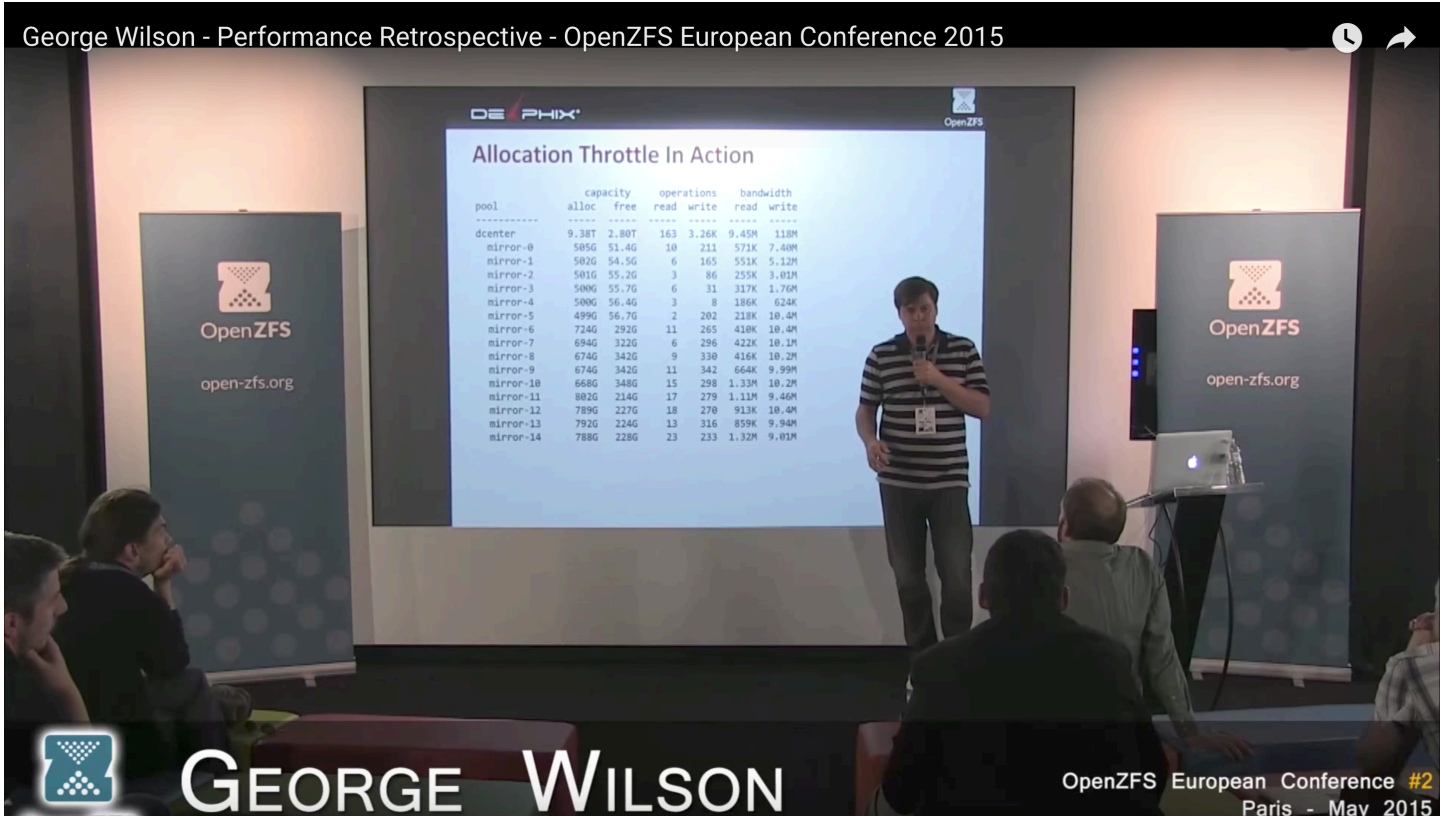
FreeBSD's Influences --

porting Hyper-V to Illumos

George Wilson | gwilson@delphix.com
@zfsdude

About me…

- Illumos Kernel Developer
- 2005 - Joined the ZFS development team
- 2010 - Joined Delphix
- Frequent speaker at OpenZFS Events
  - Metaslab performance improvements
  - Allocation throttle
  - Compressed ARC



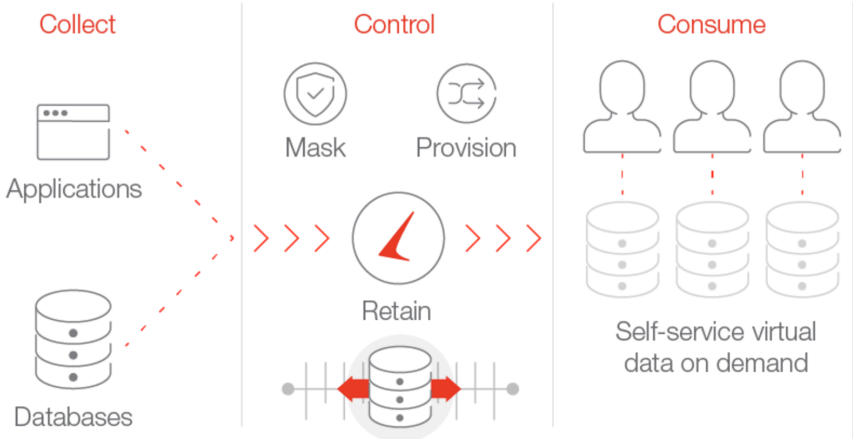
About me…

- Illumos Kernel Developer
- 2005 - Joined the ZFS development team
- 2010 - Joined Delphix
- Frequent speaker at OpenZFS Events
  - Metaslab performance improvements
  - Allocation throttle
  - Compressed ARC
- First time speaker at FreeBSD event

My Journey to Azure

What does Delphix do?

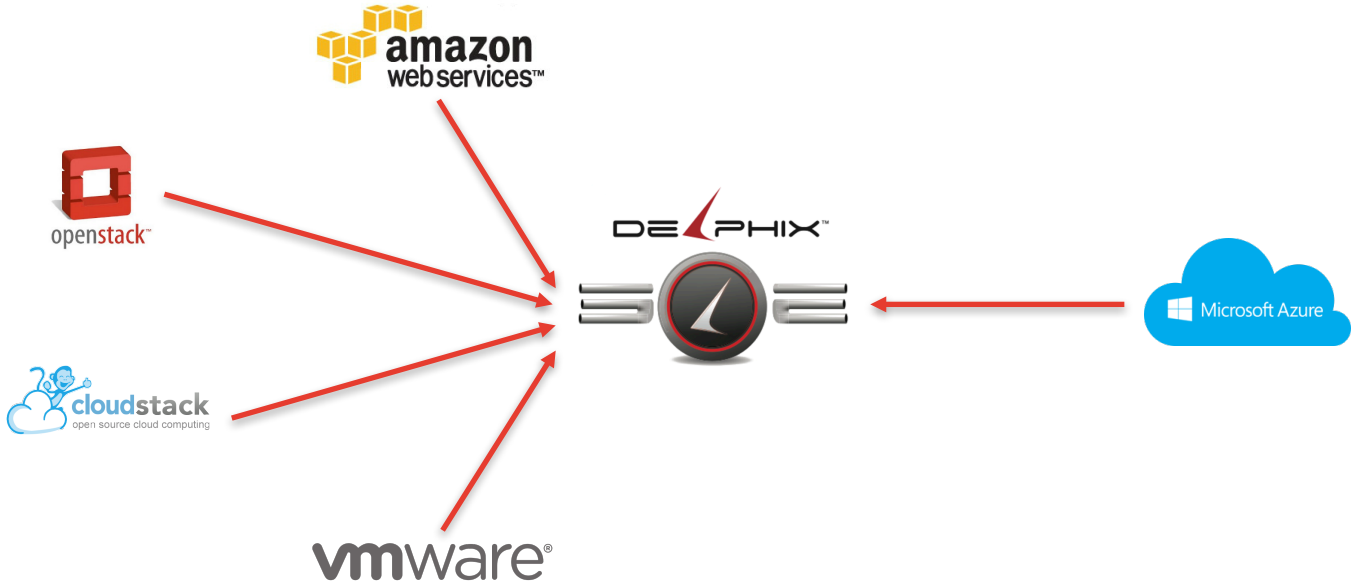
DevOps Automation



Cloud Migration

A long, long time ago…

- Delphix wanted to expand its platform offering




How did I get here?

