



NetApp®

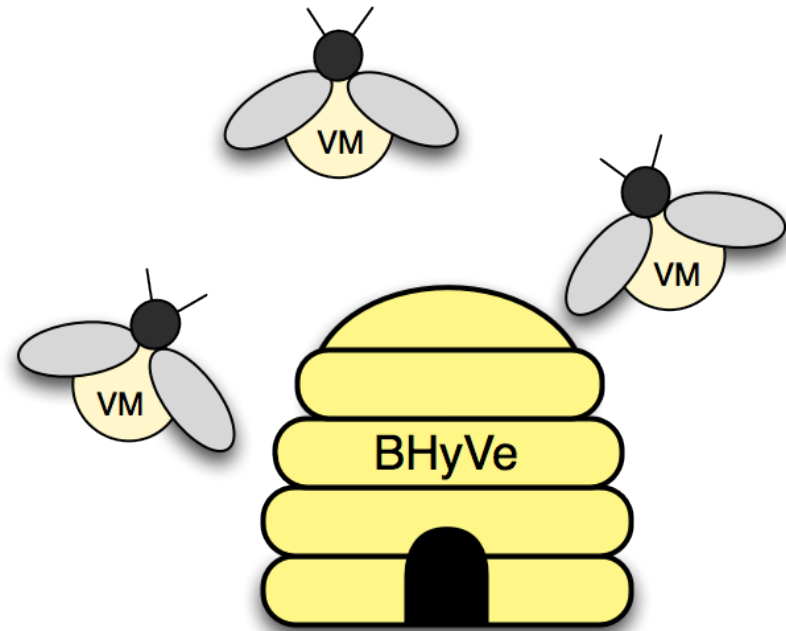
Go further, faster®

BHyVe

BSD Hypervisor

Neel Natu

Peter Grehan





Introduction

- BHyVe stands for “BSD Hypervisor”
 - Pronounced like beehive

- Type 2 Hypervisor (aka hosted hypervisor)
 - FreeBSD is the Host OS

- Availability
 - NetApp is releasing the source code under the BSD license!
 - Snapshot against 8.1 in svn repository: `/projects/bhyve_ref`

- Work In Progress



Status

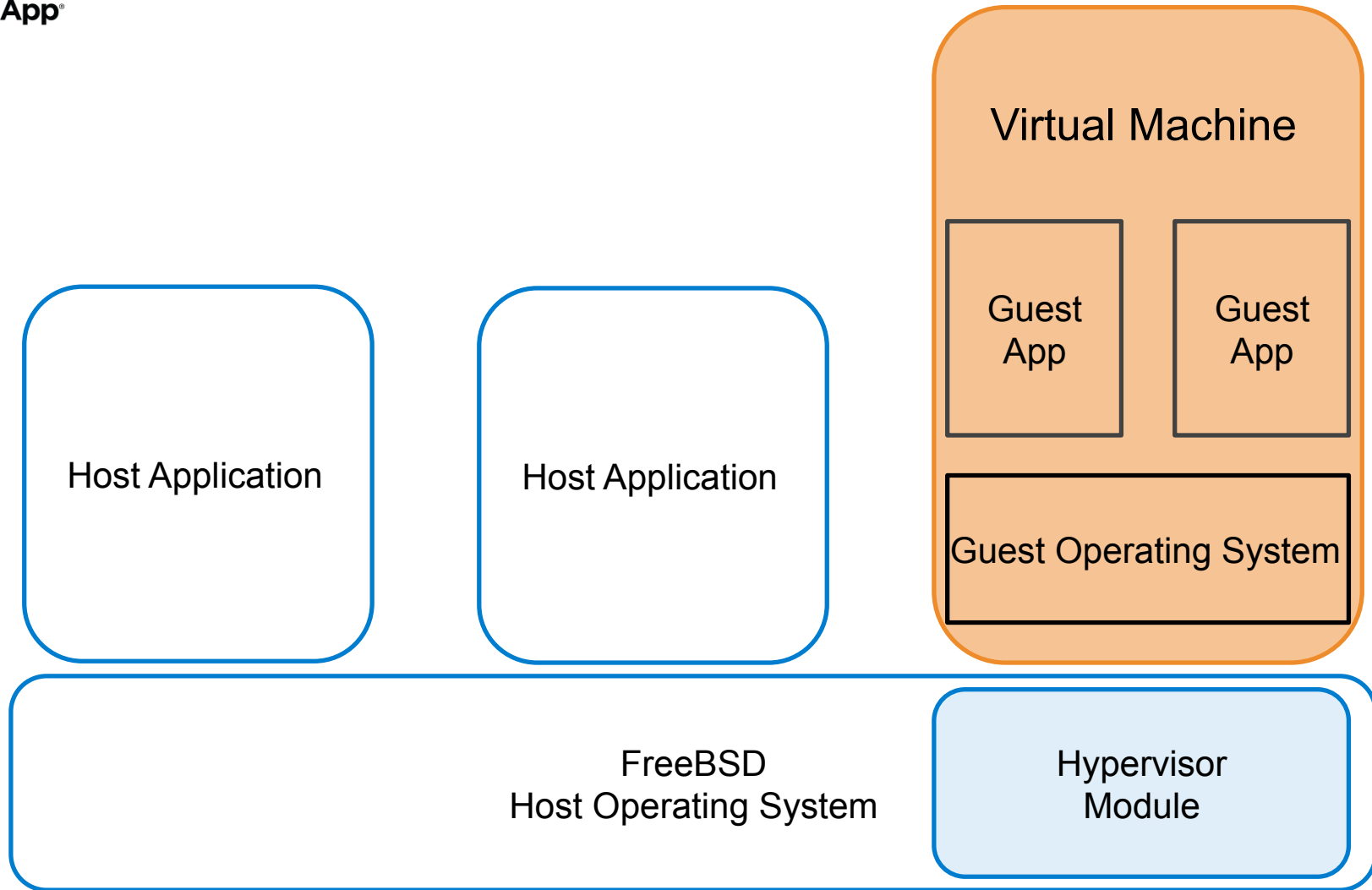
- Guest
 - FreeBSD/amd64 releases 7.2 and 8.1
 - SMP - up to 8 virtual cpus
 - I/O - virtio or pci passthru
 - Minor kernel patches required

