



Xen Releases & Roadmap

Overview



- Release schedule
- Last major release (3.1)
- Next major release (3.2)
- Selection of upcoming roadmap features



Release schedule

- Aim for quarterly major releases
 - Actually more like bi-yearly right now
- Regular minor releases during major-release development phase
 - Bug fixes and non-invasive features (3.1.1)
 - Released 'as needed' when enough important fixes are queued up
 - 3.1.2 will be released imminently



Xen 3.1, May 2007

- Lots of HVM fixes, and extra support:
 - Memory ballooning in Linux HVM guests
 - HVM save/restore/relocation
 - Hardware-assisted paging
- Run 32-bit PV guest on 64-bit hypervisor
 - Means that 64-bit Xen is always the sensible choice on 64-bit capable hardware
- Copy-on-write support via blktap
 - QEMU QCOW file format
- And lots more of course...



Xen 3.2, November 2007(?)

- Next major release is in feature freeze
- More HVM advances:
 - Device pass-through
 - Timer modes
- Host S3 (suspend-to-ram) support
- Network acceleration modules
- XSM mandatory access control



Power management

- Frequency/voltage scaling (P states)
- Deep sleeps (C states)
- Detected via ACPI DSDT
 - Xen hypervisor cannot interpret the DSDT
 - Dom0 kernel does not (necessarily) grok the physical CPU topology
 - So plan to do detection in dom0 userspace
- 'Poke' P/C-state availability into hv
 - Use this to inform the idle loop and P-state governor (ported from Linux kernel)



Performance optimizations

- Host scalability (guests per host)
 - Already pretty good but dom0 can be a bottleneck
 - So ongoing efforts to disaggregate dom0
- Guest scalability (vcpus per guest)
 - Locking granularity inside Xen ('big domain lock')
 - Locking protocols inside guests (e.g., spinlocks)
- Network I/O performance
 - NetChannel2, smart NICs
- Super-page mappings in guests
- Scheduler tweaks



HVM device pass-through

- Basic code is in the tree
 - Targets Intel VT-D
 - Also preliminary support for AMD IOMMU
- But pieces are still missing
 - ACPI pass-through
 - Device pass-through without IOMMU?
 - Message Signalled Interrupts
- And testing...

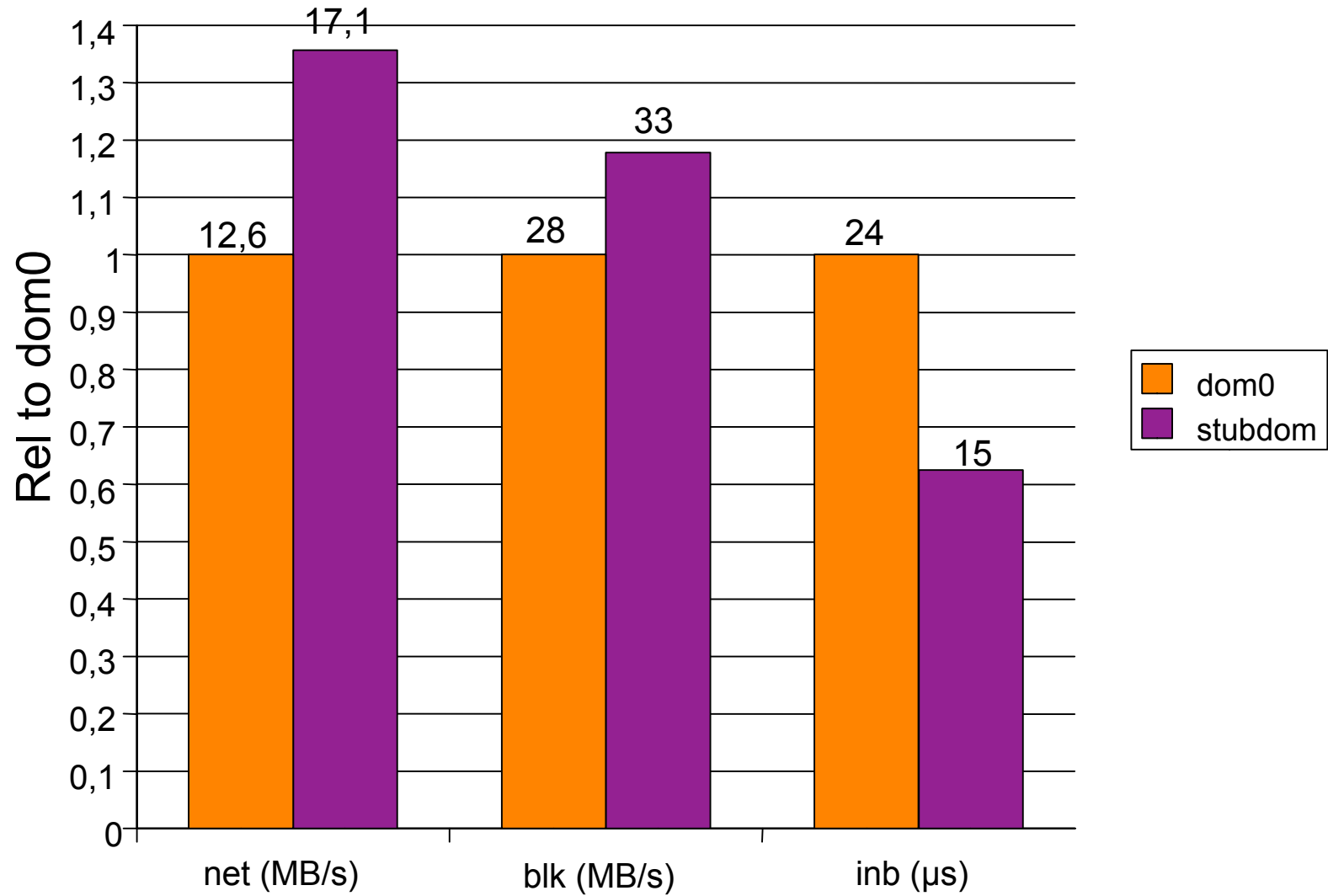


HVM service domains

- Instead of qemu-dm processes in dom0, each HVM guest has its own 'stub domain'
- This is something of a franken-kernel:
 - Device models of qemu
 - PV frontend drivers of Linux
 - Primitive OS mechanisms of minios
- Given permission to access and modify state of only its client HVM guest
- Security, scalability, performance!



Stub domain



Summary

- Lots of ongoing work, and projects to be picked up
 - Need to revise the Xen roadmap to document and track this
- Open collaborative model allows a rapid development pace
- Still lots of fun hacking to be done for 3.3 and beyond!

