

**11<sup>th</sup> Workshop on Adaptive and Reflective Middleware (ARM 2012)**

-  
**Montreal, Canada**

## A Classification of Middleware to Support Virtual Machines Adaptability in IaaS

José Simão (INESC-ID/ISEL), Luís Veiga (INESC-ID/IST)

Paper available @ <http://dl.acm.org/citation.cfm?doid=2405679.2405684>

# Introduction

---

- ▶ **Virtual machines everywhere**
  - ▶ Resource consolidation and efficiency, coarse grained resource management
- ▶ **VMs adapt resource management at runtime**
  - ▶ Monitor, Decision, Action
  - ▶ Guided by metrics *inside* the codebase or *instructed* by others
- ▶ **How to analyze the *quality of adaptation*?**
  - ▶ Responsiveness, Comprehensiveness and Intricateness



# Agenda

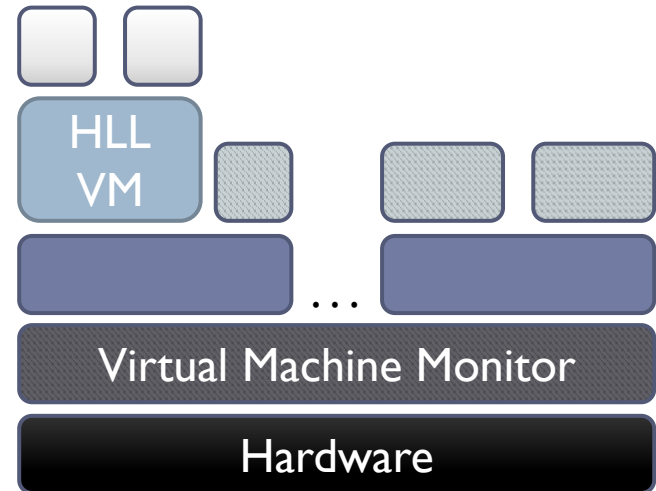
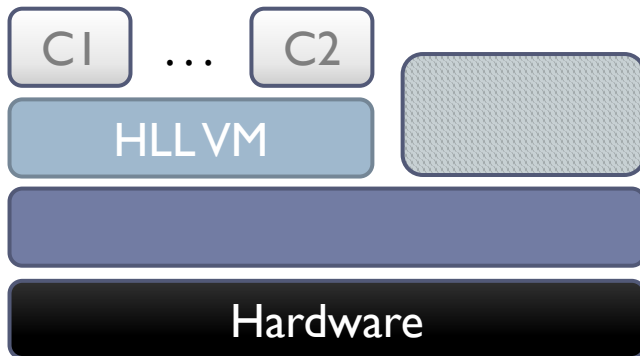
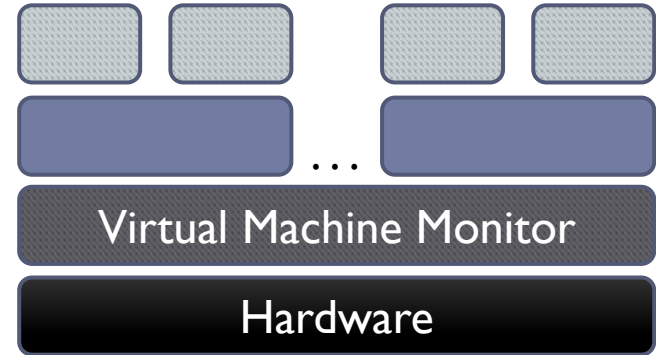
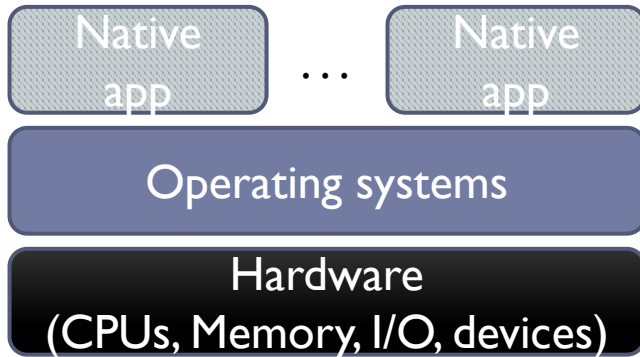
---

