

Energy-aware Processor Management for Virtual Machines

Jan Stoess Christian Lang Marcus Reinhardt
(PhD/Master) Students at the University of Karlsruhe, Germany

{stoess, chlang, reinhardt}@ira.uka.de

Over the past years, virtualization technology has regained considerable attention in the design of computer systems. Virtual machines establish a development path for incorporating new functionality (server consolidation, transparent migration, secure computing, etc.) into a system that simultaneously retains compatibility to existing OSes and applications. At the very same time, with the ever increasing power density and dissipation of modern servers, energy management is becoming more and more a key concern in the design of operating systems. Research has proposed several approaches to OS-directed control over a computer's energy consumption, including user- and service-centric management schemes [1, 5]. However, all existing approaches assume full control over and full knowledge of the underlying machine – which obviously does not hold true in virtualized environments. Conversely, current virtualization solutions disregard most or all energy-related aspects of the hardware platform, effectively limiting power management to the capabilities of the host-OS in hosted solutions and dispelling it completely from the server-oriented hypervisor solutions.

The key point of this poster is the case for energy-aware, two-level processor management in hypervisor-based server systems. We argue that virtual server infrastructure needs energy management at both layers, within VMs and across VMs. On the one hand, only guest OSes can pursue fine-grain energy management and keep up guest-intrinsic application- or user-specific service demands. On the other hand, only the hypervisor and its resource management subsystem can control global, machine-wide energy requirements and conditions. Host-level resource management also remains the resort that enforces given energy requirements for malicious, defect, or simply energy-unaware guests.

We are currently developing a management framework for energy-aware processor management in virtualized environments. A host-level scheduler subsystem, responsible for processor allocation to VMs, controls each processor's energy consumption by migrating VMs according to their respective energy consumptions; it resorts to preemption in case that migration should not be sufficient. For energy profiling of VMs, the scheduler relies on a runtime accounting infrastructure that estimates the energy consumption by instrumenting performance counters, an approach originally introduced in [1]. The infrastructure instruments both hypervisor and subordi-

nate I/O virtualization layers, to include the “hidden” energy dissipation spent in I/O emulation – whose overhead a recent study has shown to be substantial [2]. Via faithful virtualization of the processor's performance counters, which factors out the energy spent by other, concurrently executing VMs, we enable an existing energy-aware OS scheduler [3] to be reused within a guest operating system. The guest scheduler operates on a per guest-task base: it profiles each task's energy dissipation and triggers migration and preemption in case of “overheating virtual processors”. Virtual heat control is currently based on fixed per-processor, per-VM energy budgets, and a para-virtual driver module propagating the budgets to guest schedulers.

Beyond establishing a mere coexistence of the two schedulers, we are further investigating the borders of interaction of both scheduling layers. We are exploring several mechanisms: i) injection of energy consuming balloon processes into a VM, similar to the time ballooning approach in [4], to enable the virtual infrastructure to dynamically even out processor imbalances in a non-intrusive way; ii) para-virtual, guest-task aware I/O-virtualization as a mechanism to account the energy spent outside a VM (e.g., in I/O emulation routines) back to the VMs, at application granularity; and iii) virtual processor hot-plugging, as an energy valve that enables self-balancing of overheating VMs.

References

- [1] F. Bellosa, F. Weissel, M. Waitz, and S. Kellner. Event-driven energy accounting for dynamic thermal management. In *Proceedings of the Workshop on Compilers and Operating Systems for Low Power*, September 2003.
- [2] L. Cherkasova and R. Gardner. Measuring CPU overhead for I/O processing in the Xen virtual machine monitor. In *Proceedings of the USENIX annual technical conference*. USENIX Association, April 2005.
- [3] A. Merkel and F. Bellosa. Balancing power consumption in multiprocessor systems. In *Proceedings of the 1st EuroSys conference*. ACM Press, April 2006.
- [4] V. Uhlig, J. LeVasseur, E. Skoglund, and U. Dannowski. Towards scalable multiprocessor virtual machines. In *Proceedings of the third Virtual Machine Research and Technology Symposium*, pages 43–56. USENIX Association, May 6–7 2004.
- [5] H. Zeng, C. S. Ellis, A. R. Lebeck, and A. Vahdat. Ecosystem: managing energy as a first class operating system resource. In *Proceedings of the 10th international conference on Architectural support for programming languages and operating systems*. ACM Press, October 2002.