- Host
 - FreeBSD/amd64 release 8.1
 - Unmodified GENERIC kernel

- Hardware
 - Requires hardware virtualization assist with Nested Page Tables
 - Intel VT-x is supported
 - AMD-V support in progress

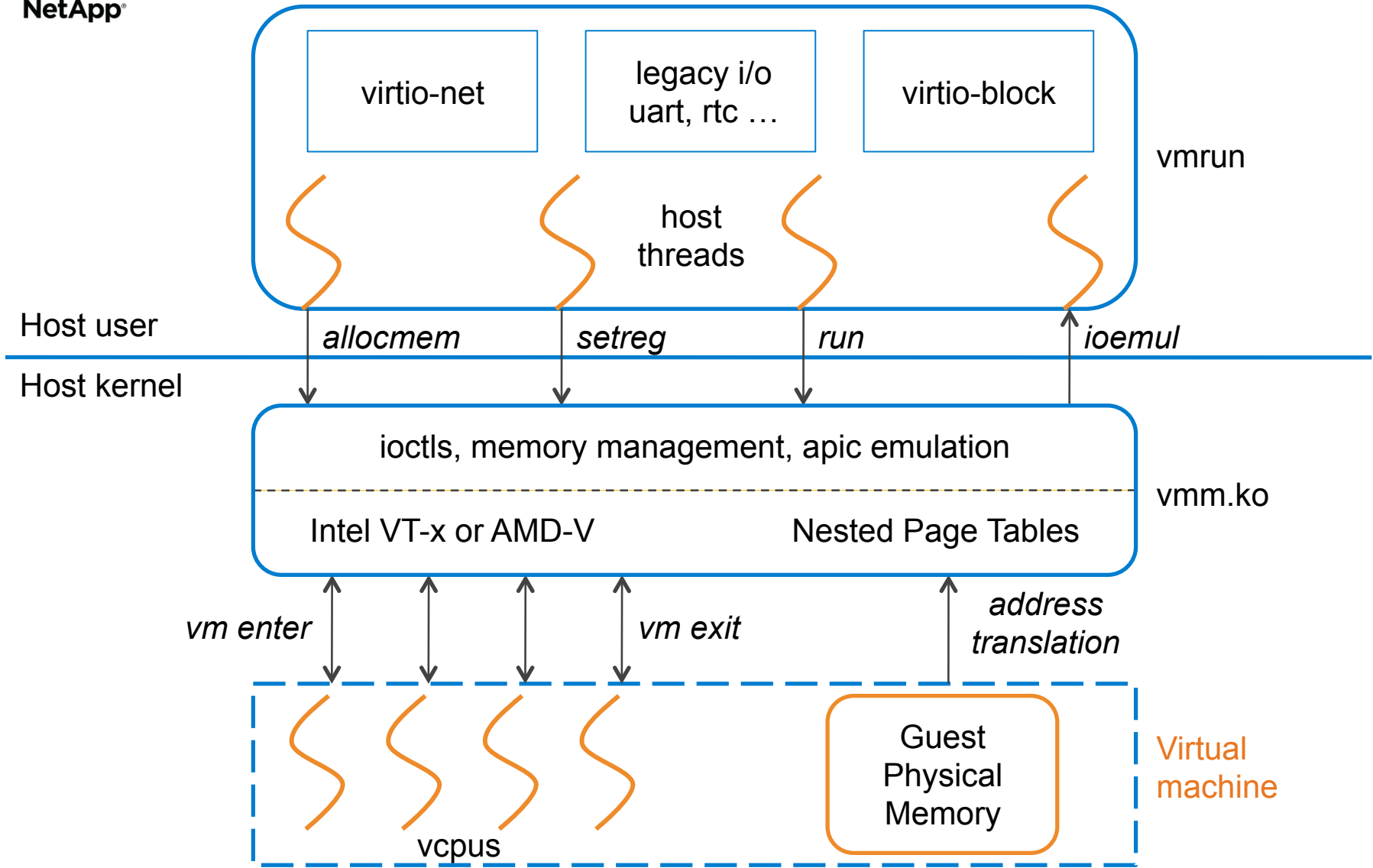


BHyVe: Logical View





BHyVe: Implementation





CPU Virtualization

- Requires Intel VT-x or AMD-V virtualization assists

- Trap into the hypervisor for a variety of reasons
 - Instructions like RDMSR, OUTB, CPUID, HLT, PAUSE
 - Hardware interrupts

- Local APIC is emulated
 - x2APIC mode
 - Accessed by the guest using RDMSR/WRMSR
 - Startup IPI is handled in user-space
 - Creates a thread context for the virtual cpu
 - IPIs between virtual cpus map to a fast host IPI



Memory Virtualization

- Requires hardware support for Nested Page Tables
 - Guest Physical to Host Physical translation

- Memory is allocated and pinned to virtual machines
 - No sharing between virtual machines
 - No allocate-on-demand
 - Hard allocation makes pci passthru a lot easier

- Memory allocated to virtual machines is hidden from the host
 - Kernel config option MAXMEM
 - hw.physmem tunable



PCI I/O Virtualization

- PCI bus topology and configuration emulated in user-space
 - Intercept access to PCI config address and data registers

- Two types of PCI devices on the virtual PCI bus
 - virtio
 - passthru

- Interrupt delivery through MSI only
 - Single as well as multi-vector MSI is supported
 - Legacy is hard because it requires IOAPIC emulation
 - MSI-X is hard because it requires instruction emulation



virtio

- Paravirtualized device specification
 - <http://ozlabs.org/~rusty/virtio-spec/virtio-paper.pdf>

- FreeBSD virtio block and net drivers from deboomerang@gmail.com
 - Not publicly available under a BSD license

- Backend virtio-net and virtio-block devices in user-space
 - virtio-net uses /dev/tapN to send and receive ethernet frames
