KVM/ARM: The Design and Implementation of the Linux ARM Hypervisor

Christoffer Dall and Jason Nieh

COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

ARM®

~ 1.2 billion

~ 300 million

Source: Gartner Market Report, Jan 2014

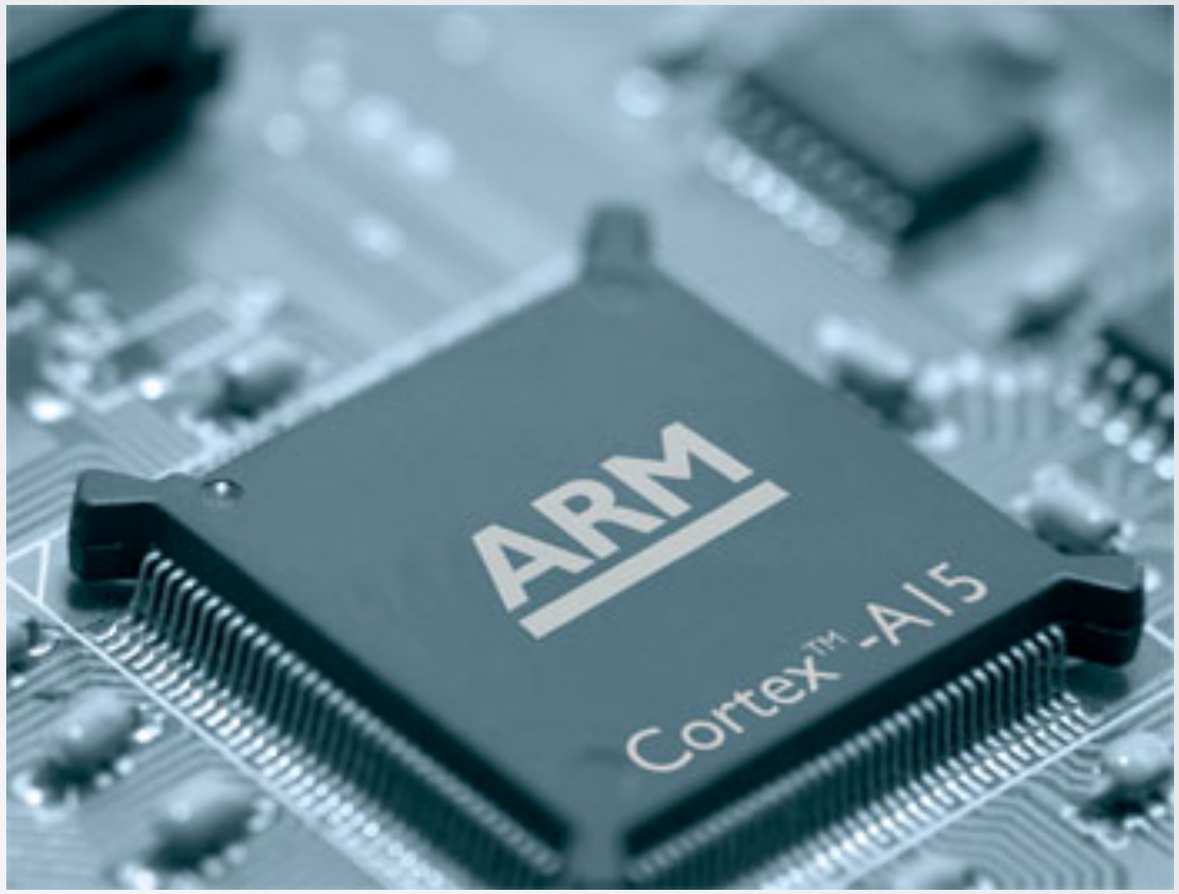
ARM®

ARM Servers

ARM Network Equipment

ARM®

Virtualization Extensions



Key Challenges

ARM® Virtualization Extensions != intel® VT-x

- No PC-standard on ARM

Key Contributions

- Split-Mode Virtualization
- Implemented and evaluated KVM/ARM
- Upstreamed implementation to mainline linux

Outline

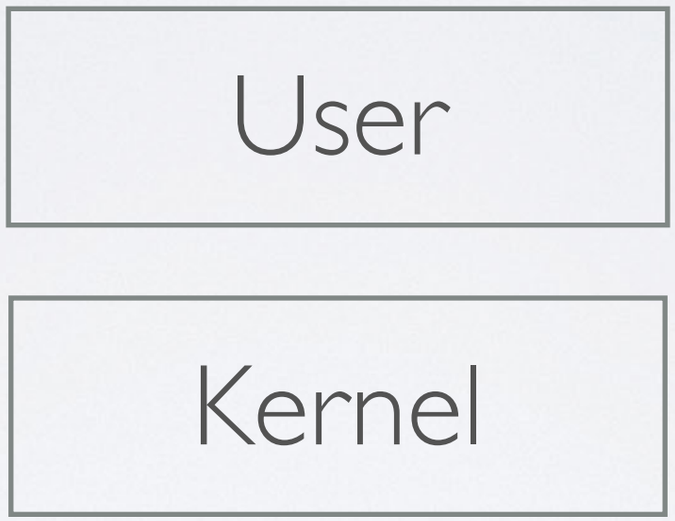
- **ARM Virtualization Extensions**
- Comparison to x86
- Split-Mode Virtualization
- Results
- Experiences

ARM Virtualization Extensions

- Provides virtualization in 4 key areas:
  - CPU Virtualization
  - Memory Virtualization
  - Interrupt Virtualization
  - Timer Virtualization

