Xen on PowerPC

Hollis Blanchard, IBM Linux Technology Center
Jimi Xenidis, IBM Research
PowerPC Architecture

- IBM pSeries and blade servers; [Power.org](http://wiki.xensource.com/xenwiki/Xen/PPC)
- exceptions: fixed entry points, real mode
- MMU translation via hash table
- Open Firmware
  - device tree
  - RTAS (Runtime Abstraction Services)
PowerPC Processors

- 32-bit, no hypervisor mode (e.g. “G4” 7450)
- 64-bit, no hypervisor mode (“G5” 970)
  - some prototype work
- 64-bit, hypervisor mode PowerPC 970
  - current Xen/PPC target (Maple evaluation board)
- Cell, POWER5, PWRficient
  - not a current target
IBM Hypervisor Support

- PAPR specification (PowerPC Architecture Platform Requirements)
  - PHYP enterprise hypervisor
- server IO topology
  - IOMMU
  - slot-level PCI error isolation
IBM Processor Extensions

- few OS changes, almost no performance impact
- additional privilege level
  - page table register no longer owned by kernel
- hypervisor decremener
  - to preempt domains even with interrupts off
IBM Processor Extensions 2

- real-mode accesses: base + limit
  - direct-to-domain exception delivery
  - must be contiguous, naturally aligned
Linux Modifications

- PowerPC Linux already well-abSTRACTED
  - page table modifications
  - interrupt controller
- PowerPC Linux changes for Xen
  - event channel
  - early boot console
  - idle loop
Current Xen/PPC Status

- Xen and Linux Domain0 boot
  - bare metal (no bootloader) and systemsim
- custom domain builder
- DomU boots to userspace
- very few Linux changes
  - same binary on PHYP, dom0, domU, hardware
Current Xen/PPC Limitations

- no SMP, domU timekeeping, grant tables, ...
- no virtual IO (not even interactive console)
- fixed-size contiguous domain memory
Roadmap

- finish domU
  - IO drivers
  - SMP
- main bottleneck: tool portability
- testing infrastructure
- software layer to run Xen on JS20 blades
Long-term Roadmap

- 970MP support
- live migration
- non-hypervisor mode 970
- Cell
- non-hypervisor mode 32-bit PPC
Simple Common Changes

- Makefile ifdef (e.g. tools/libxc/Makefile)
  
  - Xen:
    
    ifeq \(\$(CONFIG_FOO),y\)
    SRCS += foo.c
    endif
  
  - Linux:
    
    obj-\(\$(CONFIG_FOO)\) += foo.o
Simple Common Changes 2

- code #ifdefs (e.g. privcmd.c)
  - #if defined(__i386__)
    asm volatile("hypercall")
  - #elif defined(__x86-64__)
  - #elif defined(__ia64__)
    #elif defined(__ia64__)
  - call out to arch-provided function instead

- portable GDB stub – Isaku Yamahata

- unsigned long for bitops
Required Common Changes

- instruction/data cache consistency
  - required when copying code (e.g. domain building)
  - xc_copy_to_domain_page(), xc_map_memcpy()

- timer delivery
  - send_guest_virq(vcpu, VIRQ_TIMER)
  - abstract to arch_send_timer_virq(vcpu)
Tough Common Changes

- management tool address spaces
  - hypervisor has separate address space

one address space -- x86

two address spaces -- PowerPC
Optimizations

- haven't been a priority so far
- non-volatile registers
More Optimizations

- page-flipping
  - must copy real-mode area pages
    - restrict virtual IO to non-real mode pages
  - PowerPC Linux uses large (16MB) pages
    - restrict virtual IO to special 4KB page pool
Discussion Topics

- xencomm area
- merge plans
- community points of interest?