- ▶ Virtualization fundamentals
- ▶ Adaptability techniques
- ▶ A classification framework
- ▶ Systems and their classification
- ▶ Conclusions



# Virtualization at different layers

---



# System VMs

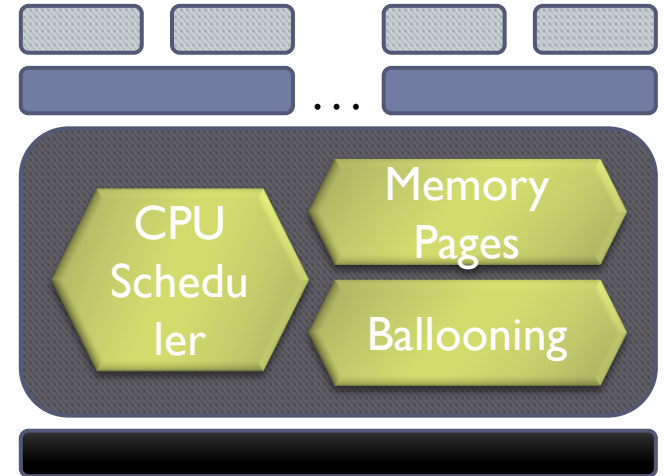
---

- ▶ **Computation as a resource**

- ▶ Emulation of different Instruction Set Architectures (ISA)
- ▶ CPU Scheduling
  - ▶ Enforces user level shares (or weights) and caps

- ▶ **Memory as a resource**

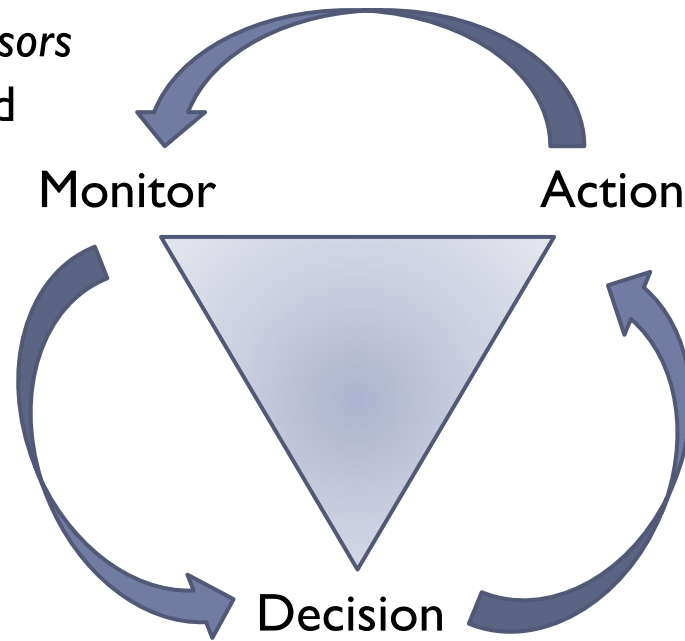
- ▶ Generalizations of OS techniques using *shadow pages*
- ▶ Pages can be shared across guests
- ▶ Transparently transfer pages between guests using memory *ballooning*



