Qemu/Xen integration

Stefano Stabellini
Anthony Perard
Xen Device Model: what is it?

the Xen architecture:
- Xen: an hypervisor designed to run PV domains
- HW assisted virtualization: introducing HVM domains
- the role of the device model
Xen Architecture:
PV domains
Xen Architecture:
HVM domains
IO Emulation: Xen vs Qemu

Xen:
- interrupt injection: pic, apic, ioapic
- timers: hpet, pit, acpi pm timer, rtc

Qemu: everything else
- piix3, piix4, e1000, cirrus vga, ...
Mapcache: why?

- 32 bit PV guests are faster than 64 bit PV guests

- Dom0 is just a management domain, we don't want it to be big

Run Xen 64bit and Dom0 32bit
Stubdoms: why?

Dom0 is just one VM with limited resources

Qemu running in Dom0:
  the cpu and memory usage issue
  the multiple schedulers issue

run Qemu in kernel mode in a separate PV guest
Qemu/Xen integration: v1

The first version of series:

- introduced a new qemu target
- the mapcache was separate from the qemu memory interface
- no ACPI support
- maaany code style issues
The latest version of the series:

- adds an “accelerator” to the i386 target
- the mapcache is integrated with the qemu memory API
- qemu piix4 ACPI implementation is supported
- fewer code style issues :-)

Qemu/Xen integration: v6
Qemu/Xen integration: what is missing?

- Stubdoms: target OS = MiniOS
- QMP support in XL/Libxenlight
- VGA dirty bitmap
- PCI passthrough
Xen, Qemu and KVM: designing common interfaces

Common problems should have a common solution:

- `cpu_physical_memory_read/write`
- `cpu_physical_memory_(un)map`
- `vga_dirty_log_start/stop`
- `cpu_physical_memory_memory_get_dirty`
- PCI passthrough
Xen, Qemu and KVM: cross-Community collaboration

- Xen Community: fix mistakes of the past (forks)

- bring the Communities closer together: cross-posting, collaboration, sharing ideas, ...

- Qemu Community should be directly involved in Xen development

- Xen should be a first class citizen within the Qemu Community
Questions?