Perfect timing

Source: FreeBSD Developer Summit 2016

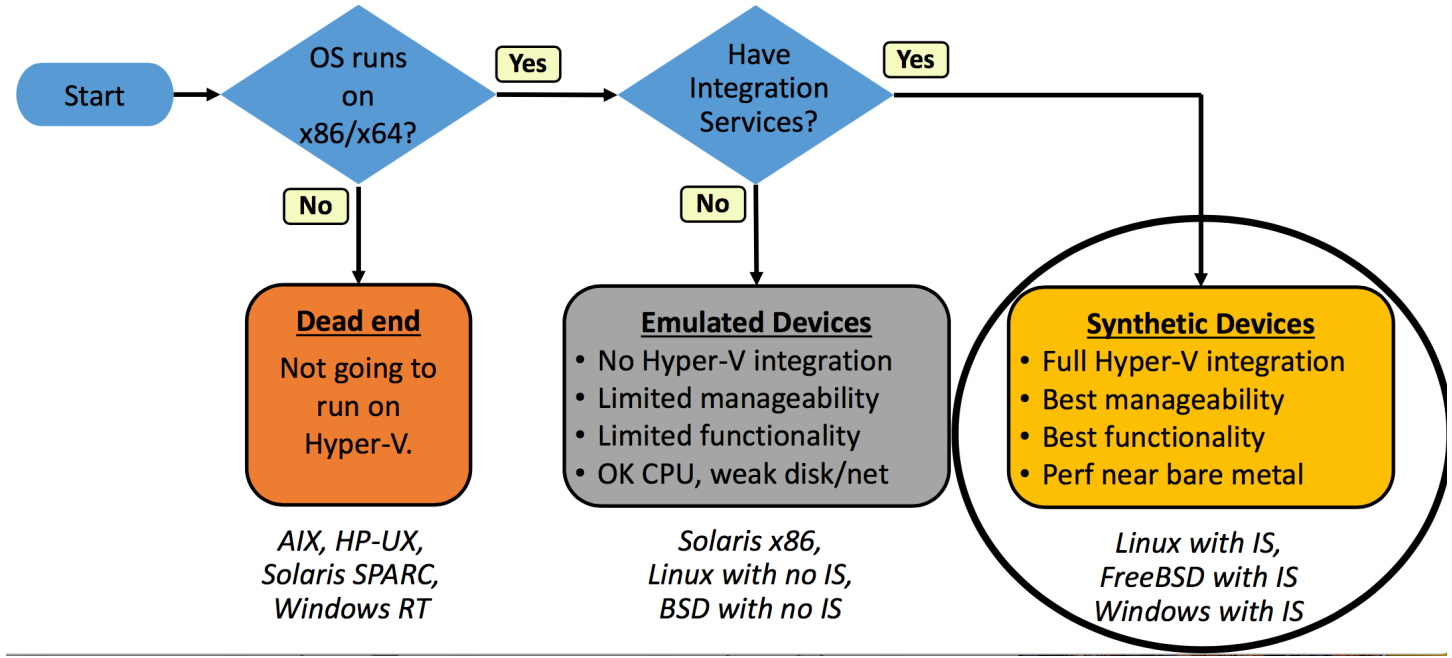
Transitive Property of Love

Then this must be true…

Then this must be true…

Yet!

Is it possible?



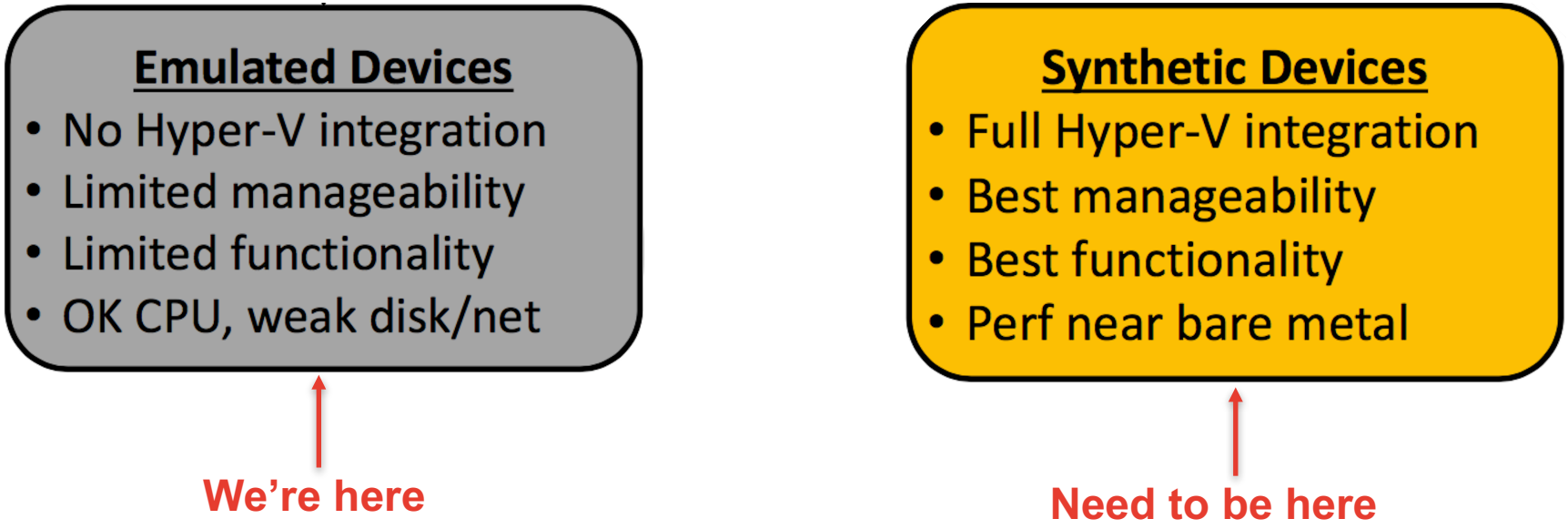
Source: FreeBSD Developer Summit 2016

Emulated Networking in Hyper-V

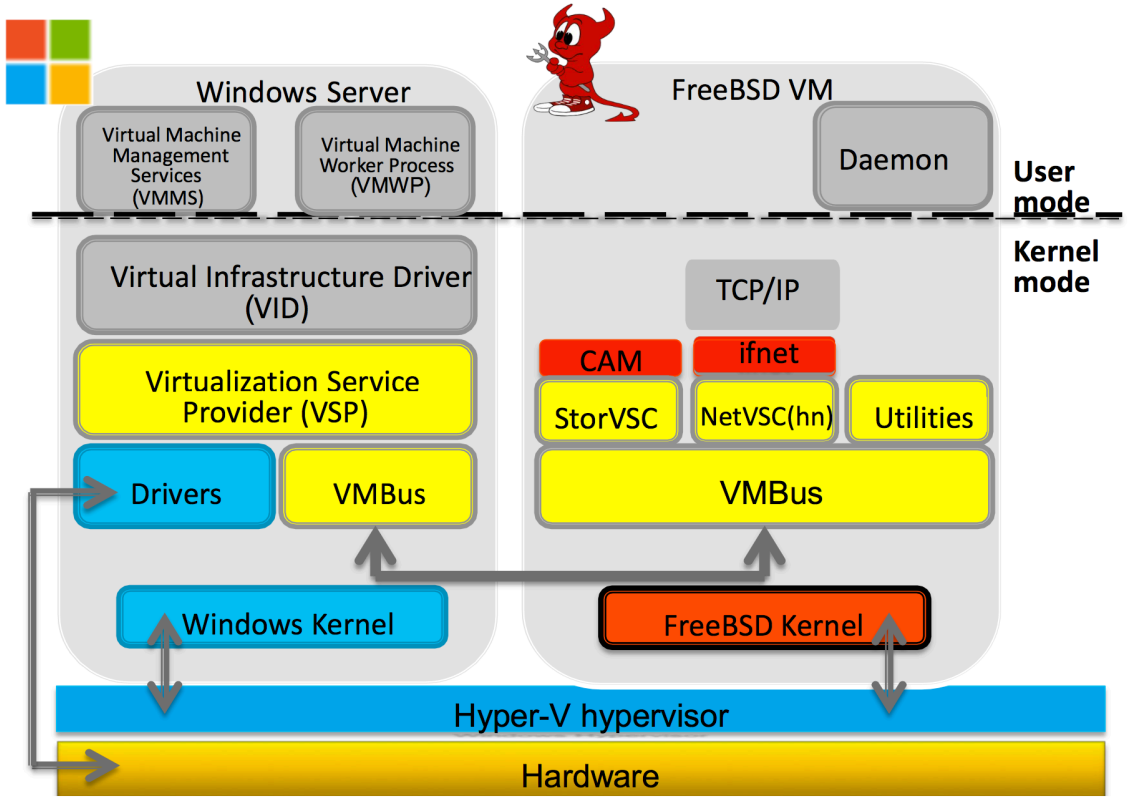
- DECnet Driver
  - Network protocol suite developed 1975
  - Solaris driver released in 2000
  - 100 Mbps (half duplex)

Source: By Flightsoffancy at English Wikipedia, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=9891488