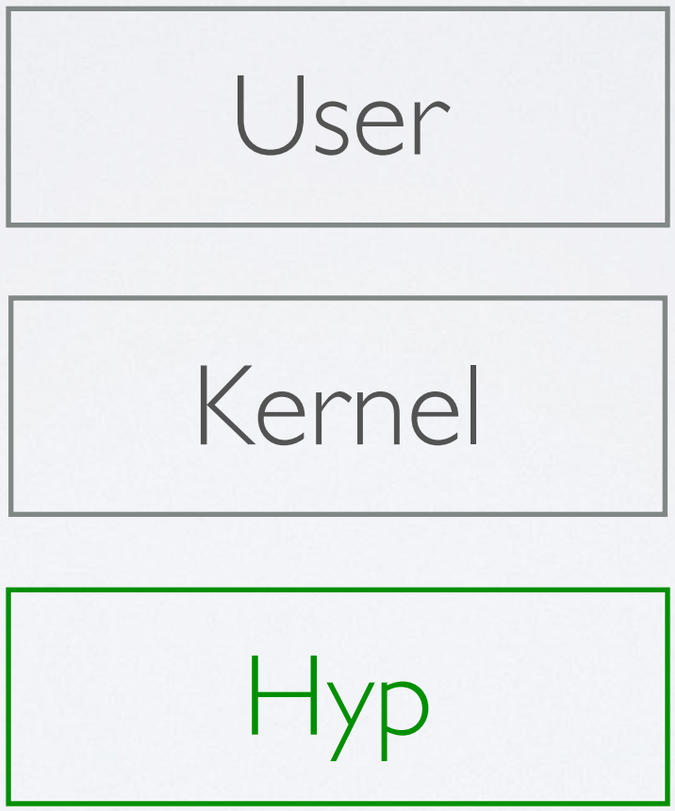
ARM Virtualization Extensions

CPU Virtualization



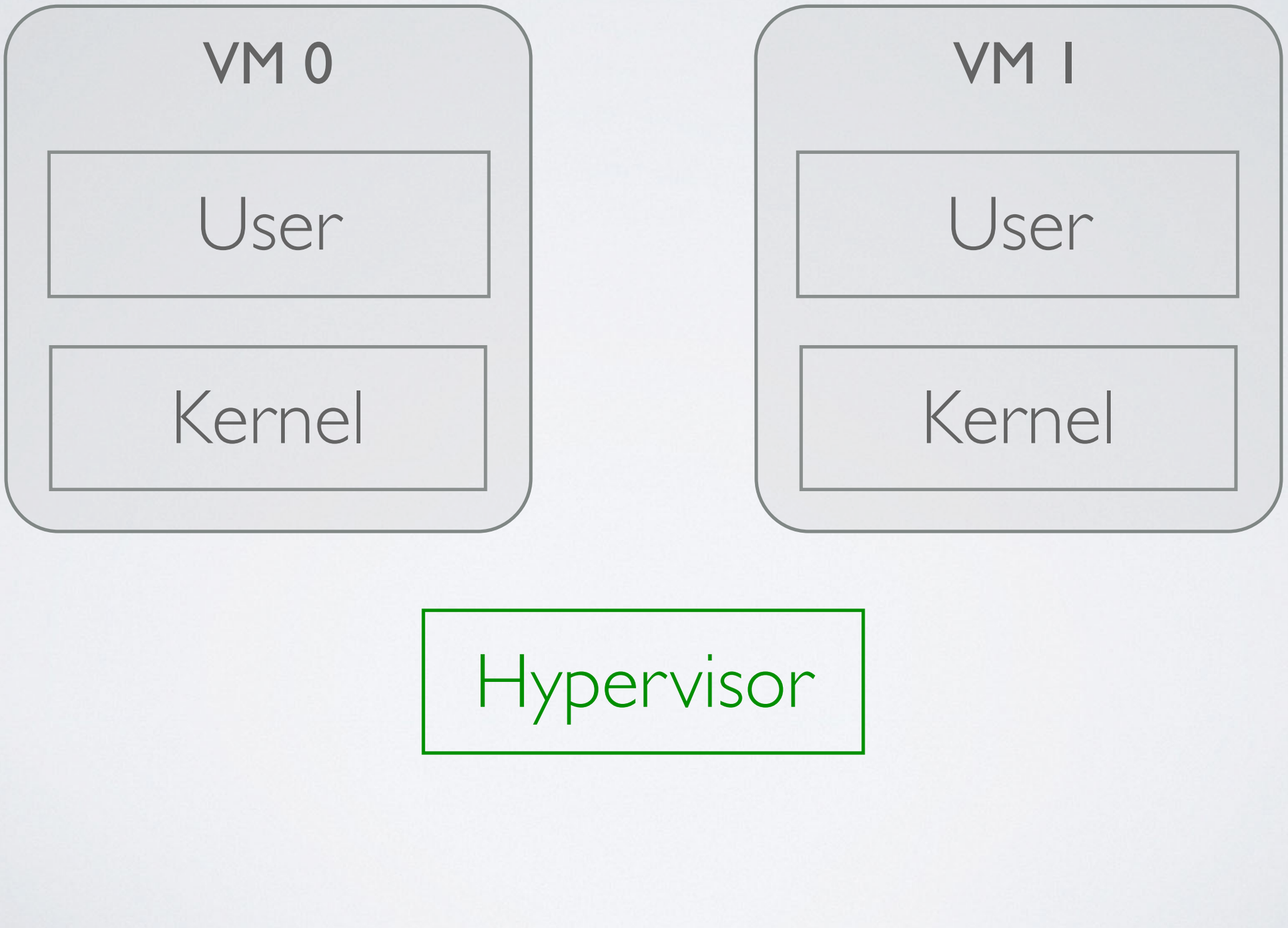
ARM Virtualization Extensions

CPU Virtualization



ARM Virtualization Extensions

CPU Virtualization



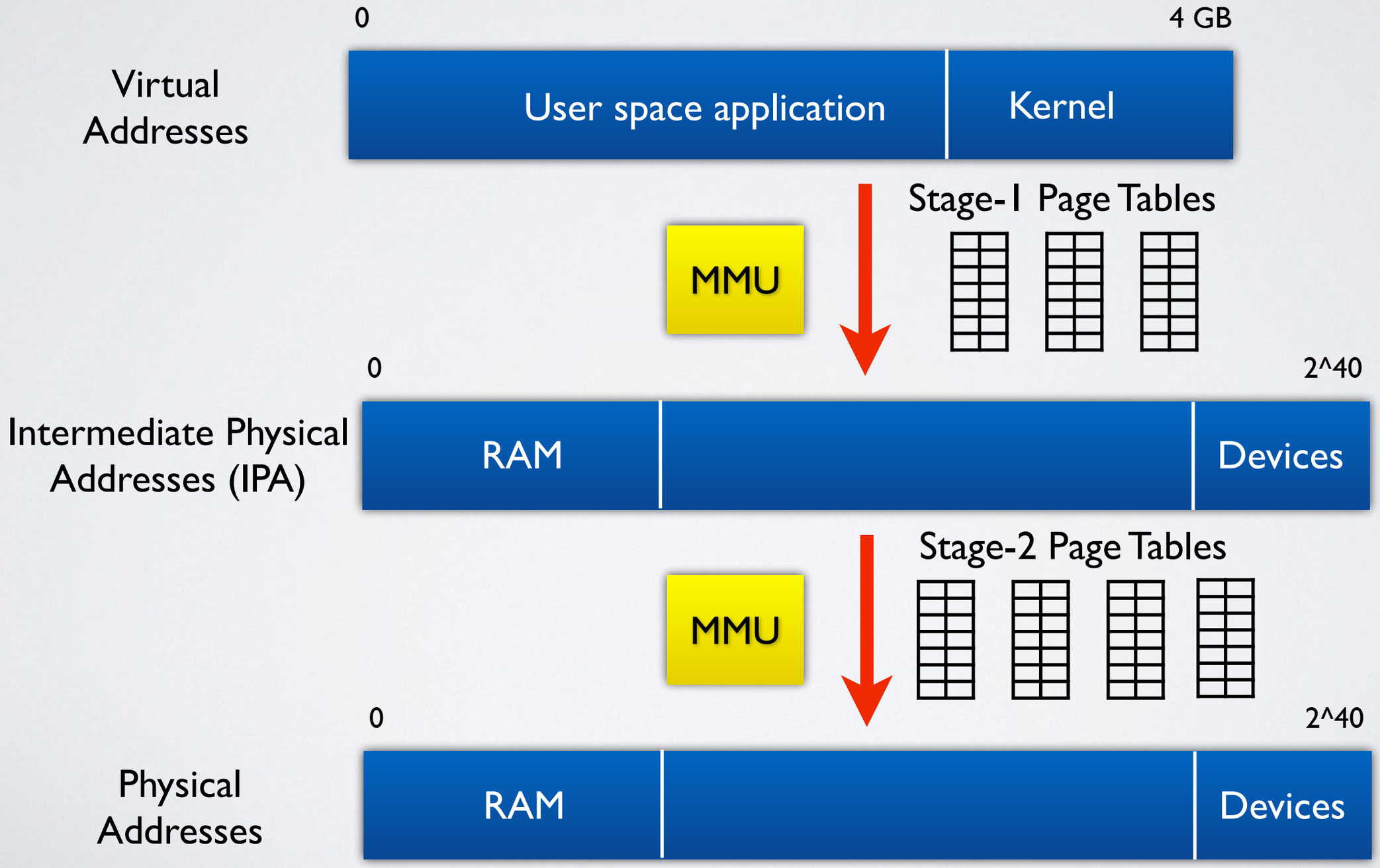
ARM Virtualization Extensions

CPU Virtualization

- Separate stack
- Different control registers
- Different page table format
- Controls Memory, Interrupt, and Timer virtualization
- Can be completely bypassed

ARM Virtualization Extensions

Memory Virtualization



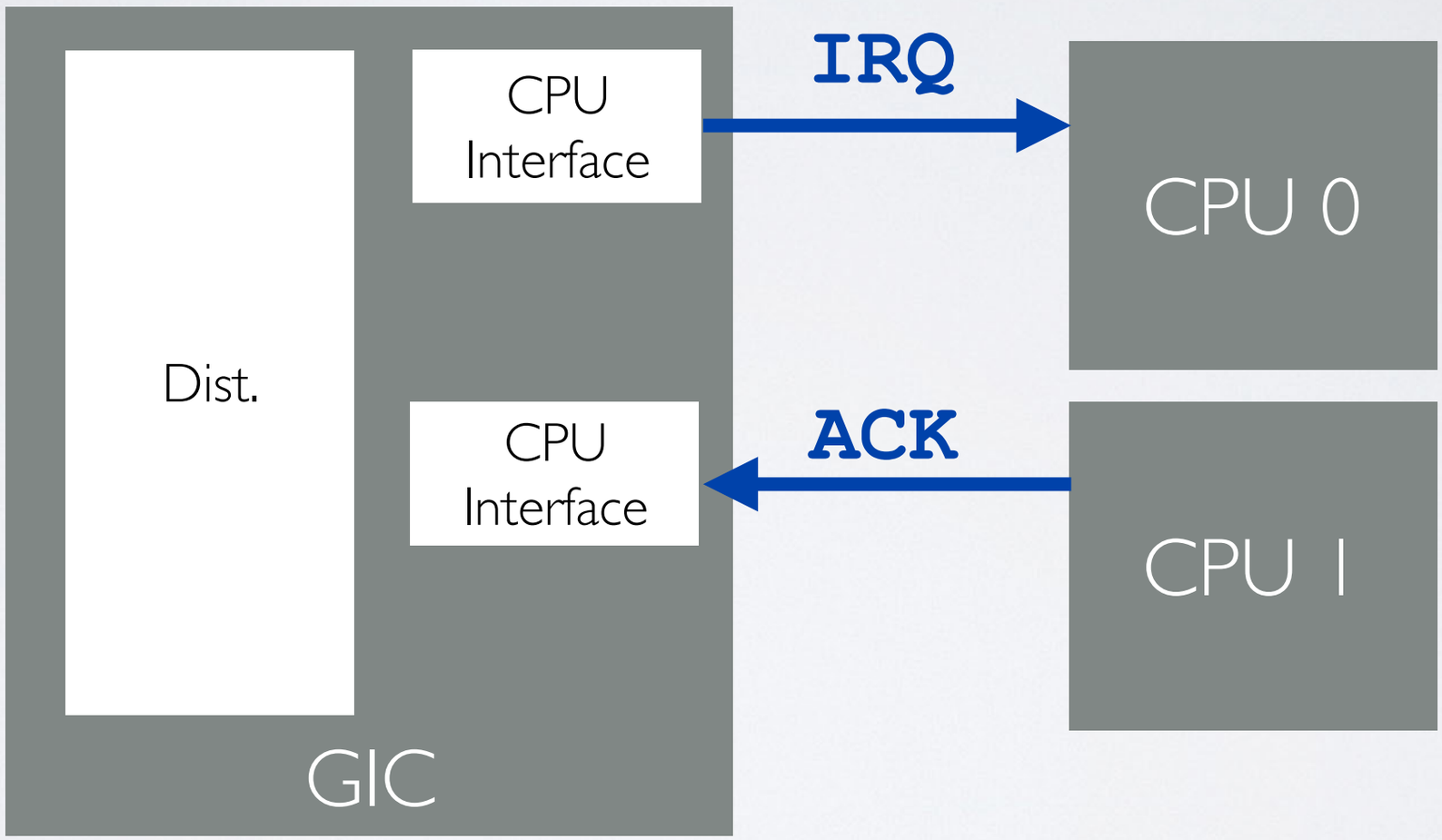
ARM Virtualization Extensions

Interrupt Virtualization

- Interrupts are an important part of any system
- The mechanism to notify CPU of events
- CPUs interact with an interrupt controller
- VMs accessing an interrupt controller traps
- Hurts performance