# Adaptability loop

---

- ▶ Collect data from *sensors*
- ▶ Event based, threshold checking

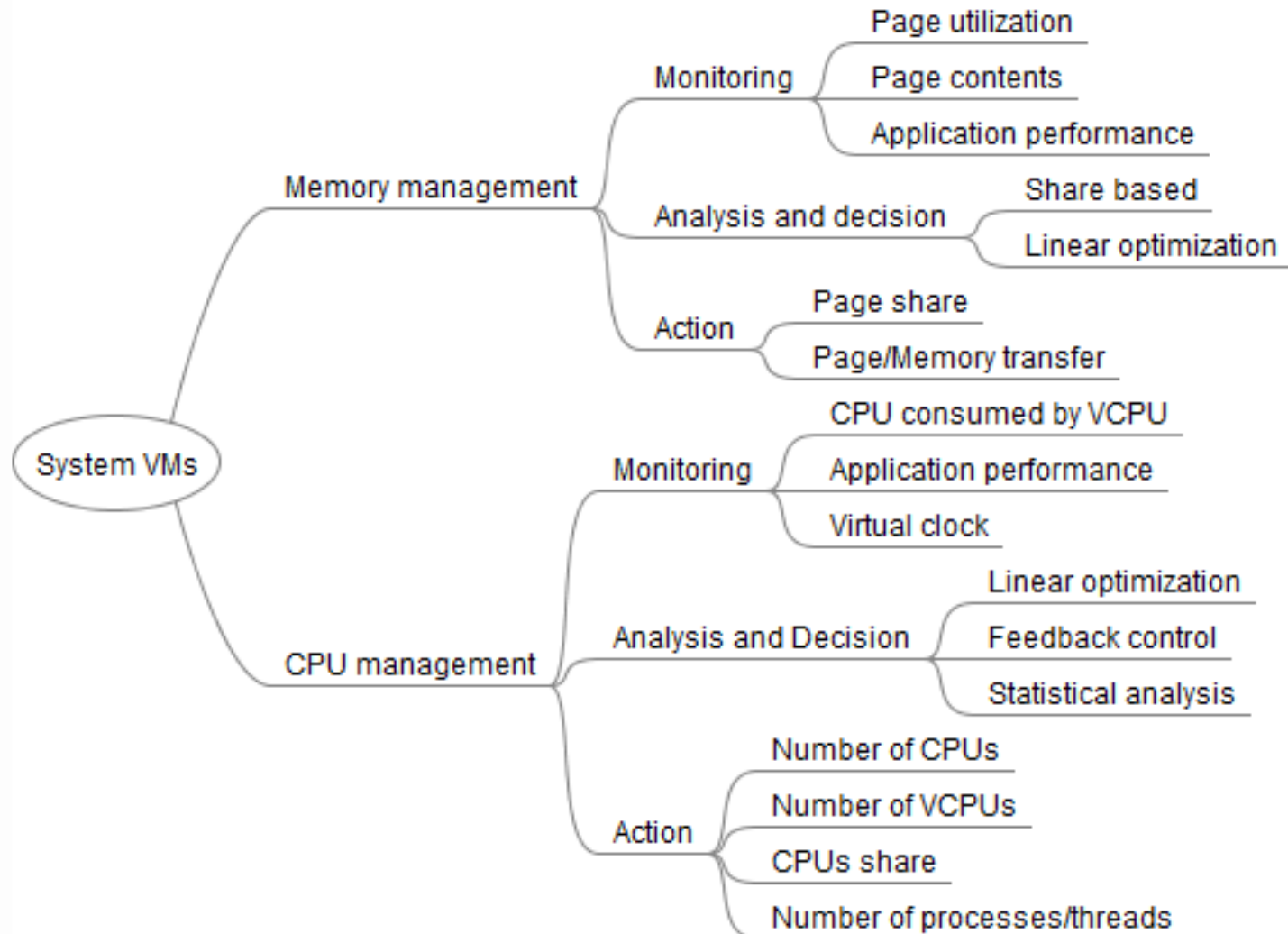


- ▶ Act according to decision using the available *effectors*
- ▶ Change Parameters, algorithms

- ▶ What needs to be changed
- ▶ Decisions made inside or outside the VM determine the complexity of the process