Our Need…



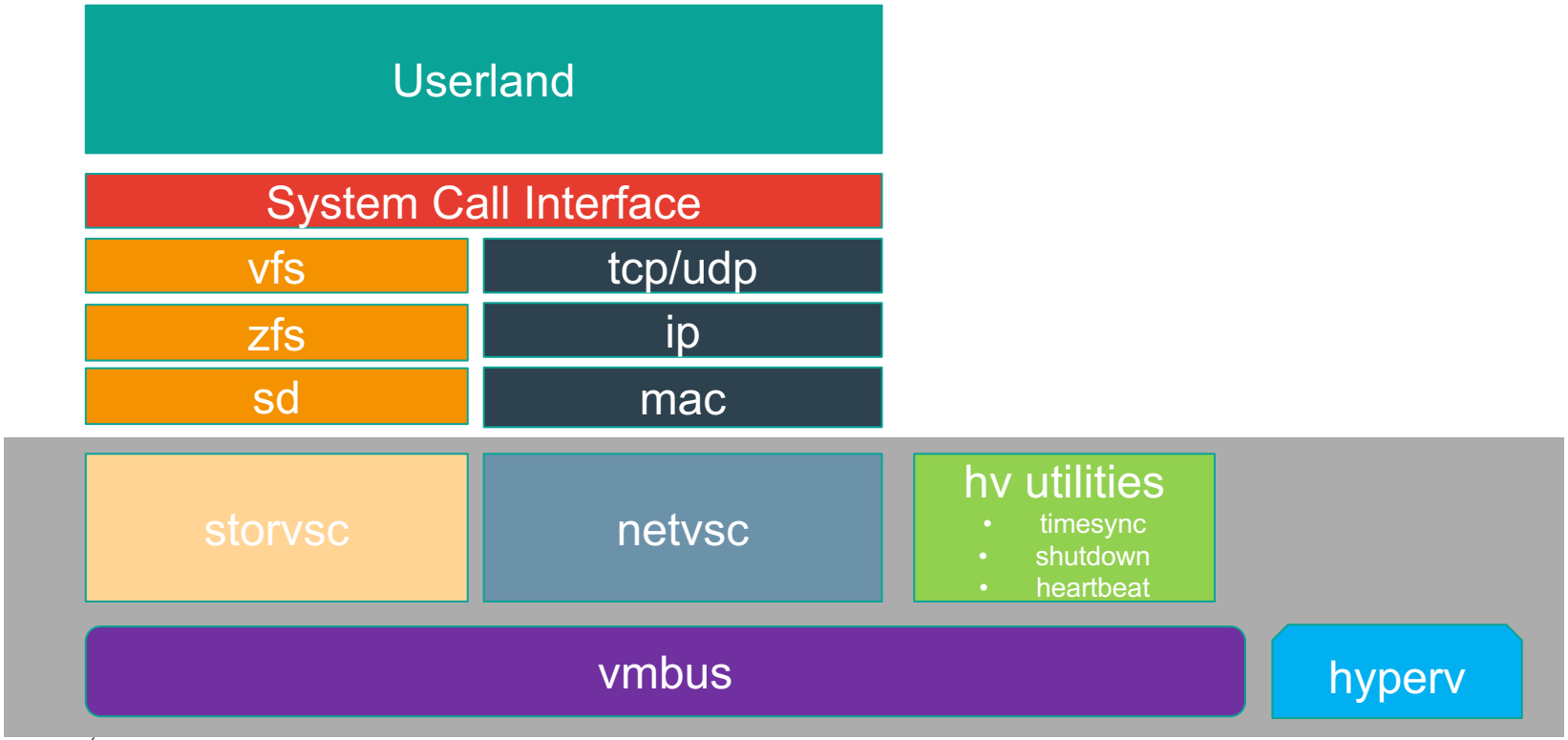
Hyper-V Overview



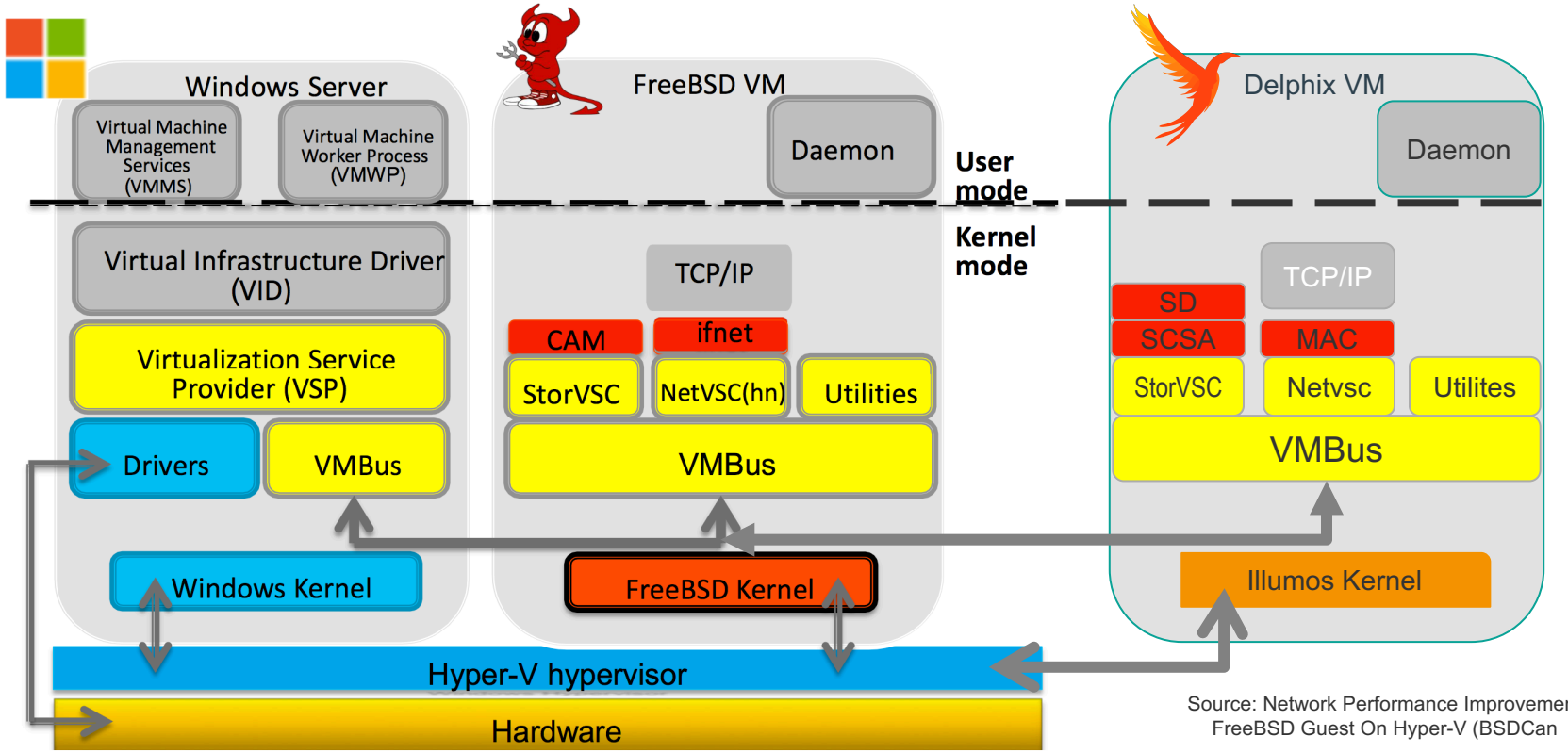
Source: Network Performance Improvements for FreeBSD Guest On Hyper-V (BSDCan 2016)

FreeBSD

Just need some drivers



Illumos Hyper-V



Source: Network Performance Improvements for FreeBSD Guest On Hyper-V (BSDCan 2016)

Porting Hyper-V drivers

Differences

- Small differences in kernel synchronization primitives
  - sema_wait() vs sema_p()
  - struct mtx vs kmutex_t
- Larger changes needed when porting subsystem interactions
  - CAM
  - sysctl
  - ifnet

Porting Guide

DMA (note: translations are approximate only)	bus_dma_tag_t	ddi_dma_handle_t (also contains info related to ddi_dma_attr_t) dev_info_t * (I have found this is easier to pass around)
	bus_dmamap_t	ddi_dma_handle_t
	bus_addr_t	paddr_t
	bus_dma_tag_create() create a bus_dma_tag_t (dma handle), that contains DMA constraints info.	ddi_dma_alloc_handle() create a handle used for dma operations. A ddi_dma_attr_t structure must be passed with the DMA constraints.
	bus_dmamem_alloc() allocate DMA memory tied to a bus_dma_tag_t	ddi_dma_mem_alloc() allocate DMA memory tied to a ddi_dma_handle_t
	bus_dmamap_load() get the physical address(es) of allocated memory. Note: if the address doesn't meet the constraints of the dma tag, a copy of the memory could be made. bus_dmamem_unload() must be called before when DMA transfer is done.	ddi_dma_addr_bind_handle() get the physical address(es) of allocated memory. The first address is contained in the cookie. Get the next addresses by calling ddi_dma_nextcookie(). Note: copying of memory might be done if original address doesn't meet DMA constraints.
	bus_dmamap_unload()	ddi_dma_addr_unbind_handle()
	bus_dmamap_create() create a new dma map that is required when using previously allocated kernel memory for DMA purposes. The map is used by functions such as bus_dmamap_load_mbuf_sg().	N/A The dma map is already contained inside a dma handle.