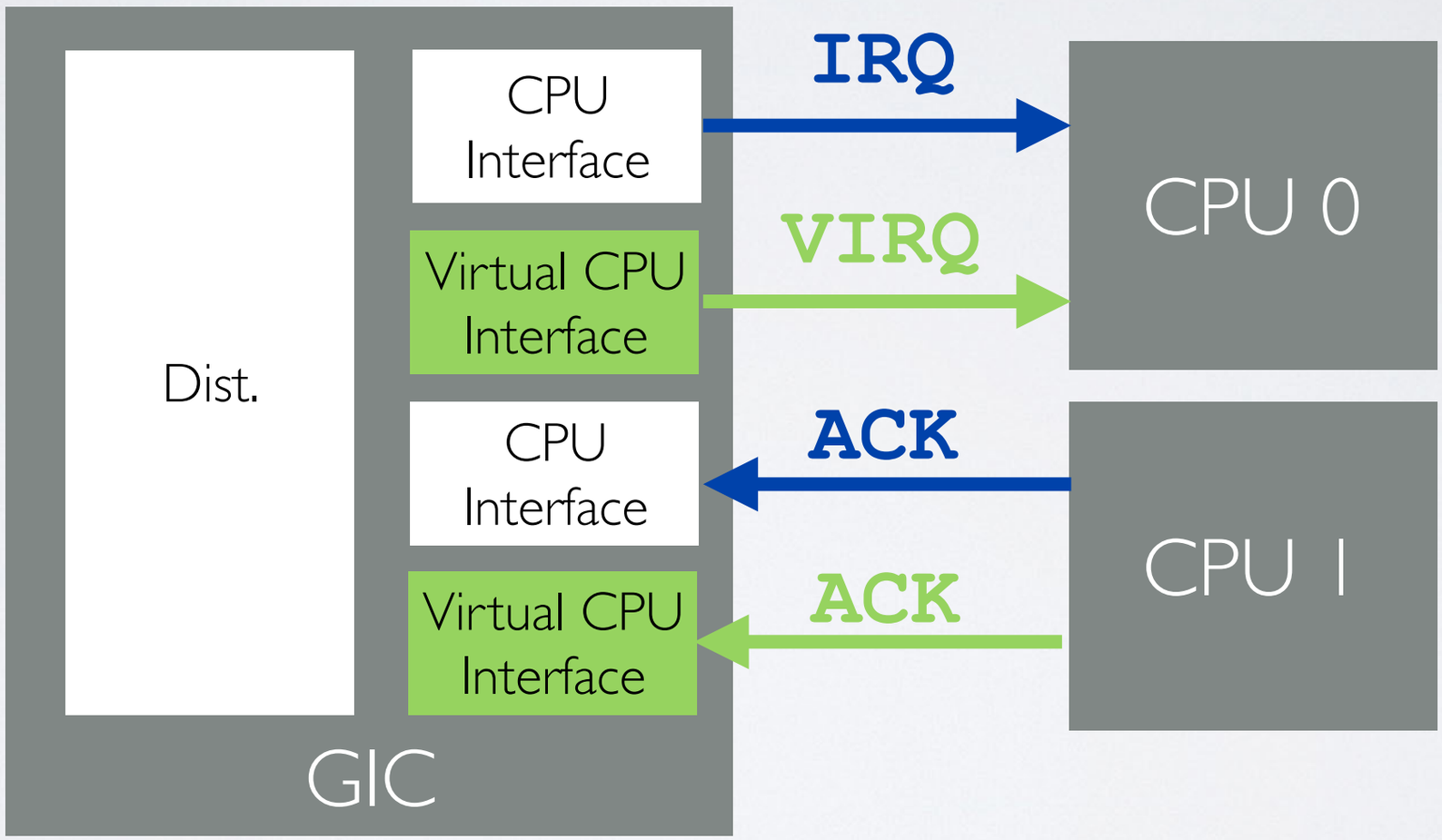
ARM Virtualization Extensions

Interrupt Virtualization



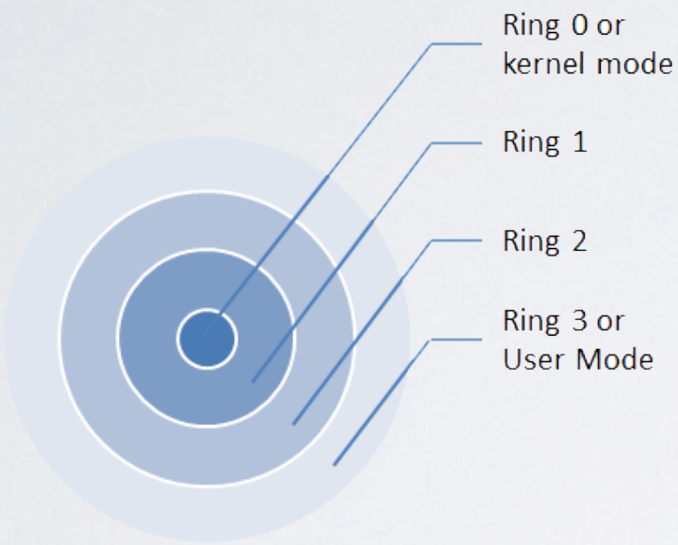
ARM Virtualization Extensions

Interrupt Virtualization



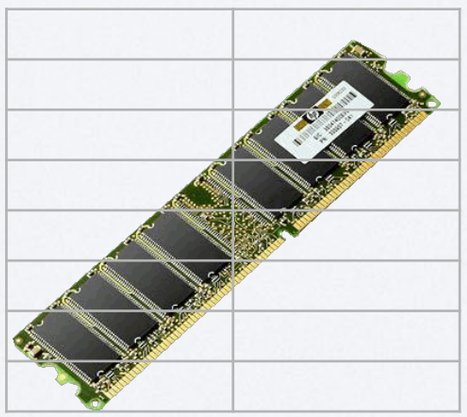
x86

Root



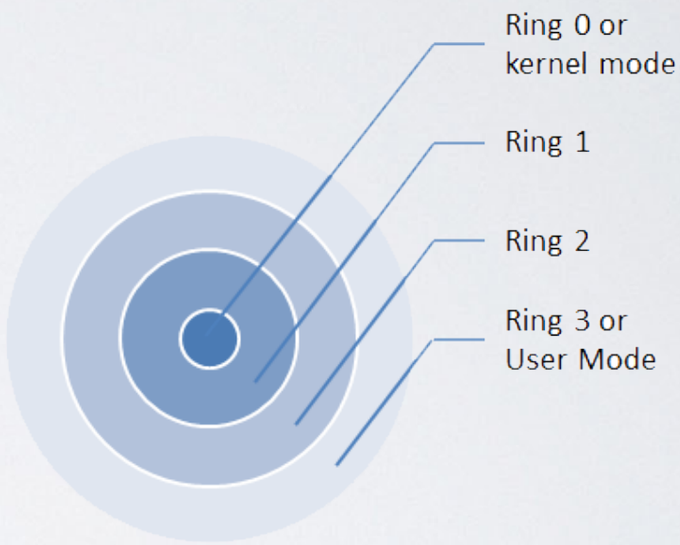
VM ENTRY

VM EXIT



VMCS

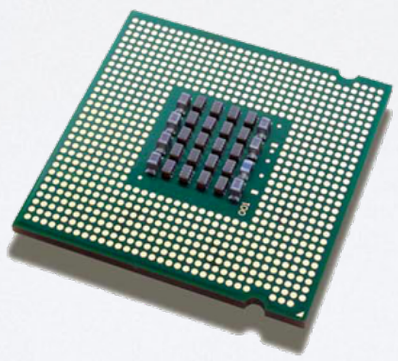
Non-Root (VM)



ARM

User Space

Kernel



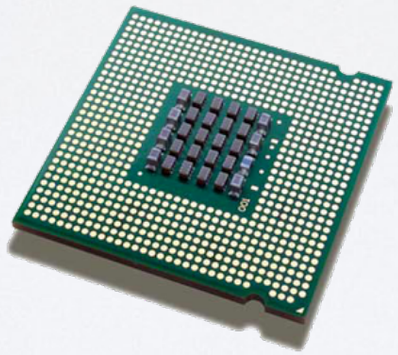
Trap

Hypervisor (HYP)

ARM

User Space

Kernel



Trap

Hypervisor (HYP)

?

Standalone hypervisor on ARM

- Reimplement basic OS functionality
- Hardware support is a pain!

