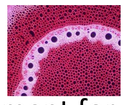
Cellule: Lightweight Execution Environment for Virtualized Accelerators

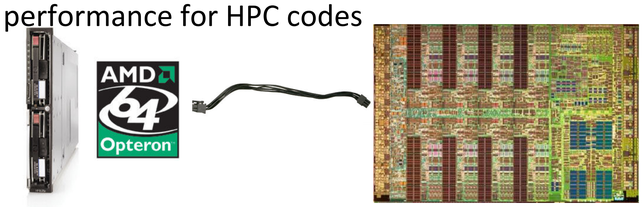
Vishakha Gupta, Jimi Xenidis, Priyanka Tembey, Karsten Schwan, Ada Gavrilovska

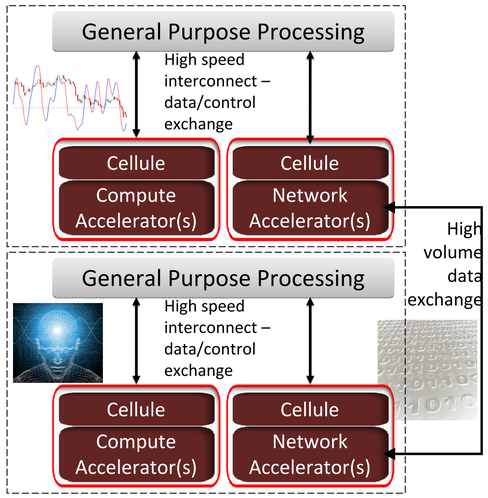
Georgia Tech College of Computing

What is Cellule?



- *Cellule* – small, custom execution environment for accelerators (*cellule* n., a small cell)
- specialized VM for access to and management of accelerator resources (e.g., STI Cell, Intel IXP, etc.).
- helps improve portability and attain high performance for HPC codes



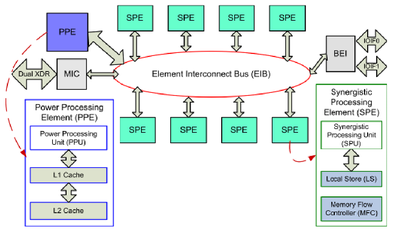


Objectives

Improved performance and lightweight, specialized environment for accelerators

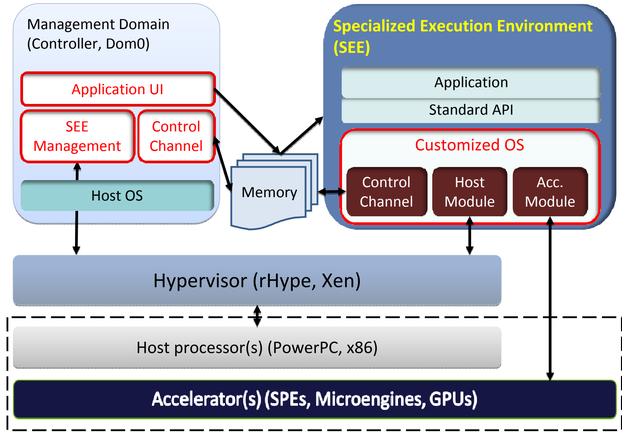
- Accelerator specific memory and execution model
- Simple task model
- Ease of customization and debugging
- Improved signaling mechanisms
- Finer grained scheduling and resource management

Example Accelerator – Cell B.E



- Importance of asynchronous operation and PPE decoupling
- Architectural support for virtualization
- Lightweight environment conducive to high performance and customization