Our Approach

High Level Goals

- Develop all drivers on Hyper-V
- Try to keep logic and code as similar to FreeBSD
- Periodically sync with upstream
- Panic, hang, and panic some more

Strategy

- Get hypercalls working
- Port hv_vmbus driver
- Port simple utility driver and attach to bus
- Deal with the hard stuff
  - Storage
  - Networking
  - Device discovery

First contact (Aug 2nd @ 4:07pm)

- hyperv driver
  - Provides hypercall glue
  - Sets up DMA

```
Aug  2 18:07:16 delphix-pks hyperv: [ID 886709 kern.notice] Hyper-V Version: 6.3.9600 [SP17]
Aug  2 18:07:16 delphix-pks hyperv: [ID 658633 kern.notice]
Features=0xe7f<VPRUNTIME,TMREFCNT,SYNIC,SYNTM,APIC,HYPERCALL,VPINDEX,REFTSC,IDLE,TMFREQ>
Aug  2 18:07:16 delphix-pks hyperv: [ID 832265 kern.notice]   PM Features=0x0 [C2]
Aug  2 18:07:16 delphix-pks hyperv: [ID 808869 kern.notice]
Features3=0x17b2<DEBUG,XMMHC,IDLE,NUMA,TMFREQ,SYNCMC,CRASH,NPIEP>
Aug  2 18:07:16 delphix-pks hyperv: [ID 689518 kern.notice] NOTICE: eax 1073741828, regs[0]=44, regs[1]=4095,
regs[2]=0, regs[3]=0
Aug  2 18:07:16 delphix-pks hyperv: [ID 601884 kern.notice] NOTICE:   Recommends: 0000002c 00000fff
Aug  2 18:07:16 delphix-pks hyperv: [ID 689518 kern.notice] NOTICE: eax 1073741829, regs[0]=64, regs[1]=512,
regs[2]=17600, regs[3]=0
Aug  2 18:07:16 delphix-pks hyperv: [ID 165917 kern.notice] NOTICE:   Limits: Vcpu:64 Lcpu:512 Int:17600
Aug  2 18:07:16 delphix-pks hyperv: [ID 689518 kern.notice] NOTICE: eax 1073741830, regs[0]=15, regs[1]=0,
regs[2]=0, regs[3]=0
Aug  2 18:07:16 delphix-pks hyperv: [ID 239943 kern.notice] NOTICE:   HW Features: 0000000f, AMD: 00000000
```

hv_vmbus

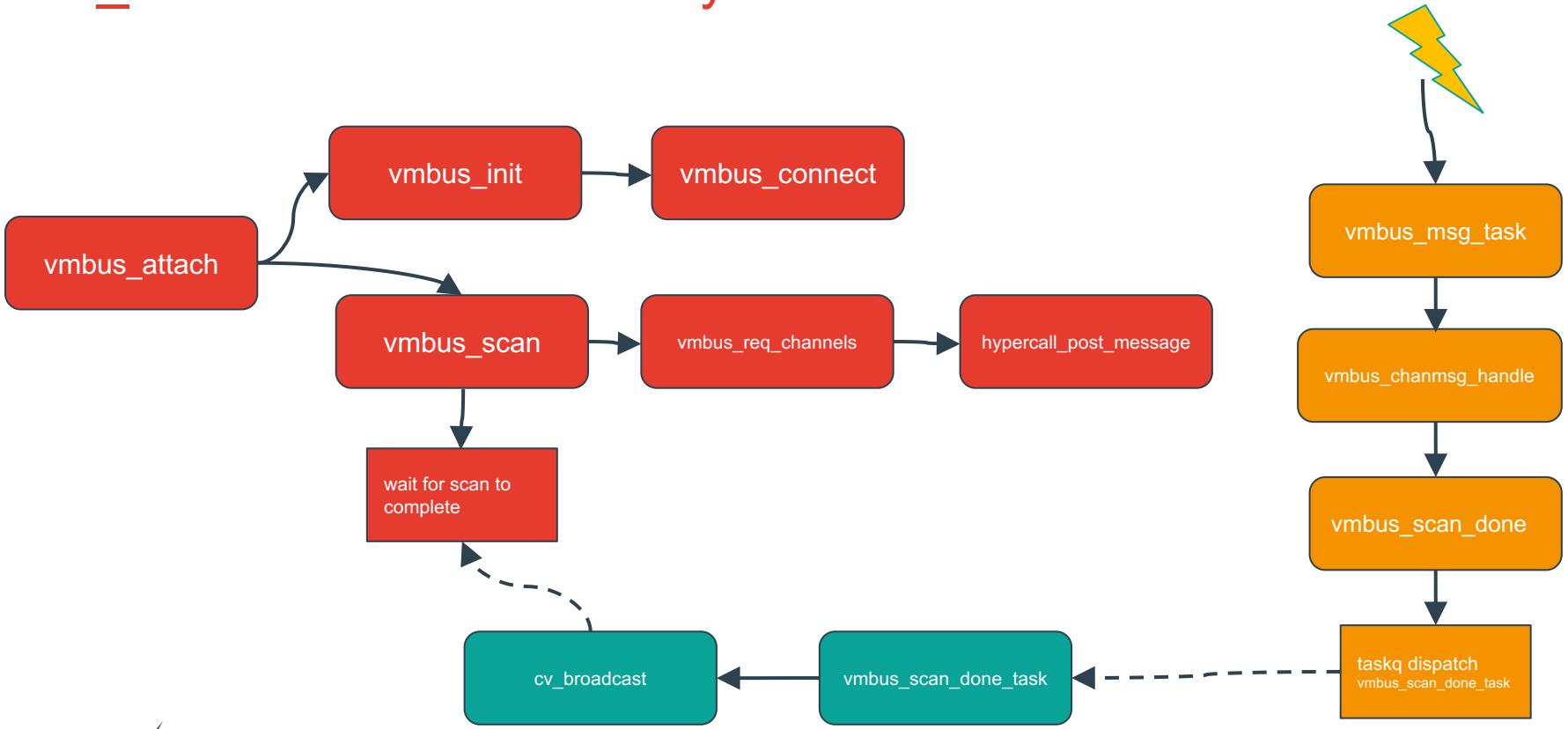
- Bus driver
  - Manages communication with the host over channels
  - Responsible for adding and destroying channels
  - All synthetic drivers attach to the bus driver
  - An open channel represents a device on the guest VM
- Developing a bus driver – use the documentation

  “A **bus nexus device** provides bus mapping and translation services to subordinate devices in the device tree. PCI - PCI bridges, PCMCIA adapters, and SCSI HBAs are all examples of nexus devices. The discussion of writing drivers for nexus devices is limited to the development of SCSI HBA drivers (see Chapter 18, SCSI Host Bus Adapter Drivers).” – **Writing Device Driver Guide**

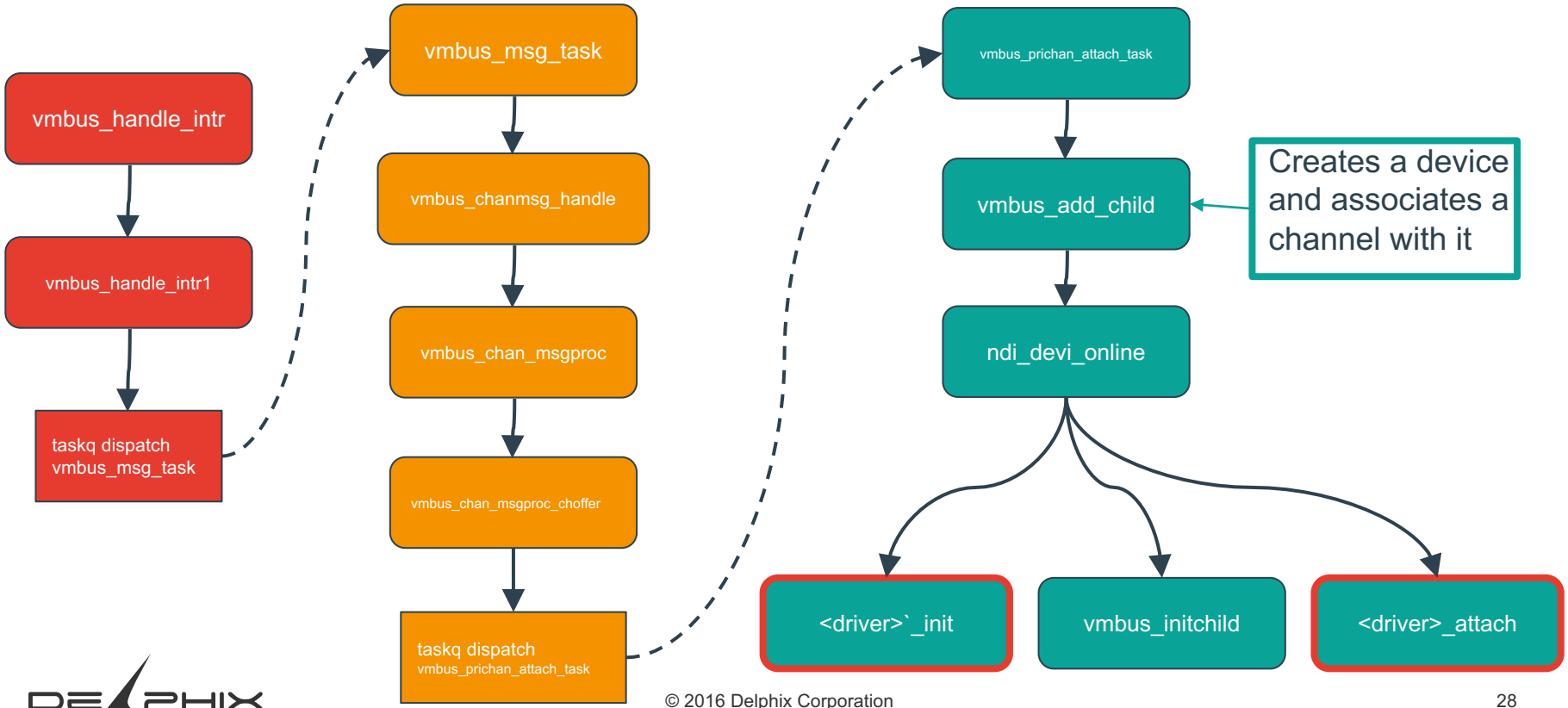
hv_vmbus

- Typical Illumos leaf devices will attach to a bus driver when they load
  - Auto device discovery → Driver loads → Bus Driver → Creates a device → Driver attaches
  - Creating devices and calling the module's attach/detach routines are all handled automatically
- Problems:
  1. The hv_vmbus driver doesn't follow the typical driver attach mechanism
  2. No documentation about how to write a bus driver