TEXAS INSTRUMENTS

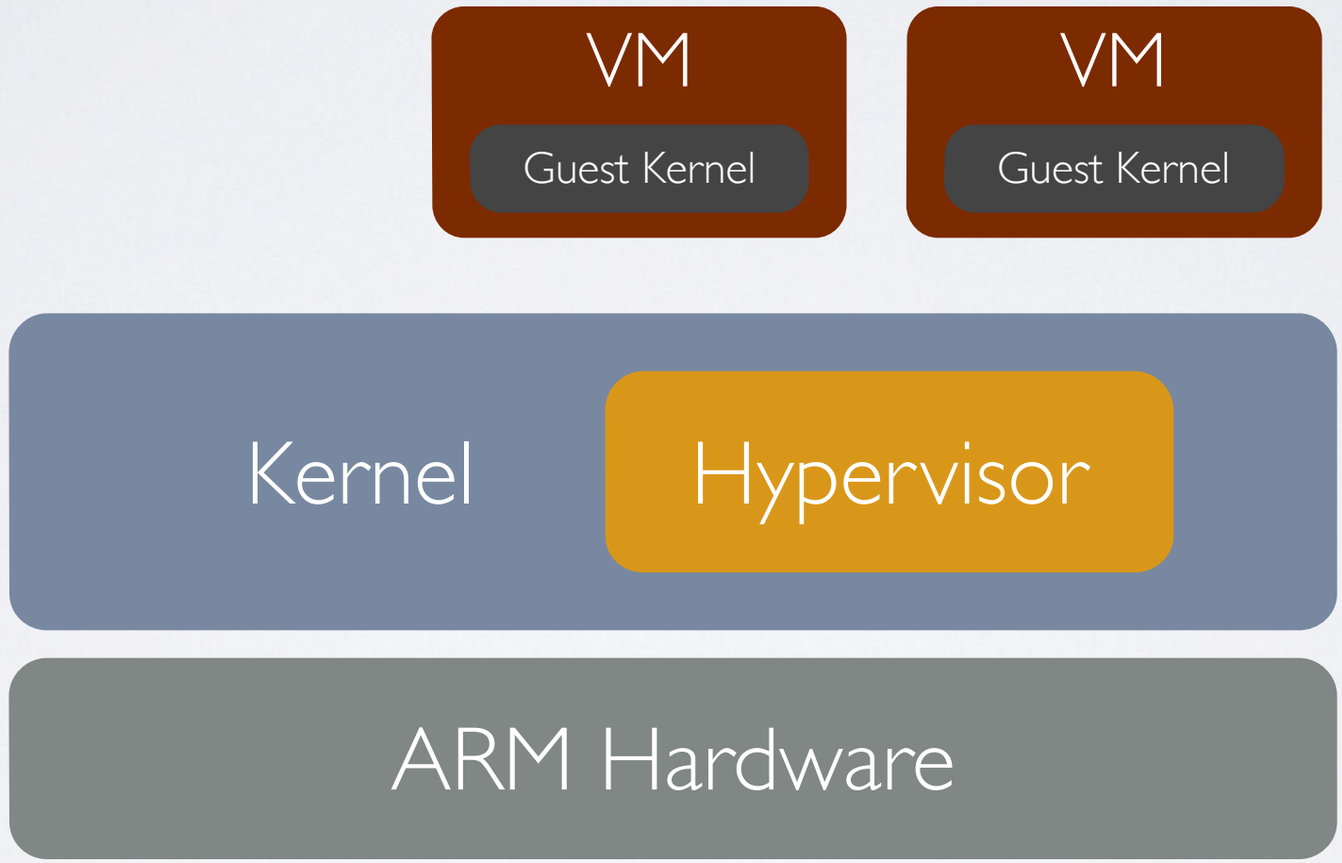
NVIDIA

SAMSUNG

QUALCOMM®

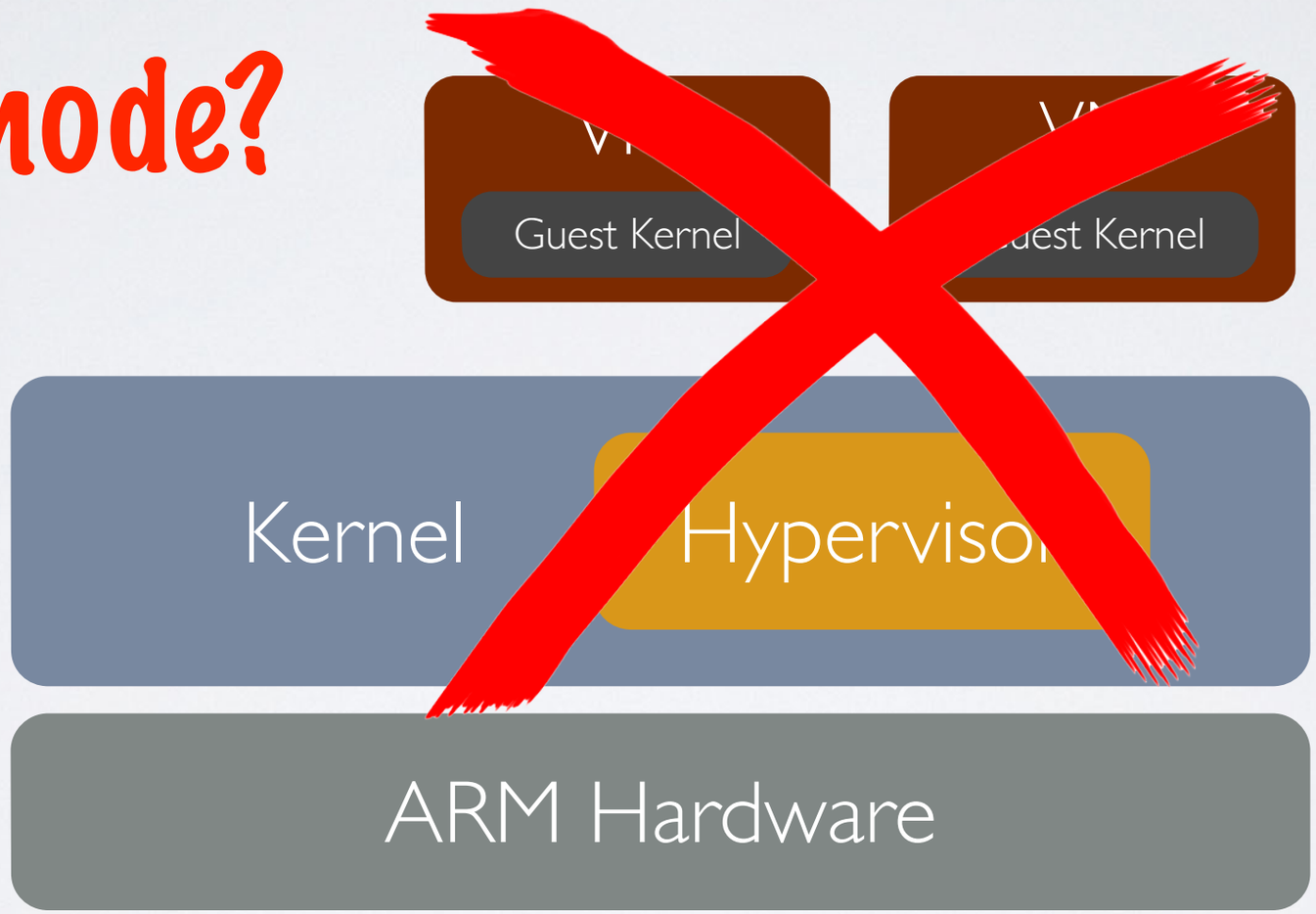
Hosted Hypervisor on ARM

Hosted Hypervisor on ARM

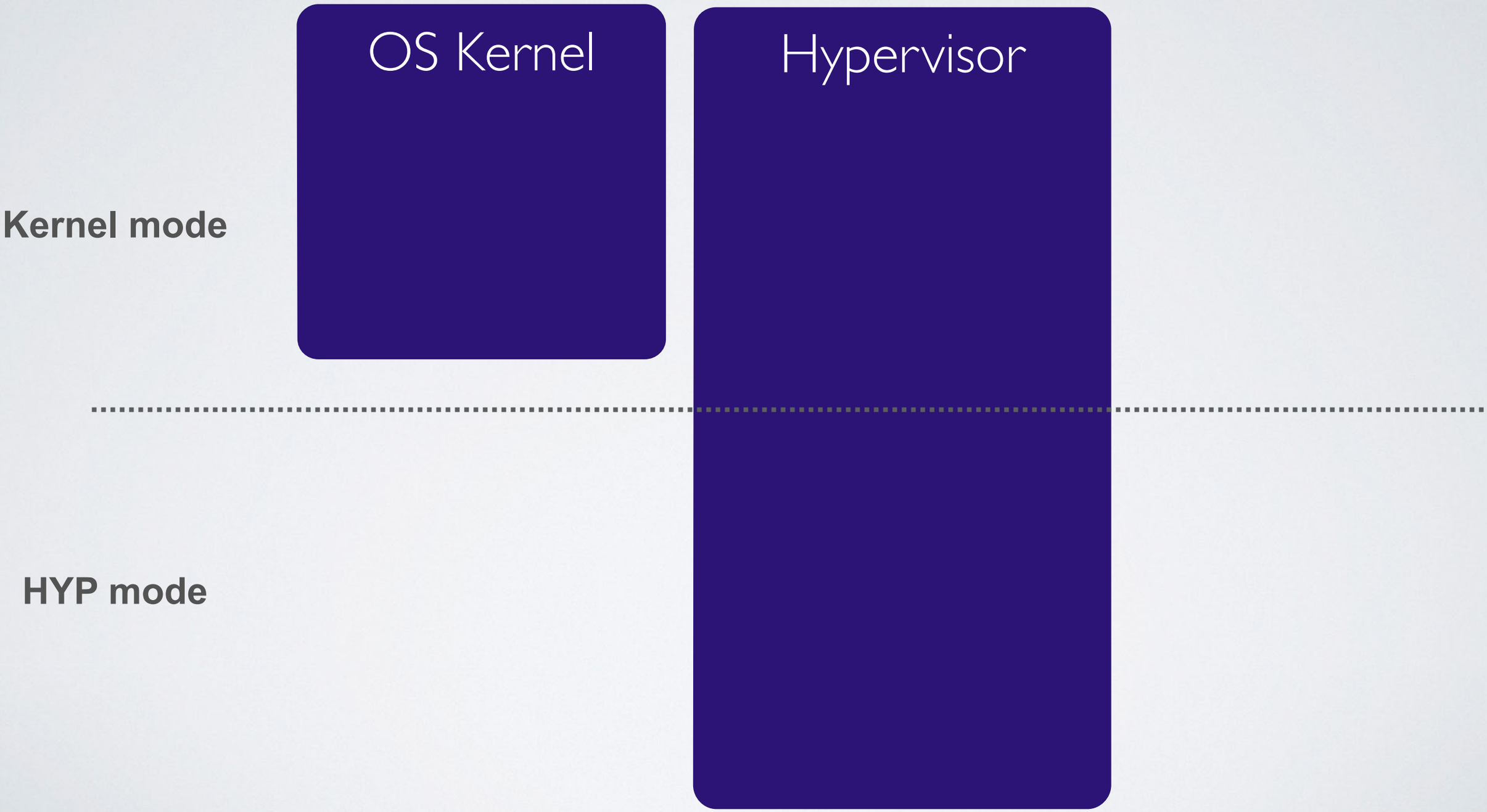


Hosted Hypervisor on ARM

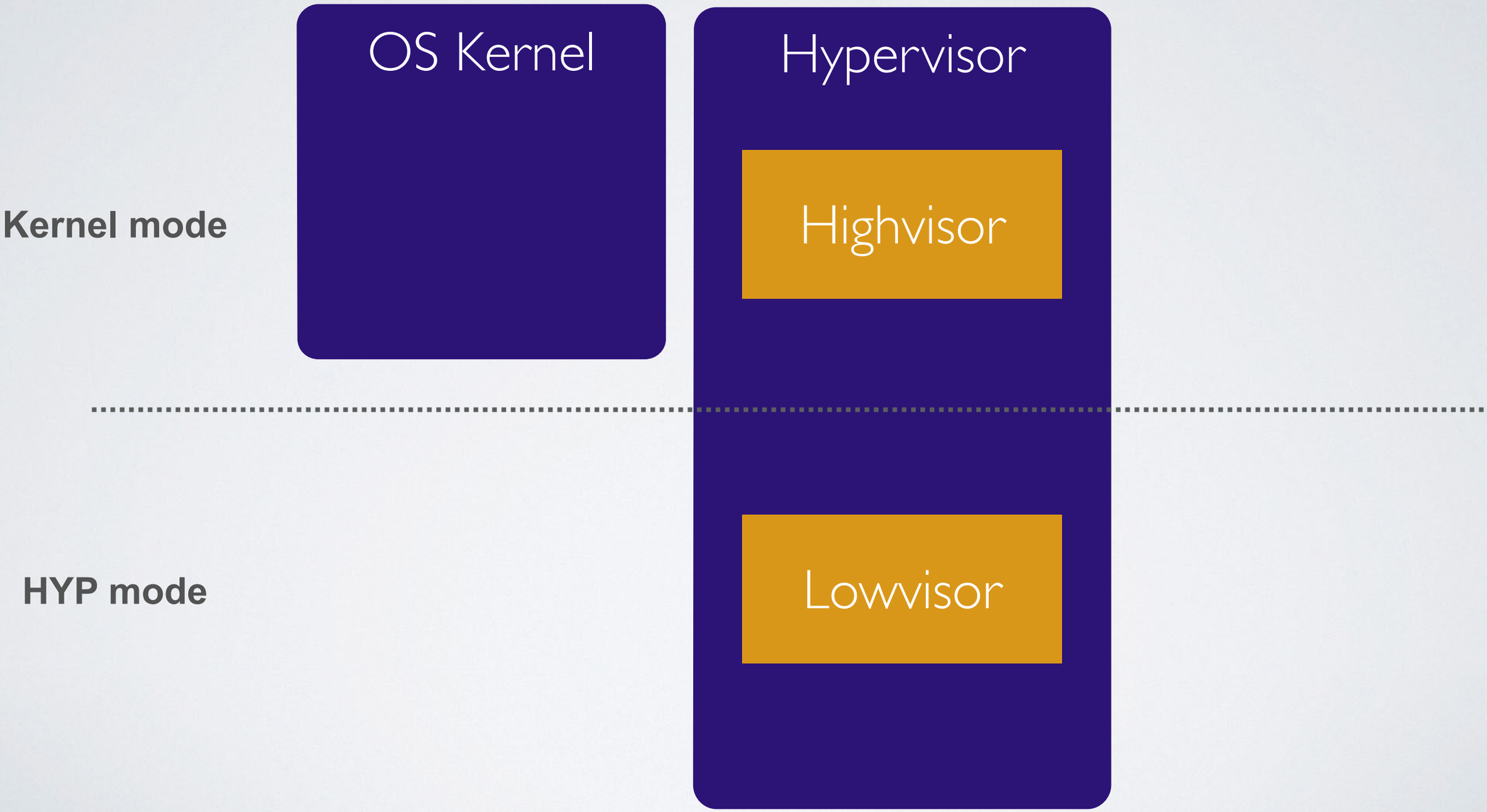
HYP mode?