# System VM techniques



# Introduction to the framework

---

- ▶ The *RCI framework* goal – understand and compare different adaptation processes
  - ▶ Responsiveness: how fast can the system adapt?
  - ▶ Comprehensiveness: which is the breadth and scope of the adaptation process?
  - ▶ Intricateness: which is the depth/complexity of the adaption process?
- ▶ The RCI conjecture
  - ▶ *A given adaptation technique aiming at achieving improvements on two of these aspects, can only do so at the cost of the remaining one.*
  - ▶ Similar to other tradeoffs in system research
    - ▶ **C**onsistency, **A**vailability, and tolerance to **P**artitions.
    - ▶ P2P: High Availability, Scalability, and support for Dynamic Populations





# System VM deployments

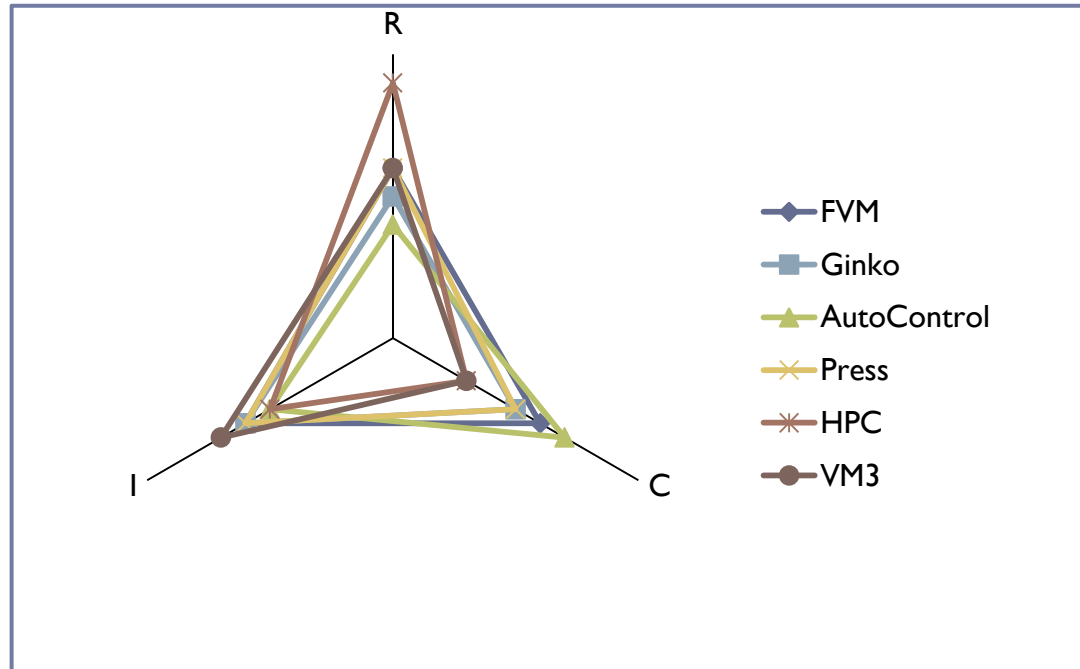
---

- ▶ **Friendly Virtual Machine [49]**
  - ▶ Virtual time clock; Feedback control; Number of processes/threads
- ▶ **HPC computing [36]**
  - ▶ CPU consumed by each VCPU; Share based; Number of VCPUs assigned to CPU
- ▶ **Ginko [28]**
  - ▶ Application's performance; Linear optimization; Page/memory transfer
- ▶ **AutoControl [34]**
  - ▶ Application's performance; CPU consumed by each VCPU; Feedback control; Change shares or caps
- ▶ **PRESS [20]**
  - ▶ CPU consumed by each VCPU; Statistical analysis; Change shares or caps
- ▶ **VM<sup>3</sup> [30]**