hv_vmbus device discovery



hv_vmbus device addition



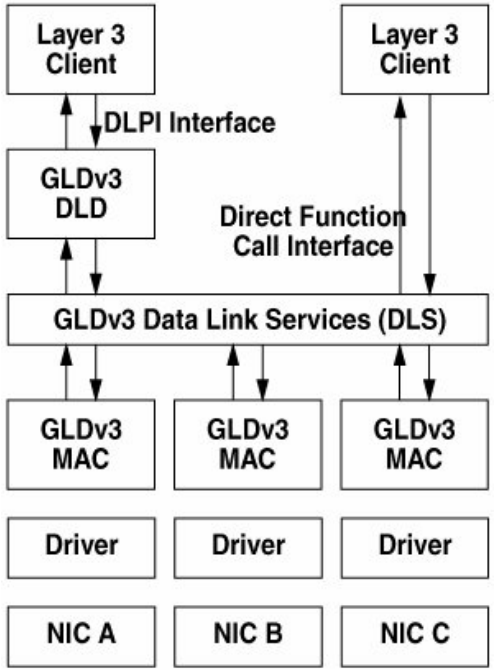
hv_vmbus gets a device (Oct 26th)

- We used `hv_heartbeat` as a simple driver to attach to `hv_vmbus`

```
Oct 26 04:20:16 delphix-gwilson hv_utils: [ID 874935 kern.notice] NOTICE: hv_heartbeat_init: init done, error: 0
Oct 26 04:20:16 delphix-gwilson hv_vmbus: [ID 141394 kern.warning] WARNING: configuring hv_utils
Oct 26 04:20:16 delphix-gwilson hv_vmbus: [ID 395608 kern.info] hv_vmbus@, hv_vmbus0
Oct 26 04:20:16 delphix-gwilson hv_vmbus: [ID 558166 kern.warning] WARNING: hv_vmbus0: INIT CHILD parent
ffffff03e6bcc360 child ffffff03e6bcf598
Oct 26 04:20:16 delphix-gwilson hv_vmbus: [ID 395608 kern.info] hv_utils@<null string>, hv_utils0
Oct 26 04:20:23 delphix-gwilson hv_vmbus: [ID 395608 kern.info] hv_utils@<null string>, hv_utils0
Oct 26 04:20:24 delphix-gwilson hv_vmbus: [ID 702116 kern.notice] NOTICE: vmbus_fini: fini done, error 16
Oct 26 04:20:24 delphix-gwilson hv_utils: [ID 919949 kern.notice] NOTICE: hv_heartbeat_fini: fini done, error: 0
Oct 26 04:20:24 delphix-gwilson hv_vmbus: [ID 702116 kern.notice] NOTICE: vmbus_fini: fini done, error 16
```

netvsc

- Network Virtualization Client
  - Synthetic network adapter
  - NDIS protocol to communicate with host
- Port from ifnet to GLDv3



netvsc (Nov 8th)

- Port ifnet interfaces to GLDv3 framework

```
ffffff03e6b29018 hv_vmbus, instance #0 (driver name: hv_vmbus)
        ffffff03e6b2a098 classid=32412632-86cb-44a2-9b5c-50d1417 (driver not attached)
        ffffff0405bfa670 classid=f8e65716-3cb3-4a06-9a60-1889c5c (driver not attached)
        ffffff0405bfa3b0 classid=cfa8b69e-5b4a-4cc0-b98b-8ba1a1f (driver not attached)
        ffffff0405bfa0f0 classid=f912ad6d-2b17-48ea-bd65-f927a61 (driver not attached)
        ffffff0405bf9e30 classid=da0a7802-e377-4aac-8e77-0558eb1 (driver not attached)
        ffffff0405bf9b70 classid=3375baf4-9e15-4b30-b765-67acb10 (driver not attached)
        ffffff0405bf98b0 classid=57164f39-9115-4e78-ab55-382f3bd (driver not attached)
        ffffff0405bf95f0 classid=a9a0f4e7-5a45-4d96-b827-8a841e8 (driver not attached)
        ffffff0405bfa930 classid=0e0b6031-5213-4934-818b-38d90ce (driver not attached)
        ffffff0405bf9330 classid=9527e630-d0ae-497b-adce-e80ab01 (driver not attached)
        ffffff0405bf9070 classid=35fa2e29-ea23-4236-96ae-3a6ebac (driver not attached)
        ffffff040dedebf8 classid=276aacf4-ac15-426c-98dd-7521ad3 (driver not attached)
        ffffff040dede938 classid=f8615163-df3e-46c5-913f-f2d2f96 (driver not attached)
        ffffff03e6b1a398 netvsc, instance #0 (driver name: netvsc)
```

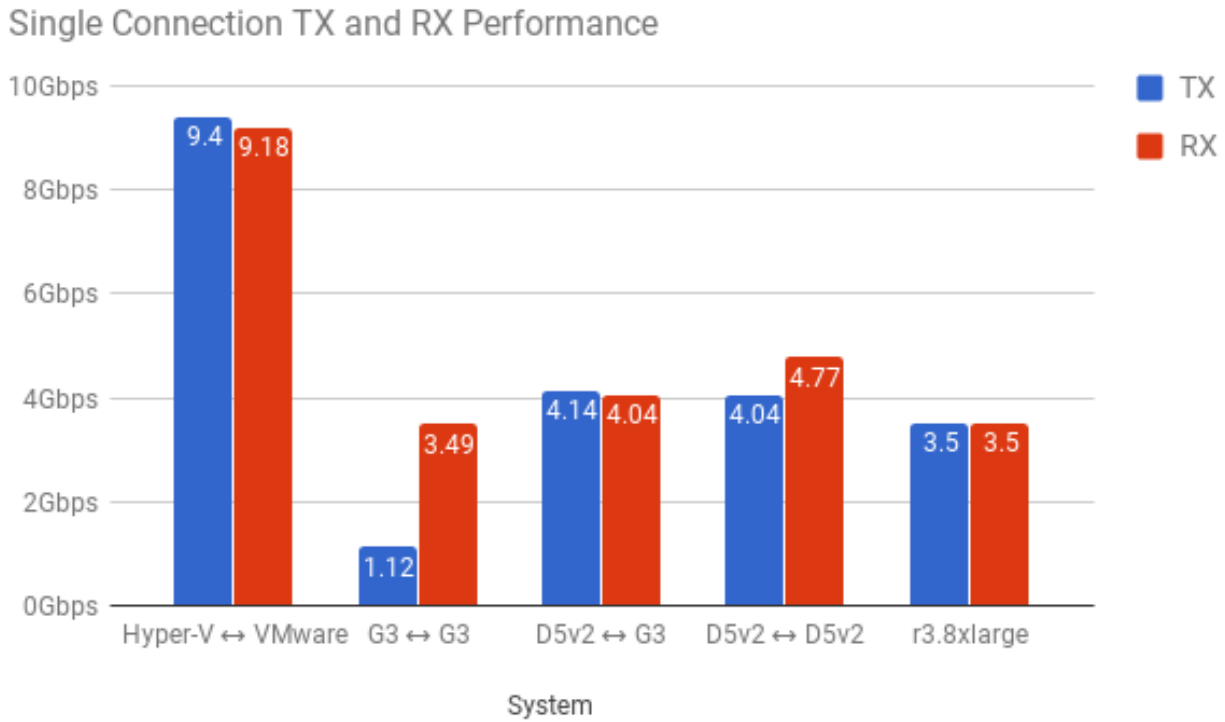
```
delphix@delphix-pzakha:/export/home/delphix$ dladm show-phys
LINK         MEDIA                STATE      SPEED  DUPLEX    DEVICE
dnet0        Ethernet             unknown    0      half      dnet0
netvsc0      Ethernet             up         0      full      netvsc0
```