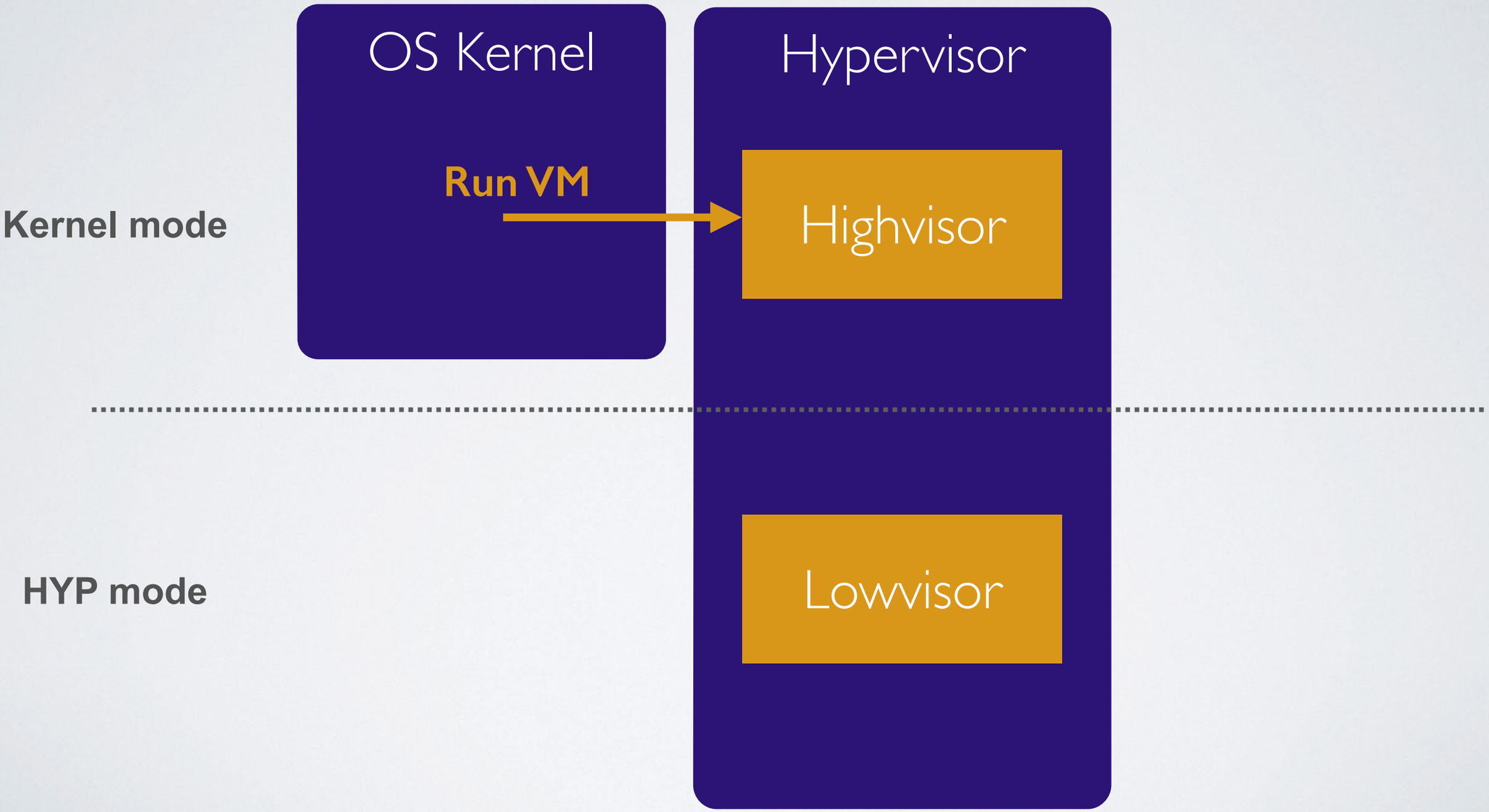
Split-Mode Virtualization



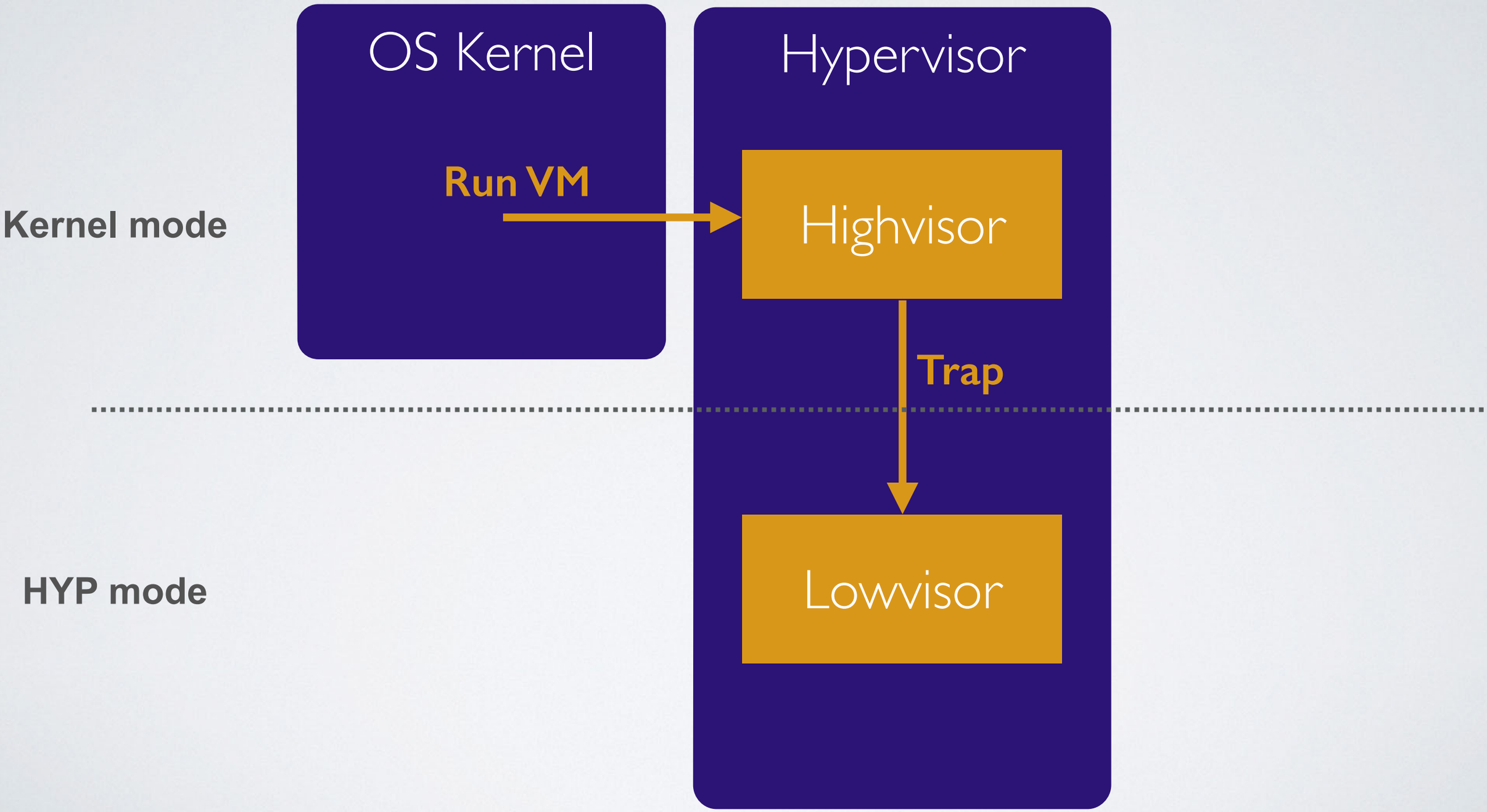
Split-Mode Virtualization



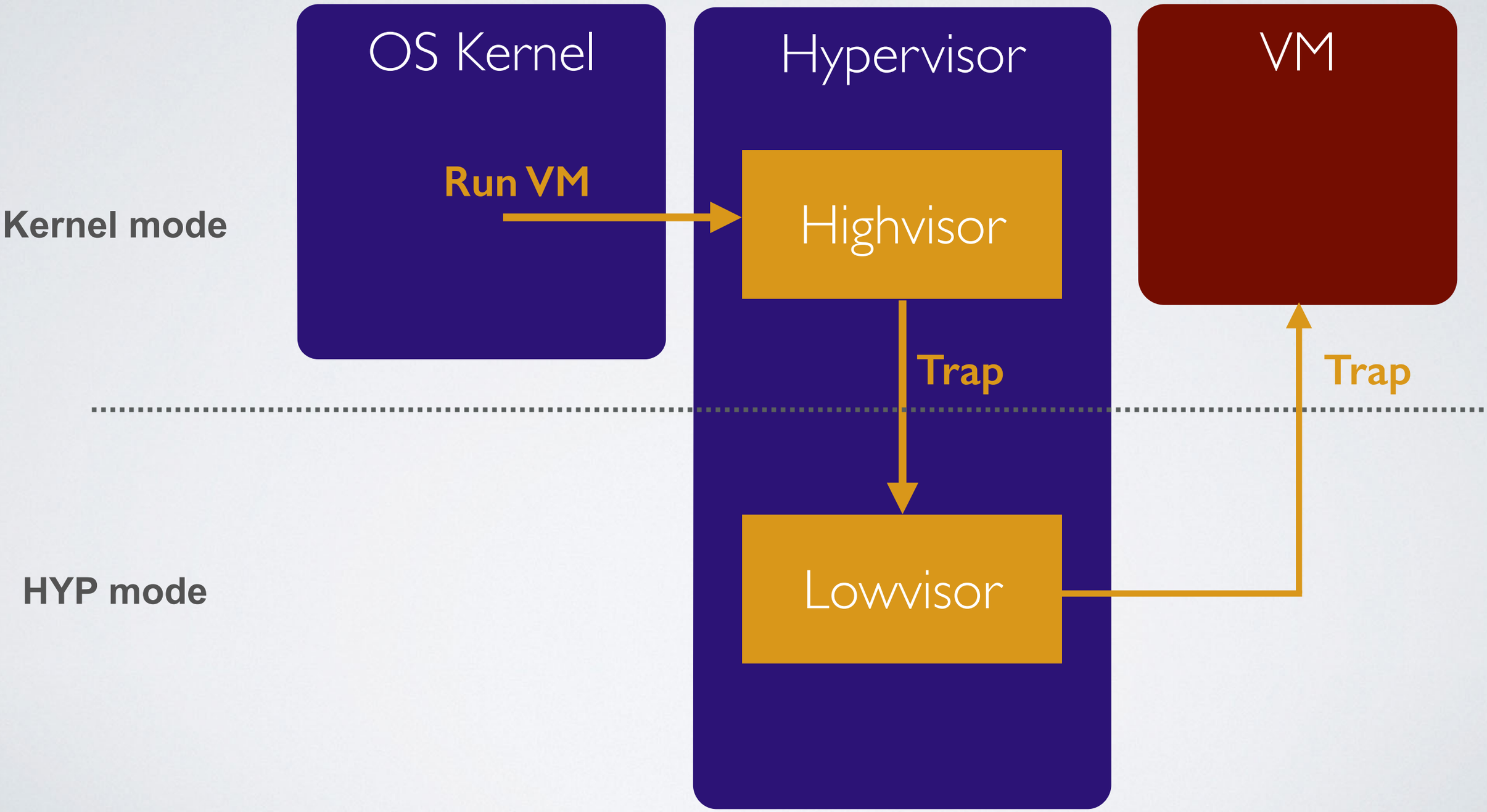
Split-Mode Virtualization