# System VM: Classification

---

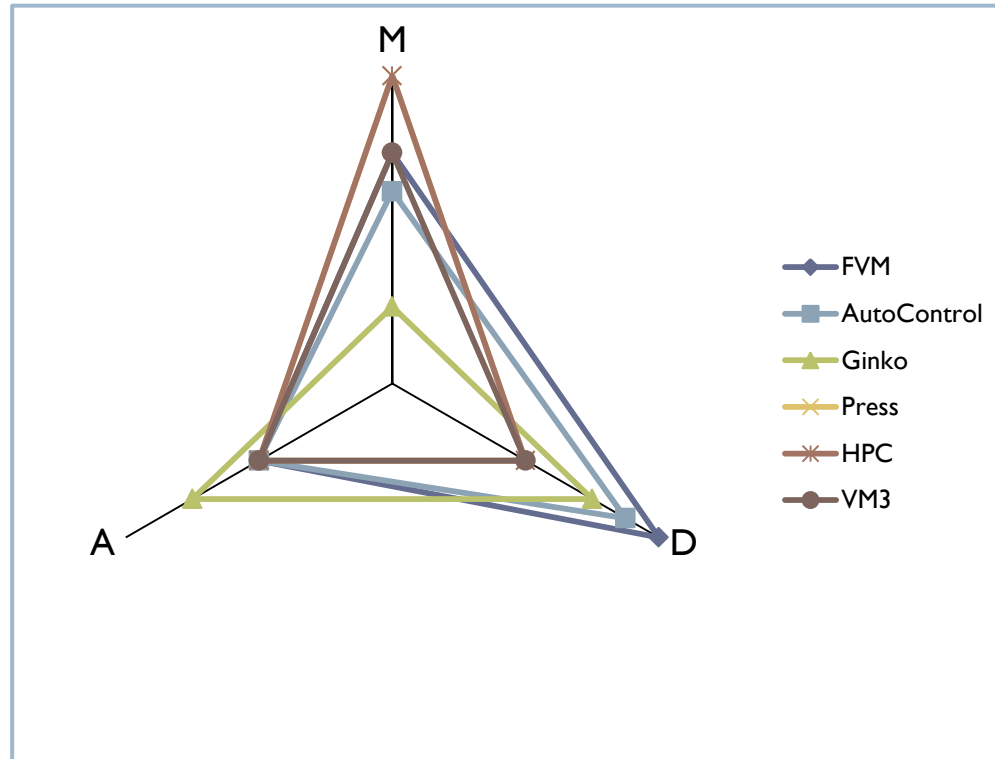


- ▶ Different systems have a different RCI coverage
  - ▶ Intricateness seems to dominate but responsiveness is also strong
  - ▶ Systems with larger R and I are less comprehensive
- 



# Characteristics of the *Adaptability loop*

---



# Conclusions

---

- ▶ Cloud infrastructures depend on VMs to provide support for multiple tenants
- ▶ Resource management is crucial and there is no one-fits-all strategy
  - ▶ VMs must adapt to their guest changing or being instructed to change their parameters or strategies
- ▶ This work
  - ▶ Surveys different adaptation techniques regarding resource management in VMs
  - ▶ Proposes a classification framework to better understand the benefits and limitations of each one
  - ▶ Surveys different systems and frames them into the classification framework
- ▶ In the future
  - ▶ New systems and new techniques can be added to enrich the analysis
  - ▶ Values regarding the RCI of techniques should also depend on measurable aspects (e.g. ratio of functional and monitoring code)



# References

---

[1] <http://wiki.xensource.com/xenwiki/creditscheduler>.

[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 Syst. J.*, 44:399{417, January 2005.

[4] G. M. Amdahl, G. A. Blaauw, and F. P. Brooks. Architecture of the ibm system/360. *IBM J. Res. Dev.*, 8:87{101, April 1964.

[8] Henry G. Baker. Thermodynamics and garbage collection. *SIGPLAN Not.*, 29:58{63, April 1994.

[9] Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, Alex Ho, Rolf Neugebauer, Ian Pratt, and Andrew Wareld. Xen and the art of virtualization. *SIGOPS Oper. Syst. Rev.*, 37:164{177, October 2003.

[12] Charles Blake and Rodrigo Rodrigues. High availability, scalable storage, dynamic peer networks: Pick two. In Michael B. Jones, editor, *HotOS*, pages 1{6. USENIX, 2003.