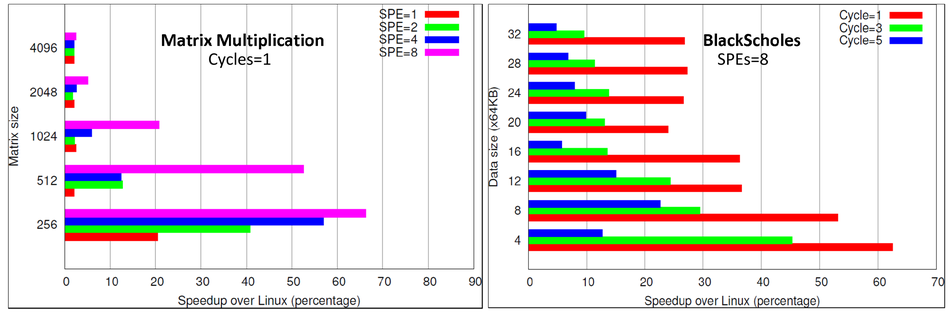
Cellule Architecture

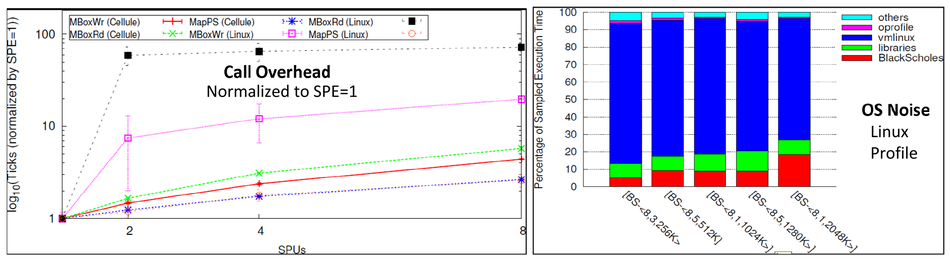


Discussion

- Adapt the architecture to various accelerators – Cell, IXP
  - Use hypervisor and a management domain to
    - Provide familiar UI to user
    - Run the requested application in a lightweight 'special execution environment' (SEE)
- Hypervisor manages resources, multiplexes SEEs
  - Bypass it once accelerator acquired by SEE – fast access

Cellule on Cell B.E vs. Linux





Benchmarks

- Matrix multiplication and Black Scholes - variable
  - Number of SPEs
  - Number of iterations
  - Data sizes
- Experiments on Cell Blade QS20

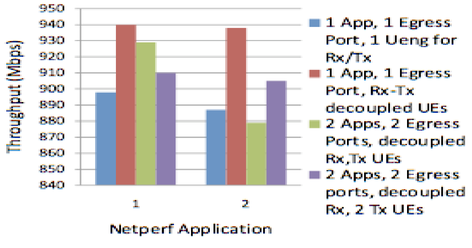
Results - highlights

- Cellule better than Linux mostly
- Improved setup and execution time due to adapted memory and simplified task model
- Particularly important for smaller problem sizes, i.e., `difficult' accelerator applications

Cellule on IXP – 'Open' vs. 'Closed' Accelerators

IXP used as an open accelerator for fine-grained resource management

- Offers gray-box view to management domain:
  - Open access to IXP processing engines and memory through well-defined kernel or hypervisor extensions
  - Lightweight signaling for fine-grain resource-control
- Scheduler co-ordination opportunities:
  - IXP and Host schedulers interact to tune asynchronous accelerator invocations with Host process or VM execution.



- Netperf TCP throughput increases by dedicating IXP resources to certain packet-flows
- Fine-gain control of IXP micro-engines to achieve throughput differentiation

Ongoing Work

- On Cell:-
  - Optimization of SEE, rHype with thread, SMP support
  - Sophisticated scheduling possibly with dynamic partitioning of SPEs
  - IO over 9P communication channel to achieve RoadRunner configuration
- Extend Cellule to other accelerators
- On IXP:
  - Expressive message abstractions between Host cores and Accelerator cores for better coordination

References

- IBM Research Hypervisor (rHype), http://www.research.ibm.com/hypervisor/
- Ammons, G., Appavoo, J., Butrico, M., Da Silva, D., et al 2007. Libra: a library operating system for a jvm in a virtualized execution environment. In VEE07.
- libSPE, http://sourceforge.net/projects/libspe
- Plan 9 from Bell Labs, http://plan9.bell-labs.com/sys/doc/9.html
- Raj, H. and Schwan K. High performance and scalable i/o virtualization via self-virtualized devices. In HPDC '07 (2007).