga. **Energy Efficiency Dilemma: P2P-Cloud vs. Datacenter (Best Paper Candidate),** *IEEE 6th International Conference on Cloud Computing Technology and Science (CloudCom 2014)*, Dec. 2014 IEEE.

[C1] José Simão and LV, **Flexible SLAs in the Cloud with Partial Utility-driven Scheduling (Best-Paper Award Runner-up)**, *IEEE 5th International Conference on Cloud Computing Technology and Science (CloudCom 2013)*, Dec. 2013, IEEE.

[W1] José Simão and Luís Veiga, **A Classification of Middleware to Support Virtual Machines Adaptability in IaaS**, *11th International Workshop on Adaptive and Reflective Middleware (ARM 2012), In conjuntion with Middleware 2012.*

[W2] José Simão and Luís Veiga, **VM Economics for Java Cloud Computing - An Adaptive and Resource-Aware Java Runtime with Quality-of-Execution**, *CCGrid 2012 - Doctoral Symposium: Cloud Scheduling, Clusters and Data Centers*, May 2012, IEEE

References

[1] H. Jin, X. Wang, S. Wu, S. Di, and X. Shi, "Towards optimized fine-grained pricing of IaaS cloud platform," IEEE Trans. Cloud Comput., vol. PP, no. 99, pp 1-1, 2014

[2] Smith, J. and Nair, R. , Virtual Machines: Versatile Platforms for Systems and Processes. Morgan Kaufmann, 2005.

[3] B. Alpern, S. Augart, S. M. Blackburn, M. Butrico, A. Cocchi, P. Cheng, J. Dolby, S. Fink, D. Grove, M. Hind, K. S. McKinley, M. Mergen, J. E. B. Moss, T. Ngo, and V. Sarkar. The Jikes research virtual machine project: building an open-source research community. IBM Systems Journal, 2005.

[4] Open JDK, http://openjdk.java.net/, visited 8-April-2015

[5] Brewer, E. A. (2010). A certain freedom: thoughts on the CAP theorem. In Richa, A. W. and Guerraoui, R., editors, PODC, page 335. ACM.

[6] Blake, C. and Rodrigues, R. (2003). High availability, scalable storage, dynamic peer networks: Pick two. In Jones, M. B., editor, HotOS, pages 1–6. USENIX.

[7] Shao, Z., Jin, H., and Li, Y. (2009). Virtual machine resource management for high performance computing applications. Parallel and Distributed Processing with Applications, International Symposium on, 0:137–144.

[8] Padala, P., Hou, K.-Y., Shin, K. G., Zhu, X., Uysal, M., Wang, Z., Singhal, S., and Merchant, A. (2009). Automated control of multiple virtualized resources. In Proceedings of the 4th ACM European conference on Computer systems, EuroSys '09, pages 13–26, New York, NY, USA. ACM.

[9] Min, C., Kim, I., Kim, T., and Eom, Y. I. (2012). VMMB Virtual machine memory balancing for unmodified operating systems. J. Grid Comput., 10(1):69–84.

[10] Raffaele Quitadamo, Giacomo Cabri, Letizia Leonardi, Mobile JikesRVM: A framework to support transparent Java thread migration, Science of Computer Programming, vol. 70, nr. 2-3, pp. 221-240, 2008, Elsevier

[11] Gupta, D., Lee, S., Vrable, M., Savage, S., Snoeren, A. C., Varghese, G., Voelker, G. M., and Vahdat, A. (2008). Difference engine: harnessing memory redundancy in virtual machines. In Proceedings of the 8th USENIX conference on Operating systems design and implementation, OSDI'08, pages 309–322, Berkeley, CA, USA. USENIX Association.

[12] C. Grzegorczyk, S. Soman, C. Krintz, and R. Wolski. Isla vista heap sizing: Using feedback to avoid paging. In Proceedings of the International Symposium on Code Generation and Optimization, CGO '07, pages 325{340, Washington, DC, USA, 2007. IEEE Computer Society.

References

[13] J. Singer, R. E. Jones, G. Brown, and M. Lujan. The economics of garbage collection. SIGPLAN Notes, 45:103-112, June 2010.

[14] Hinesa, M., Gordon, A., Silva, M., Silva, D. D., Ryu, K. D., and Ben-Yehuda, M., Applications know best: Performance-driven memory overcommit with ginkgo. In CloudCom '11: 3rd IEEE International Conference on Cloud Computing Technology and Science, pages 130–137, 2011.

[15] Yang, T., Berger, E. D., Kaplan, S. F., and Moss, J. E. B. (2006). Cramm: Virtual memory support for garbage-collected applications. In Proceedings of the 7th Symposium on Operating Systems Design and Implementation, OSDI '06, pages 103–116, Berkeley, CA, USA. USENIX Association.

[16] White, D. R., Singer, J., Aitken, J. M., and Jones, R. E. (2013). Control theory for principled heap sizing. In Proceedings of the 2013 International Symposium on Memory Management, ISMM '13, pages 27–38, New York, NY, USA. ACM.

[17] Soman, S. and Krintz, C. (2007). Application-specific garbage collection. J. Syst. Softw., 80:1037–1056.