Dealing with multiple instances

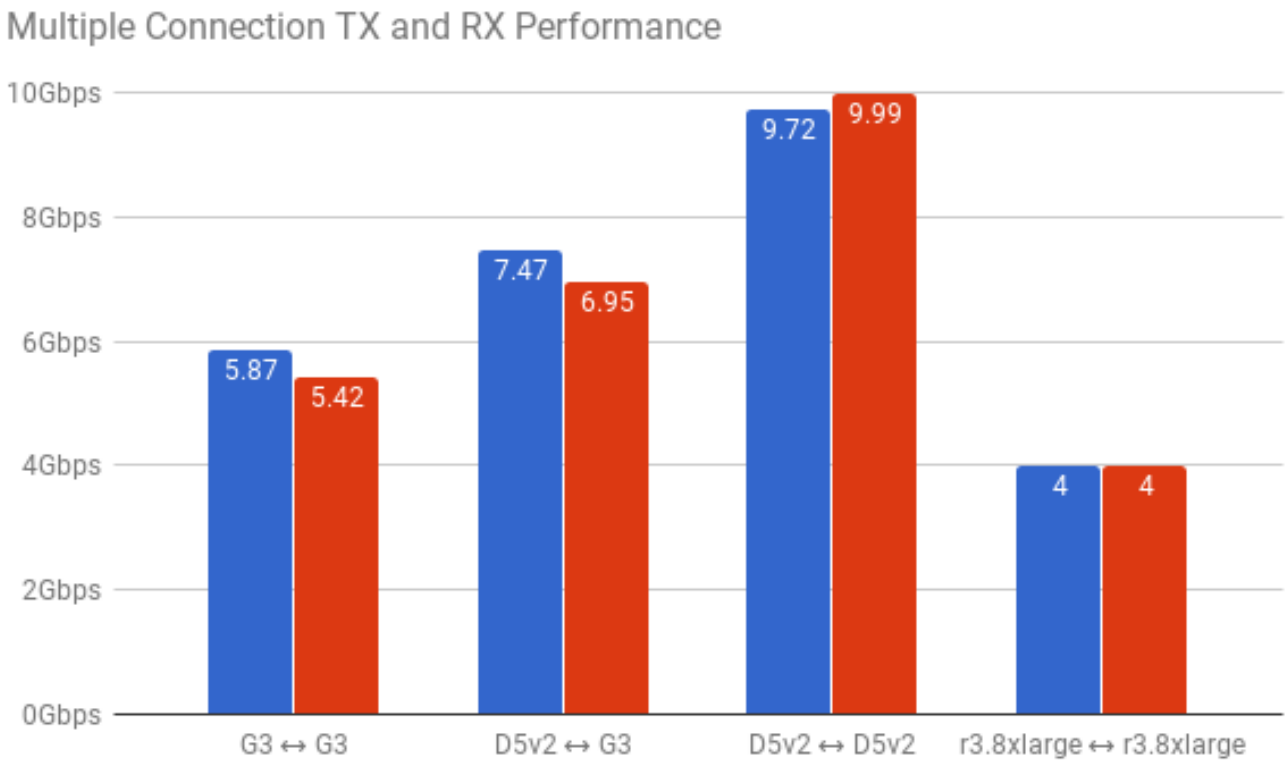
- Needed a unique way to track multiple instances of the same device
- When a device is added, we associated a channel with the device
- Use the channel information to provide unique instances

```
struct vmbus_channel {
<snip>
    struct hyperv_guid ch_guid_type;
    struct hyperv_guid ch_guid_inst;
<snip>
}
```

Netvsc Performance (1 connection)

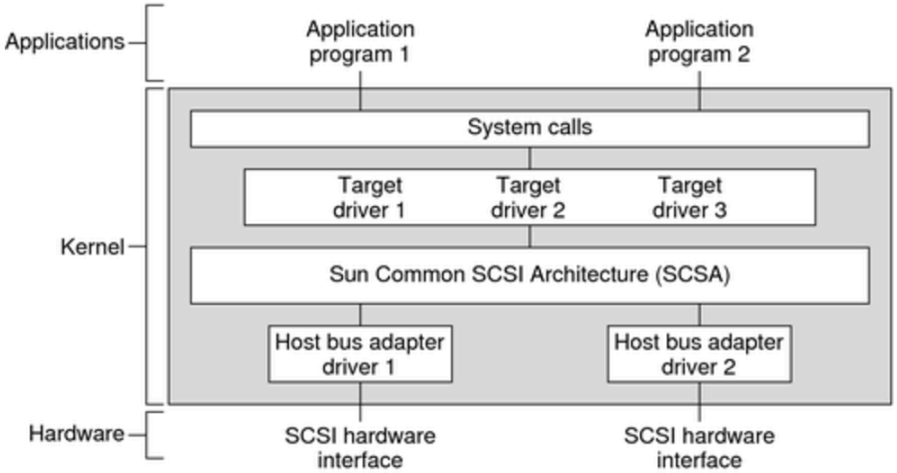


Netvsc Performance (4 connections)



hv_storvsc

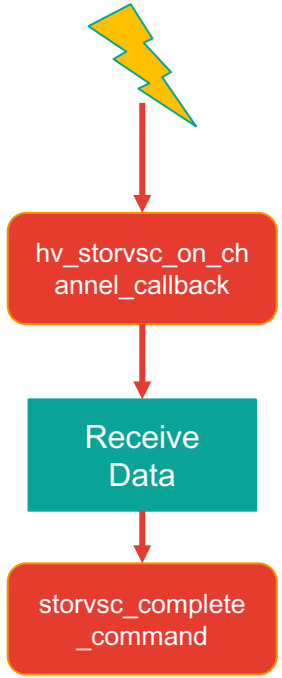
- Storage Virtualization Client
  - Sends and receives VSCSI packets through the Hyper-V vmbus
  - Implements virtual HBA
- Port CAM to SCSA (Sun Common SCSI Architecture)
  - Allow common tools like `format` to work with `storvsc`



Source: https://docs.oracle.com/cd/E26502_01/html/E29051/scsi-3.html

Issuing I/O


```
hv_storvsc_io_request sc ffffff03f4c74000 sending gpa_len 8192 prp_cnt 2,
ch_sel 0
hv_storvsc_io_request submitted, cmd 0x8, pkt ffffff042ca5a620, timeout 60,
target 0, lun 0
hv_storvsc_on_channel_callback req ffffff042ca5a380 pkt ffffff042ca5a620
storvsc_complete_command completed I/O, cmd: 0x8, pkt ffffff042ca5a620,
pkt_resid 0
storvsc_complete_command: calling ffffffffff8178a00, cmd 0x8
```



Hanging I/O

- DMA is great until it break
  - If only we had Dtrace for Windows
  - Commands are issued to hypervisor but never return

```
storvsc_init_pkt bp bcount 8192, pkt_resid 0
storvsc_init_pkt cmd_cookiec 1 cmdlen 6, gpa_len 8192
hv_storvsc_io_request sc ffffff03f4c74000 sending gpa_len 8192 prp_cnt 1, ch_sel 1
hv_storvsc_io_request submitted, cmd 0x8, pkt ffffff042ca50fb0, timeout 60, target 0, lun
0
```

hv_storvsc (Jan 11th first milestone)

- Successful I/O with multiple block sizes
- Able to see the device in `format`

```
AVAILABLE DISK SELECTIONS:
0. c3d0 <Unknown-Unknown-0001 cyl 3129 alt 2 hd 255 sec 63>          /pci@0,0/pci-
ide@7,1/ide@0/cmdk@0,0
1. c5t0d0 <Msft-Virtual Disk-1.0-10.00GB>          /hv_vmbus/hv_storvsc@504610ba-c5fb-
48ce-9602-e2b516a1898d/disk@0,0
Specify disk (enter its number)
```

```
root@delphix:/home/gwilson# dd if=/dev/rdsk/c5t0d0p0 of=/dev/null bs=4k count=10000
10000+0 records in
10000+0 records out
40960000 bytes (41 MB) copied, 2.17833 s, 18.8 MB/s
```

hv_storvsc (Jan 18th)