 - virtio-block reads/writes to a file on the host filesystem



PCI Passthru

- Guest has direct access to a PCI device
- Some configuration registers are still emulated
 - BAR registers
 - MSI capability
- DMA transfers will target guest physical addresses
 - IOMMU translates from guest physical to host physical addresses
- Stub driver in the host forwards interrupts from the device to the guest
- Virtual MSI capability for passthru devices that only support legacy interrupts
- 'blackhole' driver prevents the host from attaching to passthru devices



Guest Modifications

- Custom console and debug port
 - Done for expediency
 - Not necessary if we have a 16550 device model

- Local APIC access via x2APIC MSRs

- AP bringup changed to start execution directly in 64-bit mode
 - Required if real-mode guest execution is not supported



User-space API

- A virtual machine appears in the host filesystem as a device node

- ioctls used to control and configure the virtual machine
 - 20 in total
 - For e.g. setreg, pincpu, run, interrupt, getstats

- Can read(), write() and mmap() the virtual machine device node
 - Useful to inspect the virtual machine's memory
 - `dd if=/dev/vmm/testvm of=memdump bs=1024 count=1024`



Performance

- Features
 - Address space identifiers for virtual cpus
 - Minimal overhead host IPIs
 - Some guest state is lazily saved only on “slow” trap to user-space
 - Guest floating point registers
 - System call related MSR
- “make buildworld”
 - 4 cores, 2GB memory, 1GbE NIC, 1 SATA disk
 - /usr/src is mounted over NFS
 - /usr/obj is mounted on a block device

Configuration	Build time in seconds
Bare Metal	1308
Partitioned	1336
Virtualized	1446



Future Opportunities

- Support Windows, Linux and *BSD guests
- Support AMD's hardware virtualization assist
- Guest suspend/resume and live migration
- BIOS emulation



Thank you