[13] Eric A. Brewer. A certain freedom: thoughts on the cap theorem. In Andrea W. Richa and Rachid Guerraoui, editors, *PODC*, page 335. ACM, 2010.

[14] Ludmila Cherkasova, Diwaker Gupta, and Amin Vahdat. Comparison of the three cpu schedulers in xen. *SIGMETRICS Perform. Eval. Rev.*, 35:42{51, September 2007.

[20] Zhenhuan Gong, Xiaohui Gu, and J. Wilkes. Press: Predictive elastic resource scaling for cloud systems. In *Network and Service Management (CNSM), 2010 International Conference on*, pages 9 {16, oct. 2010.

[21] Kinshuk Govil, Dan Teodosiu, Yongqiang Huang, and Mendel Rosenblum. Cellular disco: resource management using virtual clusters on shared-memory multiprocessors. *SIGOPS Oper. Syst. Rev.*, 33:154{169, December 1999.

[24] Ajay Gulati, Arif Merchant, Mustafa Uysal, and Peter J. Varman. Ecient and adaptive proportional share i/o scheduling. Technical report, HP Laboratories Palo Alto, 2007.

[28] Michael Hines, Abel Gordon, Marcio Silva, Dilma Da Silva, Kyung Dong Ryu, and Muli Ben-Yehuda. Applications know best: Performance-driven memory overcommit with ginkgo. In *CloudCom '11: 3rd IEEE International Conference on Cloud Computing Technology and Science*, 2011.

[30] Ravi Iyer, Ramesh Illikkal, Omesh Tickoo, Li Zhao, Padma Apparao, and Don Newell. Vm3: Measuring, modeling and managing vm shared resources. *Computer Networks*, 53(17):2873{288, December 2009.

---



# References

---

- [34] Pradeep Padala, Kai-Yuan Hou, Kang G. Shin, Xiaoyun Zhu, Mustafa Uysal, Zhikui Wang, Sharad Singhal, and Arif Merchant. 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, 2009. ACM.
- [35] Mazeiar Salehie and Ladan Tahvildari. Self-adaptive software: Landscape and research challenges. ACM Trans. Auton. Adapt. Syst., 4:14:1{14:42, May 2009.
- [36] Zhiyuan Shao, Hai Jin, and Yong Li. Virtual machine resource management for high performance computing applications. Parallel and Distributed Processing with Applications, International Symposium on, 0:137{144, 2009.
- [39] Jim Smith and Ravi Nair. Virtual Machines: Versatile Platforms for Systems and Processes. Morgan Kaufmann, 2005.
- [42] I. Stoica, H. Abdel-Wahab, and K. Jeay. On the duality between resource reservation and proportional share resource allocation. Technical report, Norfolk, VA, USA, 1996.
- [43] Andrew S. Tanenbaum. Modern Operating Systems. Prentice Hall Press, Upper Saddle River, NJ, USA, 3rd edition, 2007.
- [44] VMware. Vmware vspher 4: The cpu scheduler in vmware esx 4.
- [45] Carl A. Waldspurger. Lottery and stride scheduling: flexible proportional-share resource management. PhD thesis, 1995.
- [46] Carl A. Waldspurger. Memory resource management in vmware esx server. SIGOPS Oper. Syst. Rev., 36:181{194, December 2002.
- [47] Zhao Weiming and Wang Zhenlin. Dynamic memory balancing for virtual machines. In Proceedings of the 2009 ACM SIGPLAN/SIGOPS international conference on Virtual execution environments, VEE '09, pages 21{30, 2009.
- [48] Paul R. Wilson. Uniprocessor garbage collection techniques. In Proceedings of the International Workshop on Memory Management, IWMM '92, pages 1{42, London, UK, 1992. Springer-Verlag.
- [49] Yuting Zhang, Azer Bestavros, Mina Guirguis, Ibrahim Matta, and Richard West. 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, 2005. ACM.
- 