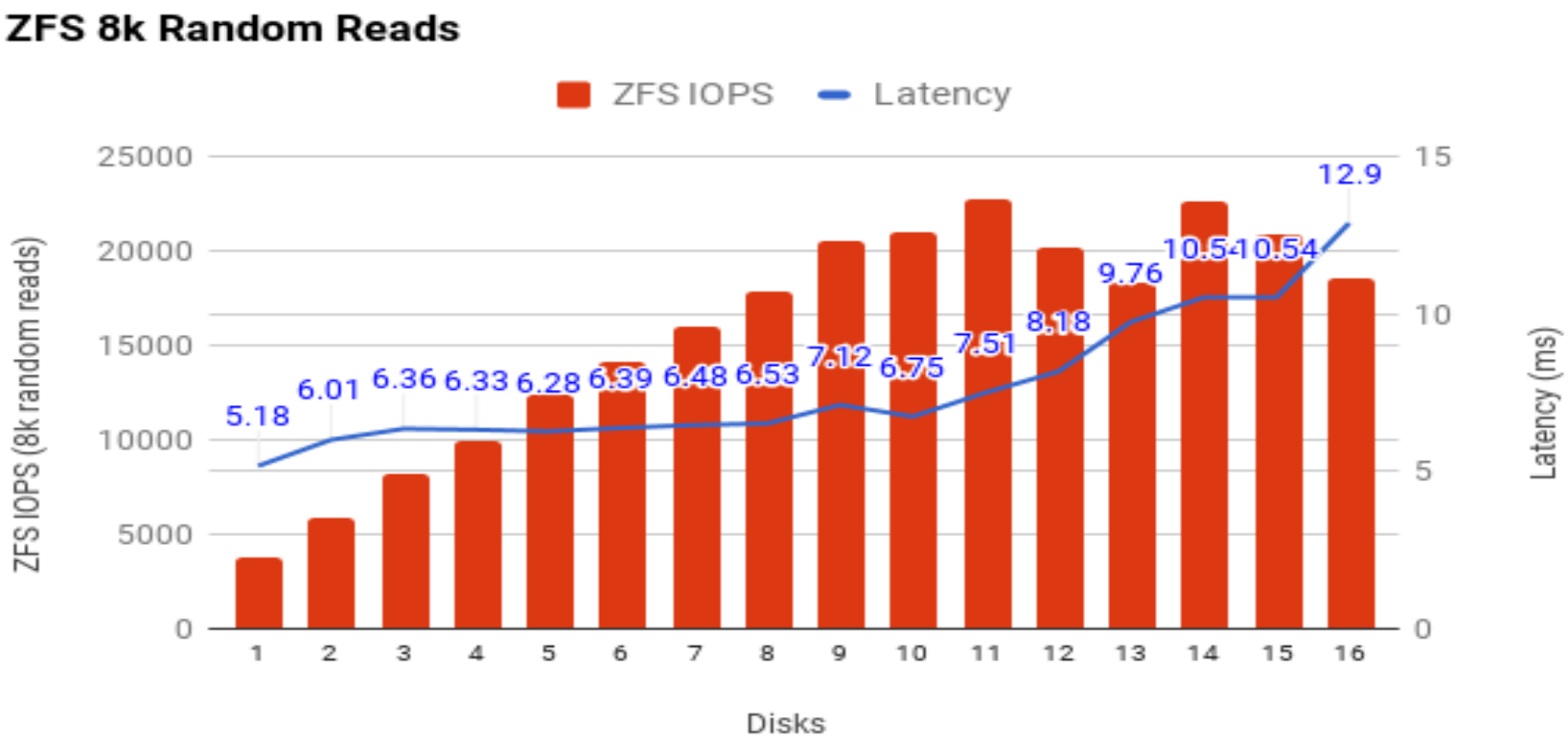
- Able to see multiple targets

```
root@dlpx-debug:/var/crash# cfgadm -al
Ap_Id                          Type         Receptacle   Occupant     Condition
c1                             scsi-sas     connected    configured   unknown
c1::dsk/c1t0d0                 disk         connected    configured   unknown
c1::dsk/c1t0d2                 disk         connected    configured   unknown
c2                             scsi-sas     connected    configured   unknown
c2::dsk/c2t0d1                 disk         connected    configured   unknown
```

- Unique instances

```
"/hv_vmbus/storvsc@d6ed68a2-92de-45e1-bdbb-ff1e12cd4f0e/disk@0,0" 1 "sd”
"/hv_vmbus/storvsc@d6ed68a2-92de-45e1-bdbb-ff1e12cd4f0e/disk@0,2" 2 "sd”
"/hv_vmbus/storvsc@1f303630-5f66-4a87-b88b-a12161fc8ee2/disk@0,1" 3 "sd"
```

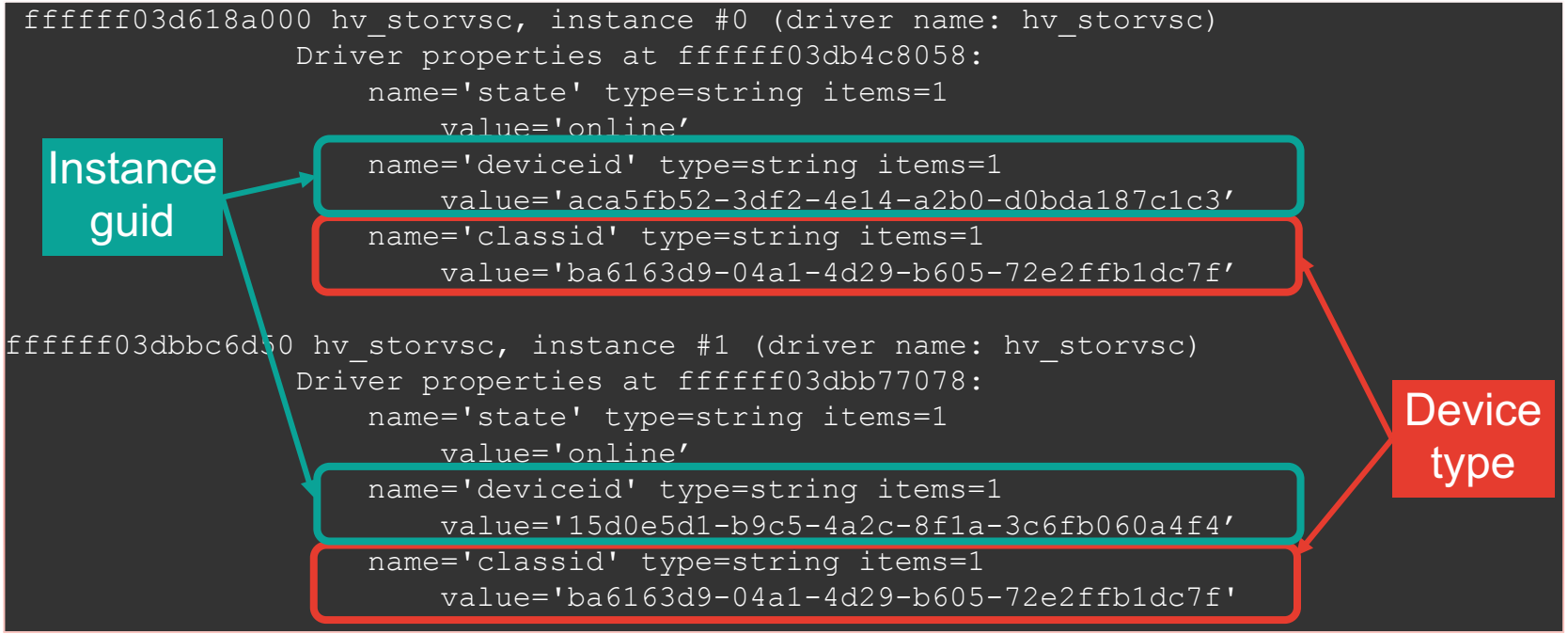
Scaling on Azure



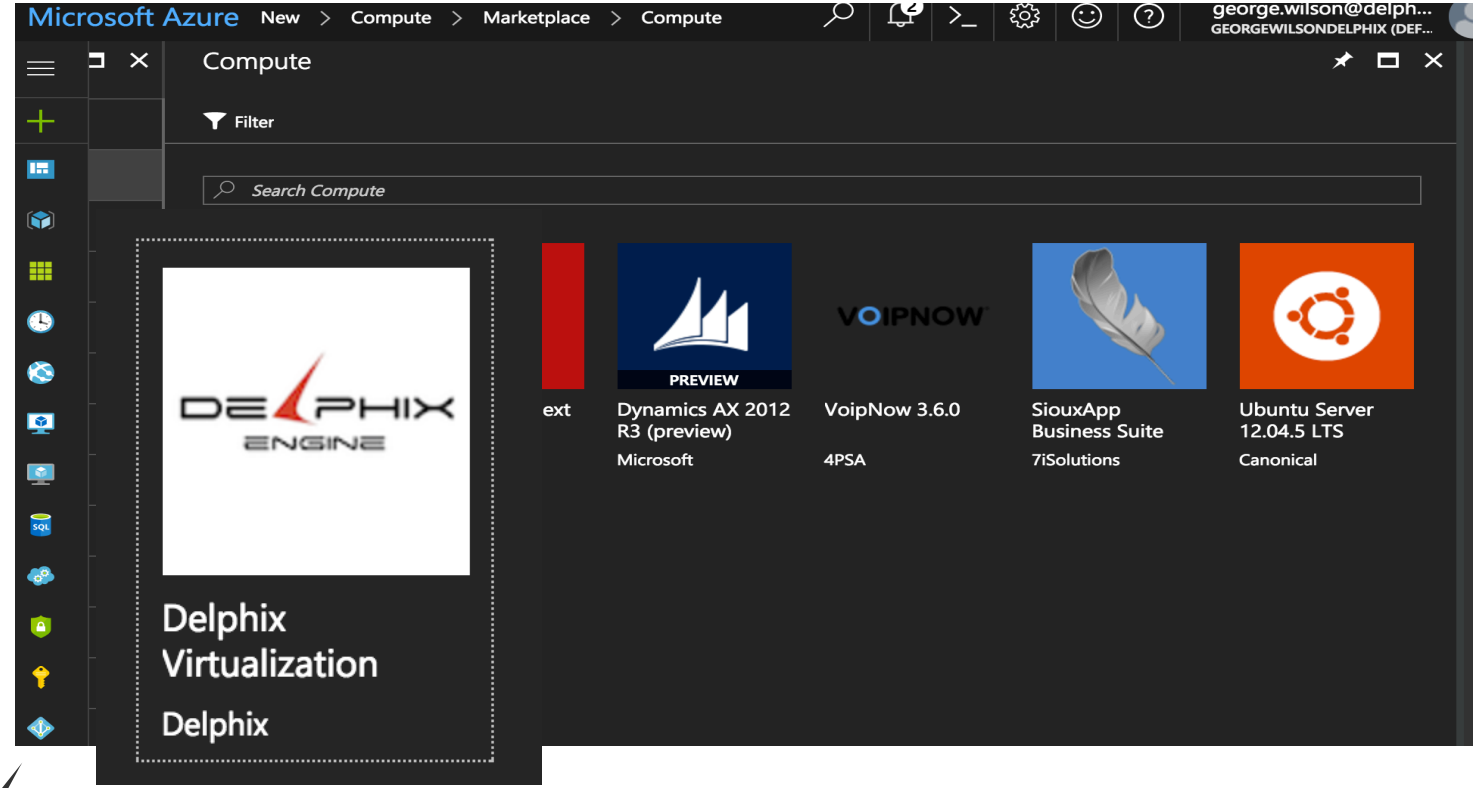
Putting it all together…

```
> ffffff03d6060d48::prtconf
DEVINFO          NAME
ffffff03d4cc1d50 i86pc (driver name: rootnex)
    ffffff03d6060d48 hv_vmbus, instance #0 (driver name: hv_vmbus)
        ffffff03d618a000 hv_storvsc, instance #0 (driver name: hv_storvsc)
            ffffff03d9cb3000 scsiclass,00, instance #0 (driver name: sd)
        ffffff03dbbc6d50 hv_storvsc, instance #1 (driver name: hv_storvsc)
            ffffff03dbbc6aa8 scsiclass,00, instance #1 (driver name: sd)
            ffffff03dbbc6800 scsiclass,00, instance #2 (driver name: sd)
        ffffff03dbbc6558 hv_heartbeat, instance #0 (driver name: hv_heartbeat)
        ffffff03dbbc62b0 hv_kvp, instance #0 (driver name: hv_kvp)
        ffffff03dbbc6008 hv_shutdown, instance #0 (driver name: hv_shutdown)
        ffffff03dc023d58 hv_timesync, instance #0 (driver name: hv_timesync)
        ffffff03dc023ab0 netvsc, instance #0 (driver name: netvsc)
```