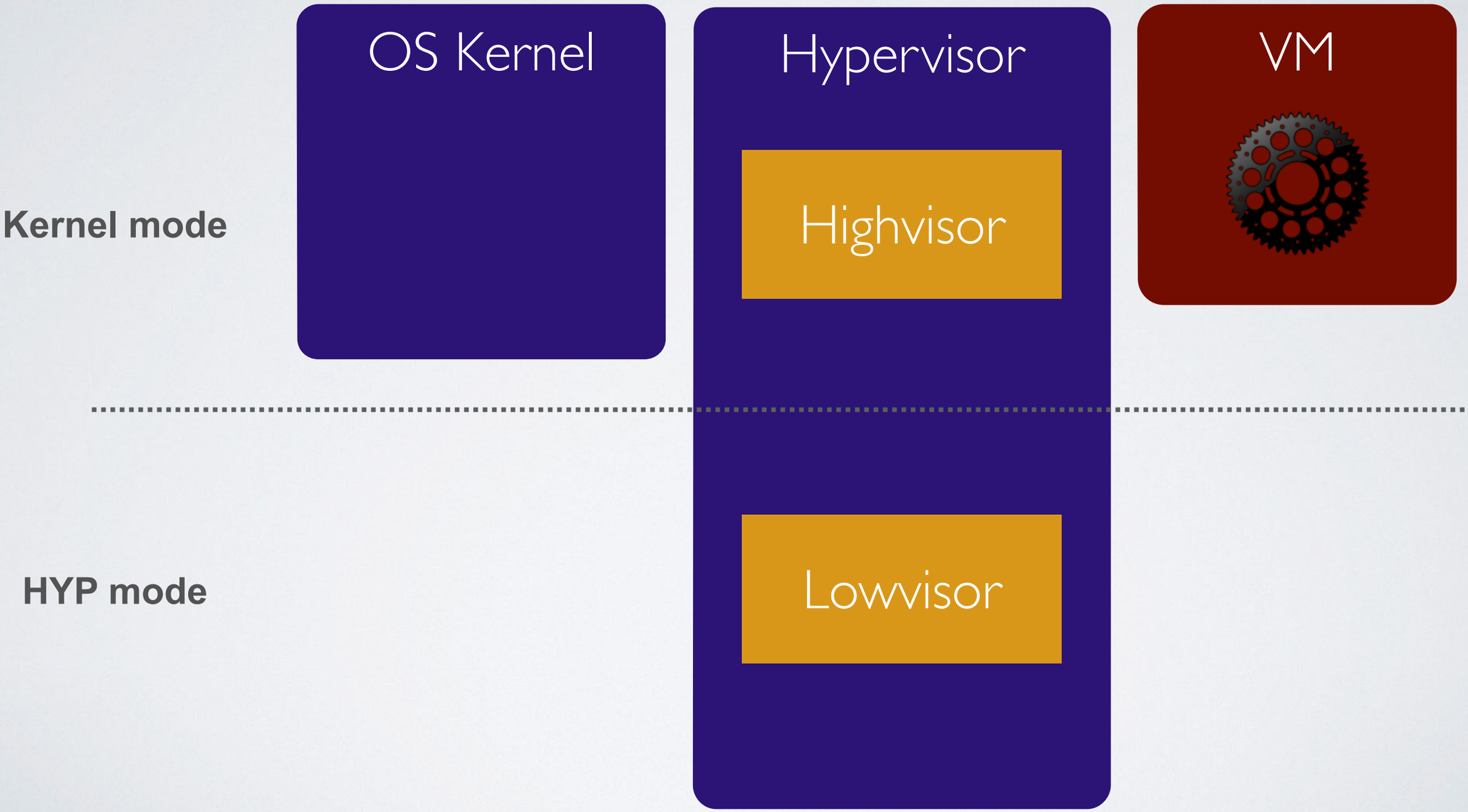
Split-Mode Virtualization



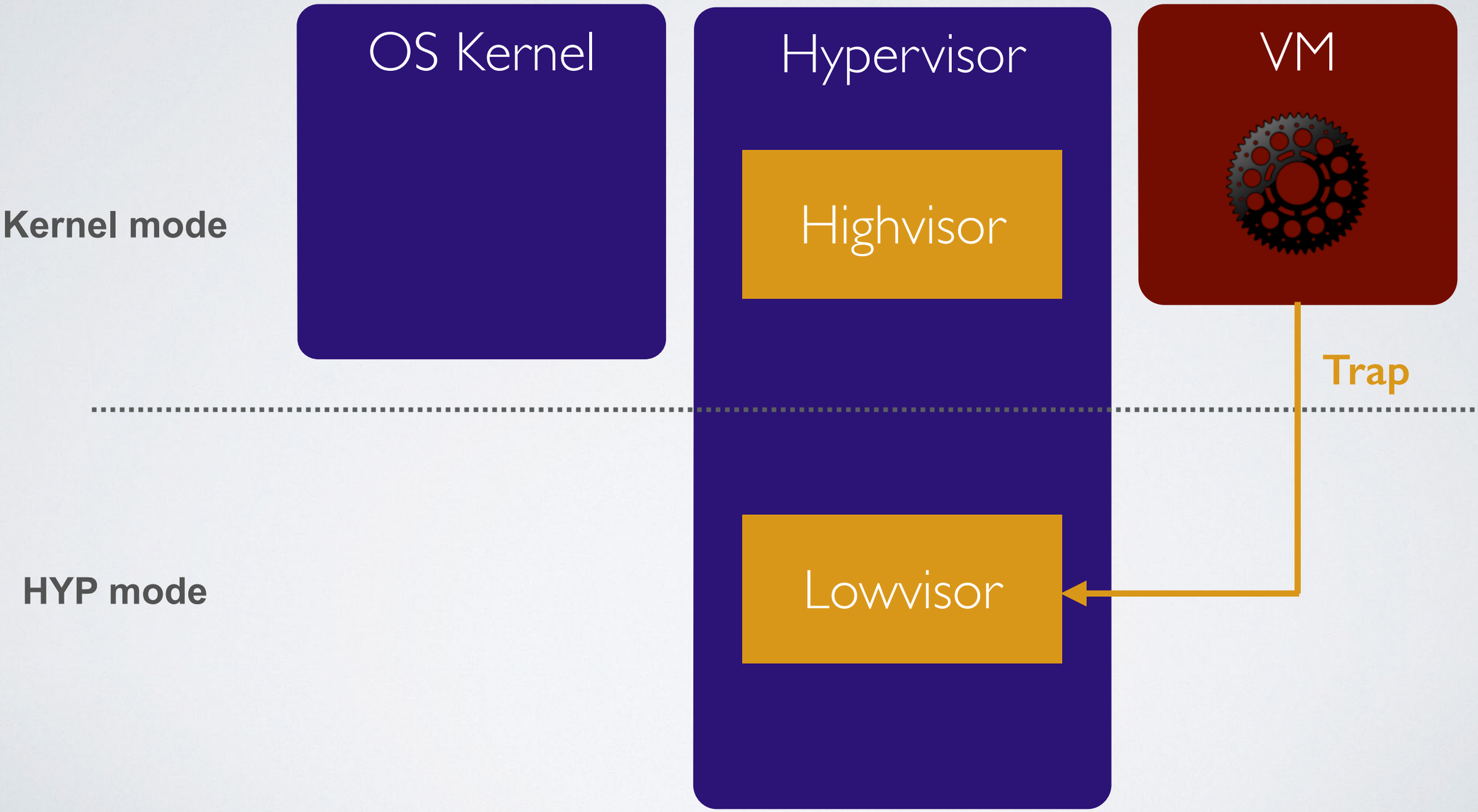
Split-Mode Virtualization



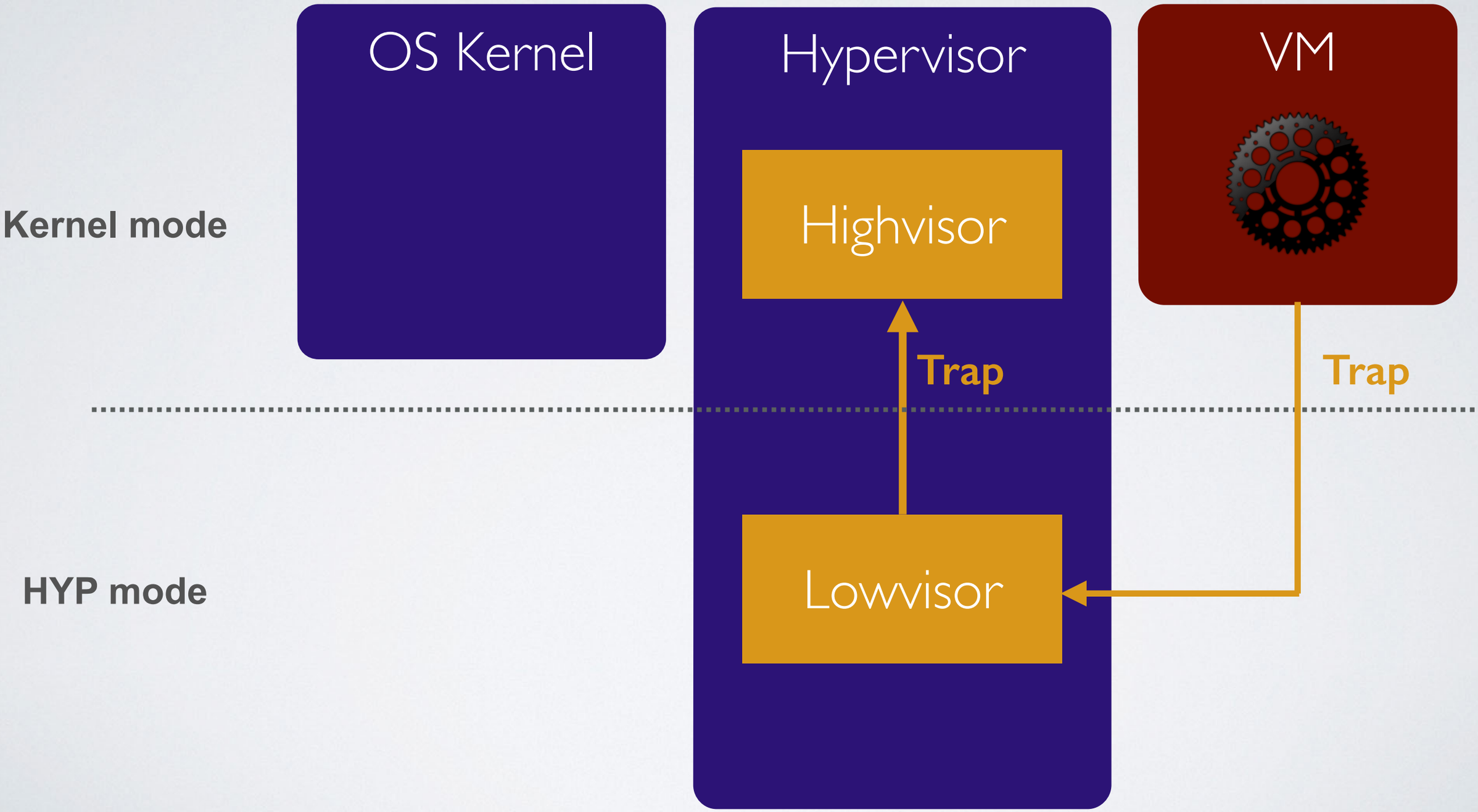
Split-Mode Virtualization