[18] S. M. Blackburn, R. Garner, C. Homann, A. M. Khang, K. S. McKinley, R. Bentzur, A. Diwan, D. Feinberg, D. Frampton, S. Z. Guyer, M. Hirzel, A. Hosking, M. Jump, H. Lee, J. E. B. Moss, B. Moss, A. Phansalkar, D. Stefanovikc, T. VanDrunen, D. von Dincklage, and B. Wiedermann. The DaCapo benchmarks: Java benchmarking development and analysis. In OOPSLA '06: Proceedings of the 21st annual ACM SIGPLAN conference on Object-oriented programming systems, languages, and applications, NY, USA, 2006. ACM.

[19] A. Beloglazov and R. Buyya, Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers,Concurrency Comput.: Prac. Exp., vol. 24, no. 13, pp. 1397–1420, 2012.

[20] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. De Rose, and R. Buyya, CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms, Softw. Pract. Exper., vol. 41, no. 1, pp. 23–50, Jan. 2011.

[21] Chun, B., Culler, D., Roscoe, T., Bavier, A., Peterson, L., Wawrzoniak, M., and Bowman, M. (2003). Planetlab: An overlay testbed for broad-coverage services. SIGCOMM Comput. Commun. Rev., 33(3):3–12.

[22] K. Park and V. S. Pai, CoMon: A mostly-scalable monitoring system for planetlab, SIGOPS Oper. Syst. Rev., vol. 40, no. 1, pp. 65–74, Jan. 2006.

[23] http://ec.europa.eu/digital-agenda/en/news/roadmap-advanced-cloud-technologies-under-h2020-december-2012, published December 2012, visited April 8 2015

References

[24] Memory overbooking and dynamic control of xen virtual machines in consolidated environments. In Proceedings of the 11th IFIP/IEEE international conference on Symposium on Integrated Network Management, IM'09, pages 630–637, Piscataway, NJ, USA. IEEE Press.

[25] Gong, Z., Gu, X., and Wilkes, J. (2010). Press: Predictive elastic resource scaling for cloud systems. In Network and Service Management (CNSM), 2010 International Conference on, pages 9 –16.

[26] Zhang, Y., Bestavros, A., Guirguis, M., Matta, I., and West, R. (2005). Friendly virtual machines: leveraging a feedback-control model for application adaptation. In Proceedings of the 1st ACM/USENIX international conference on Virtual execution environments, VEE '05, pages 2–12, New York, NY, USA. ACM.

[27] Weng, C., Liu, Q., Yu, L., and Li, M. (2011). Dynamic adaptive scheduling for virtual machines. In Proceedings of the 20th International Symposium on High Performance Distributed Computing, HPDC '11, pages 239–250, New York, NY, USA. ACM.

Wrapping up – Summary of publications



[J1] José Simão and L Veiga, **Partial Utility-driven Scheduling for Flexible SLA and Pricing Arbitration in Cloud**, *IEEE Transactions on Cloud Computing*, Nov. 2014, IEEE, *online first*

[J2] José Simão and LV, **Adaptability Driven by Quality Of Execution in High Level Virtual Machines for Shared Environments**, *International Journal of Computer Systems Science and Engineering,* Nov. 2013, CRL publishing.

[J3] José Simão et al. , **A Checkpointing-enabled and Resource-Aware Java VM for Efficient and Robust e-Science Applications in Grid Environments**, *Concurrency and Computation: Practice and Experience,* Sep. 2012, Wiley

[C1] José Simão and LV, **Flexible SLAs in the Cloud with Partial Utility-driven Scheduling (Best-Paper Award Runner-up)**, *IEEE 5th International Conference on Cloud Computing Technology and Science (CloudCom 2013)*, Dec. 2013, IEEE.

[C2] José Simão et.al., **A²-VM: A Cooperative Java VM with Support for Resource-Awareness and Cluster-Wide Thread Scheduling**, *19th International Conference on COOPERATIVE INFORMATION SYSTEMS (CoopIS 2011)*, Sep. 2011, Springer

[C3] José Simão and LV, **A Progress and Profile-driven Cloud-VM for Improved Resource-Efficiency and Fairness in e-Science Environments**, *28th ACM Symposium On Applied Computing (SAC 2013)*, Mar. 2013, ACM

[C4] João Pedro Silva and José Simão and Luís Veiga, **Ditto – Deterministic Execution Replayability-as-a-Service for Java VM on Multiprocessors**, ACM/IFIP/Usenix International Middleware Conference (Middleware 2013), Dec. 2013, Springer.

[C5] Leila Sharifi, Navaneeth Rameshan, Felix Freitag, Luis Veiga. **Energy Efficiency Dilemma: P2P-Cloud vs. Datacenter (Best Paper Candidate),** *IEEE 6th International Conference on Cloud Computing Technology and Science (CloudCom 2014)*, Dec. 2014 IEEE.

[C1] José Simão and LV, **Flexible SLAs in the Cloud with Partial Utility-driven Scheduling (Best-Paper Award Runner-up)**, *IEEE 5th International Conference on Cloud Computing Technology and Science (CloudCom 2013)*, Dec. 2013, IEEE.

[W1] José Simão and Luís Veiga, **A Classification of Middleware to Support Virtual Machines Adaptability in IaaS**, *11th International Workshop on Adaptive and Reflective Middleware (ARM 2012), In conjuntion with Middleware 2012.*

[W2] José Simão and Luís Veiga, **VM Economics for Java Cloud Computing - An Adaptive and Resource-Aware Java Runtime with Quality-of-Execution**, *CCGrid 2012 - Doctoral Symposium: Cloud Scheduling, Clusters and Data Centers*, May 2012, IEEE