Device details



Coming soon…



Panic in the cloud

```
Delphix Engine: 5.1.6.0
Hyper-V Version: 6.3.9600 [SP18
  Features=0xe7f<VPRUNTIME,TMREFCNT,SYNIC,SYNTM,APIC,HYPERCALL,VPINDEX,REFTSC,IDLE,TMFREQ>
  Features1=0x8b0<PostMessages,SignalEvents>
  Features2(PM)=0x0 [C2]
  Features3=0x17b2<DEBUG,XMMHC,IDLE,NUMA,TMFREQ,SYNCMC,CRASH,NPIEP>

panic[cpu0]/thread=ffffff021f677740: BAD TRAP: type=e (#pf Page fault) rp=ffffff0007acff70 addr=0 occurred in module "<unknown>" due to
a NULL pointer dereference
dladm: #pf Page fault
Bad kernel fault at addr=0x0
pid=124, pc=0x0, sp=0xffffff0007ad0068, eflags=0x10246
cr0: 8005003b<pg,wp,ne,et,ts,mp,pe> cr4: 1406f8<smep,osxsav,xmme,fxsr,pge,mce,pae,pse,de>
cr2: 0cr3: 1bfd18000cr8: c
        rdi:                1 rsi:         ffffffff rdx: ffffff021f677740
        rcx:                9  r8: ffffff021eb41100  r9: ffffff0007acfe50
        rax: ffffff021d9eb028 rbx:                1 rbp: ffffff0007ad04c0
        r10:                1 r11:                0 r12: ffffff021bb680c0
        r13: ffffff0007ad0090 r14: ffffff021bb68000 r15:             4080
        fsb:                0 gsb: fffffffffbc30d00  ds:               4b
         es:               4b  fs:                0  gs:              1c3
        trp:                e err:               10 rip:                0
         cs:               30 rfl:            10246 rsp: ffffff0007ad0068
         ss:               38
```

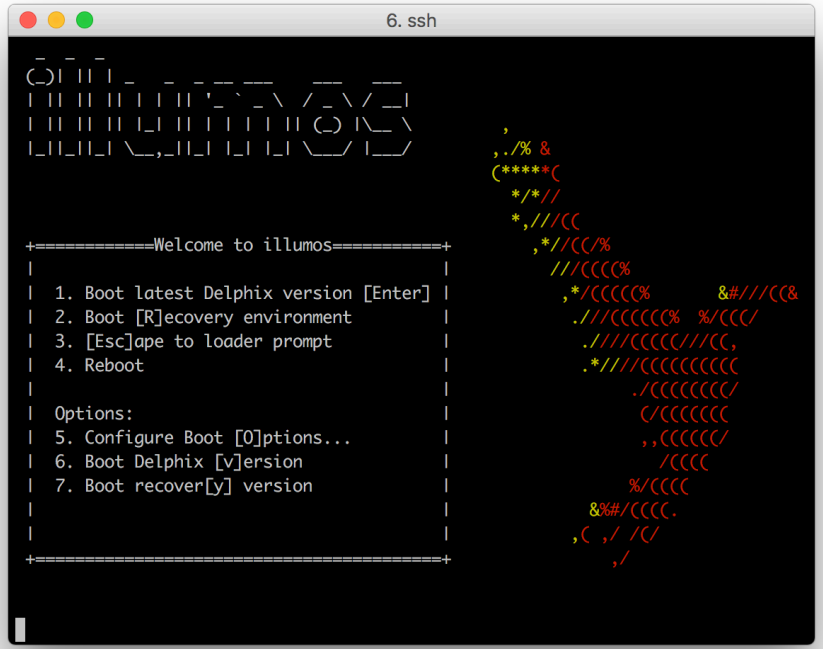
How to debug in the cloud?

- Limited console access
  - Read-only console
  - Unable to debug panic loops
- Boot failures are typically catastrophic
  - Misconfigured boot services
  - Broken upgrades
  - ZFS bugs

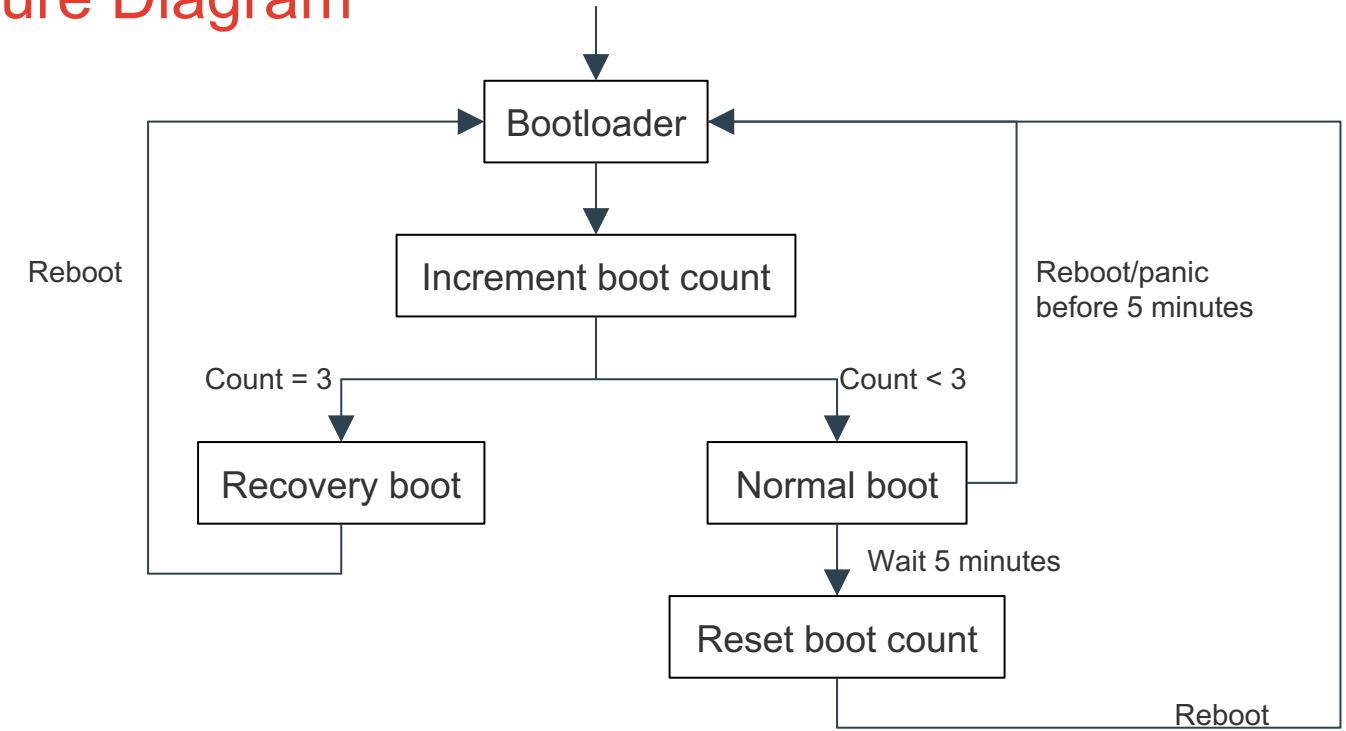
FreeBSD

FreeBSD Bootloader Extended

- FreeBSD bootloader is now the default for Illumos
- Introduce new recovery boot environment
  - ZFS bootable dataset
  - Ramdisk based
  - Automatically engaged after failed boots
  - Provides debugging tools and SSH access



Architecture Diagram




Observations…

Many Thanks!

Special Thanks To:

Kylie Liang
Yamin Qiao (sephe@)
Dexuan Cui
Hongjiang Zhang

George Wilson
@zfsdude
gwilson@delphix.com