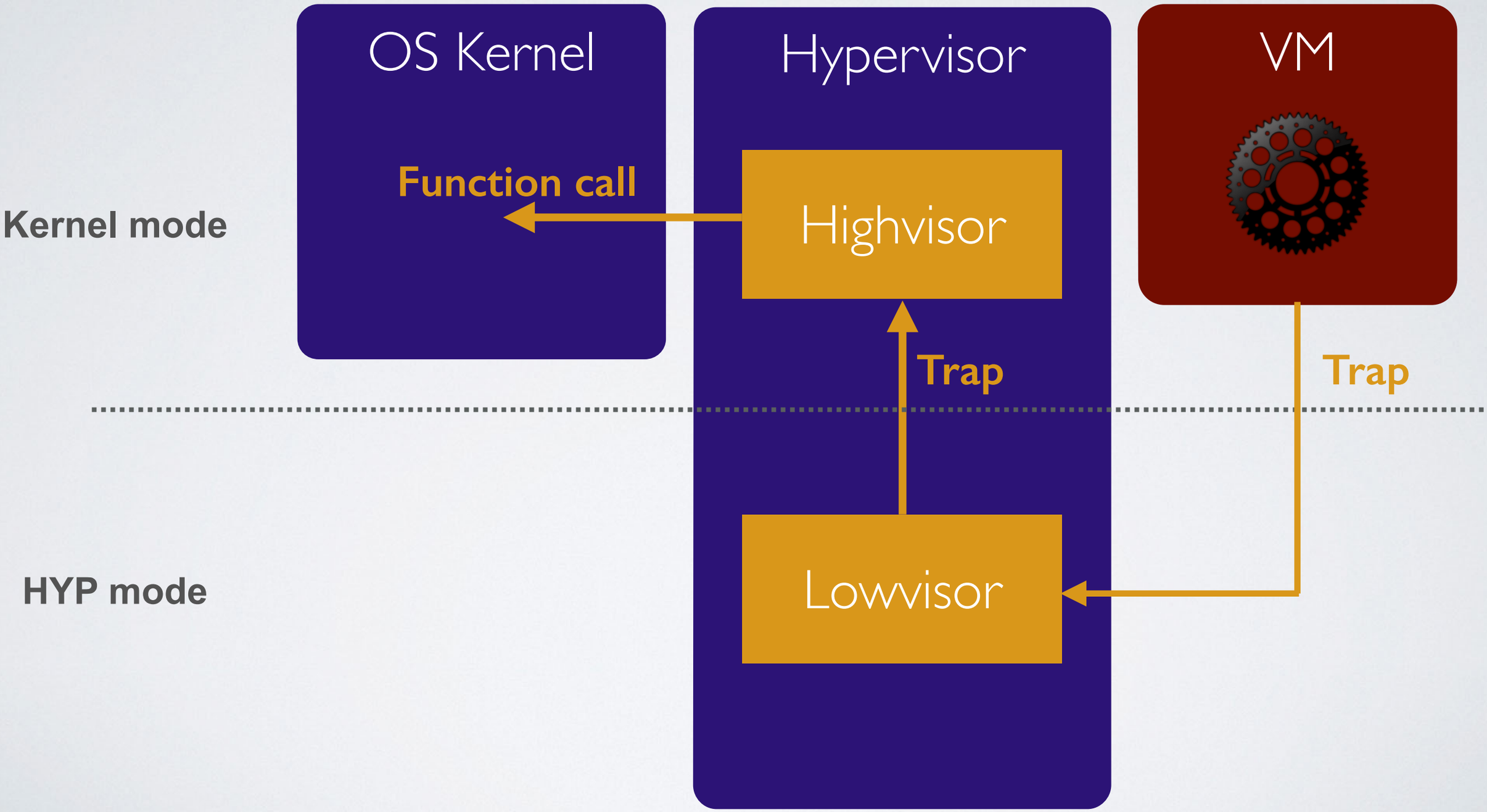
Split-Mode Virtualization



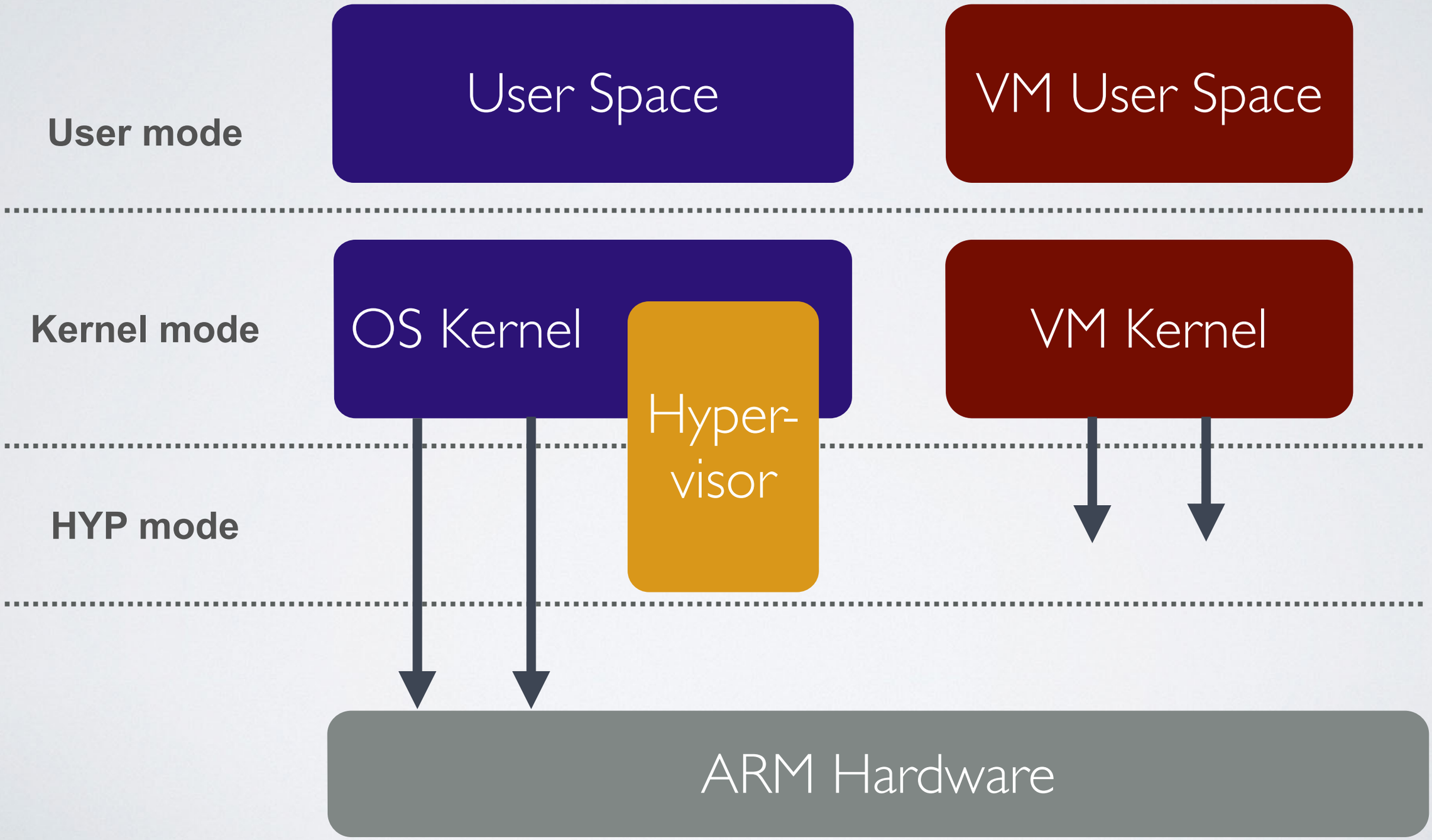
Split-Mode Virtualization



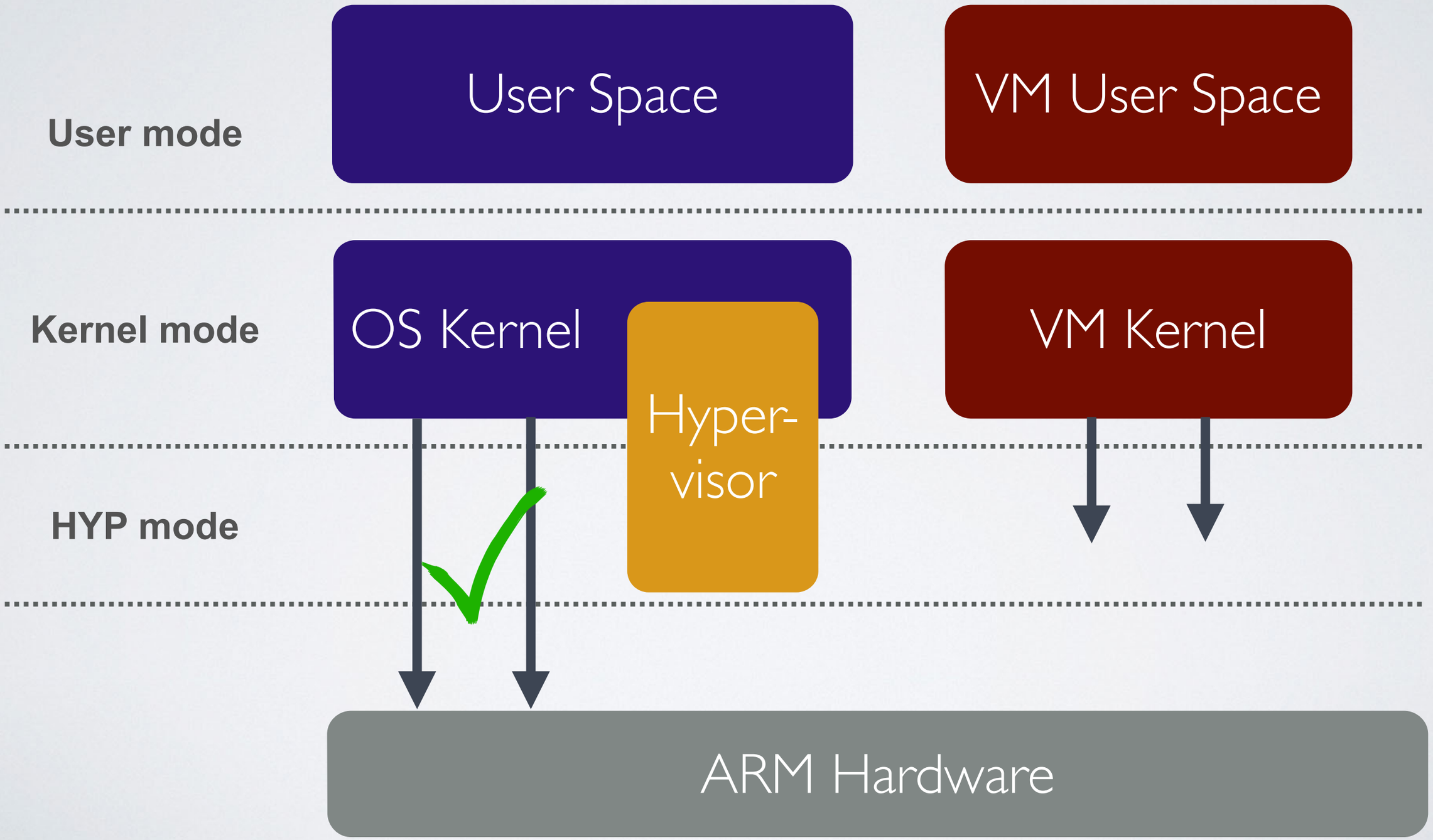
Split-Mode Virtualization



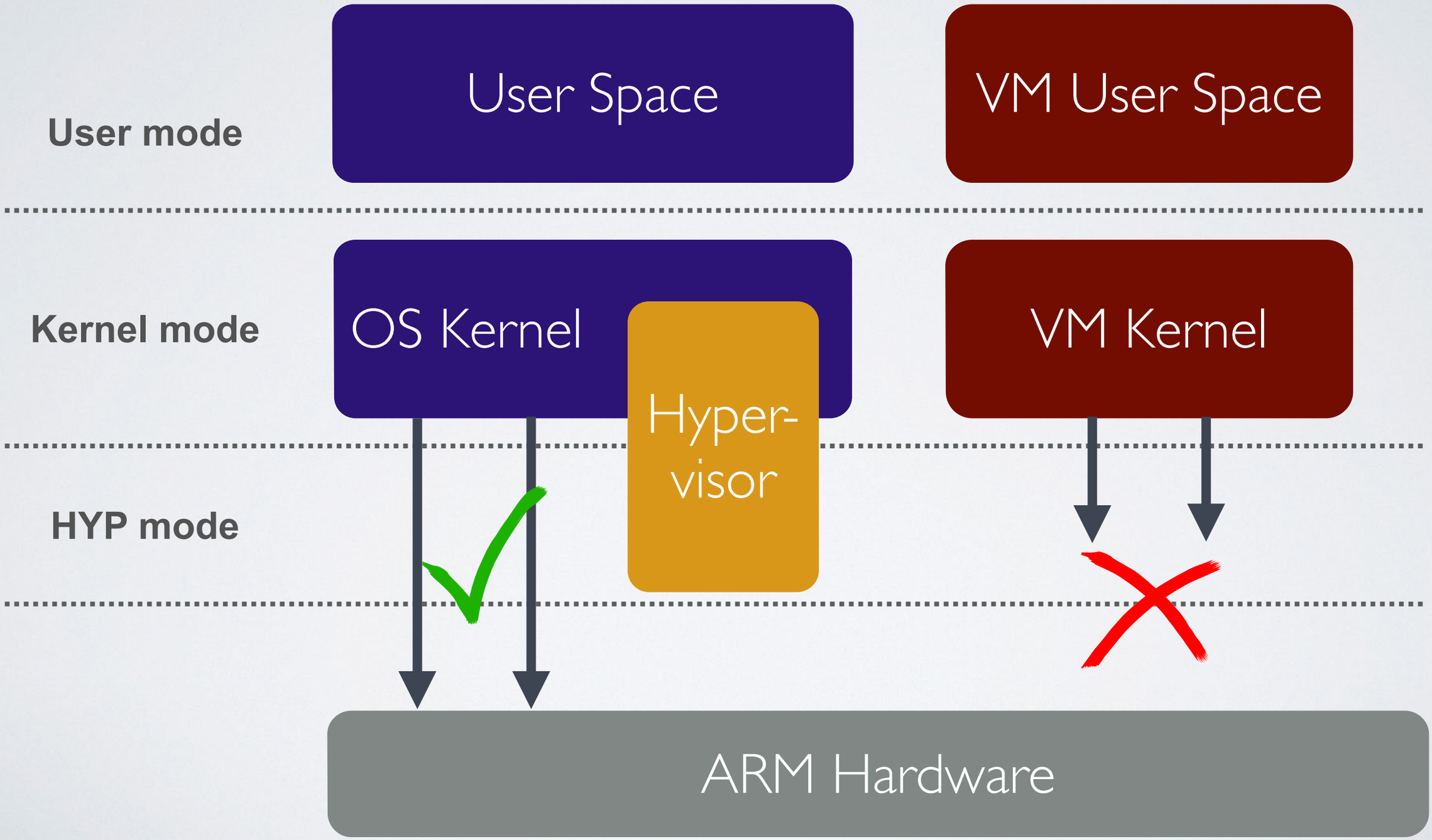
Split-Mode Virtualization



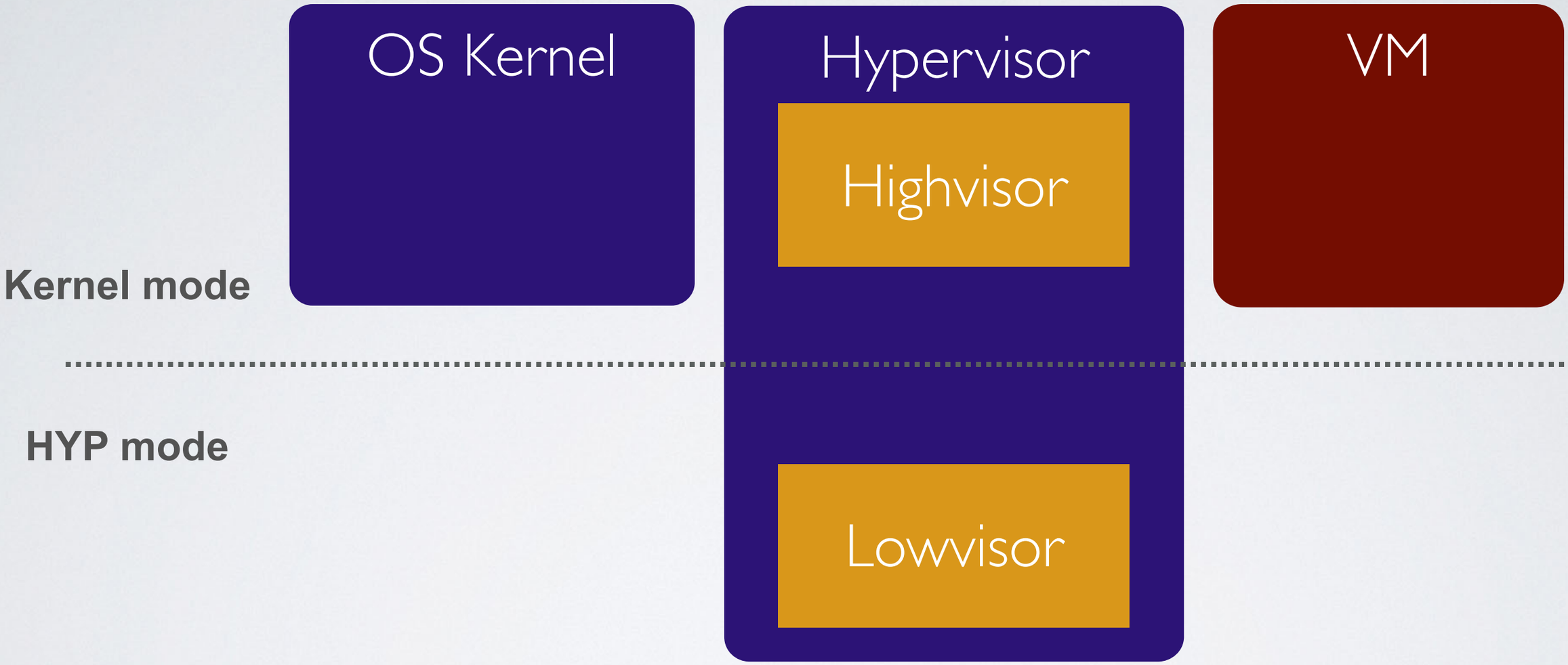
Split-Mode Virtualization



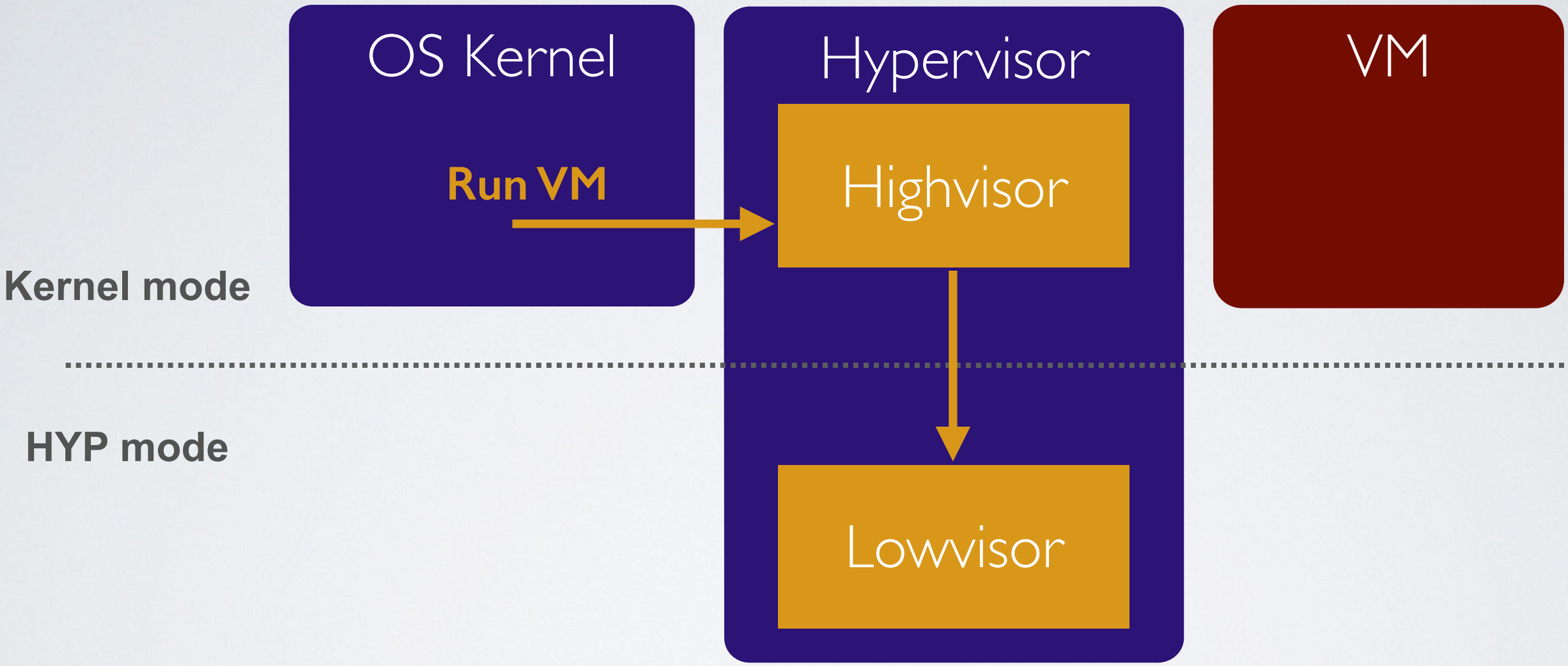
Split-Mode Virtualization



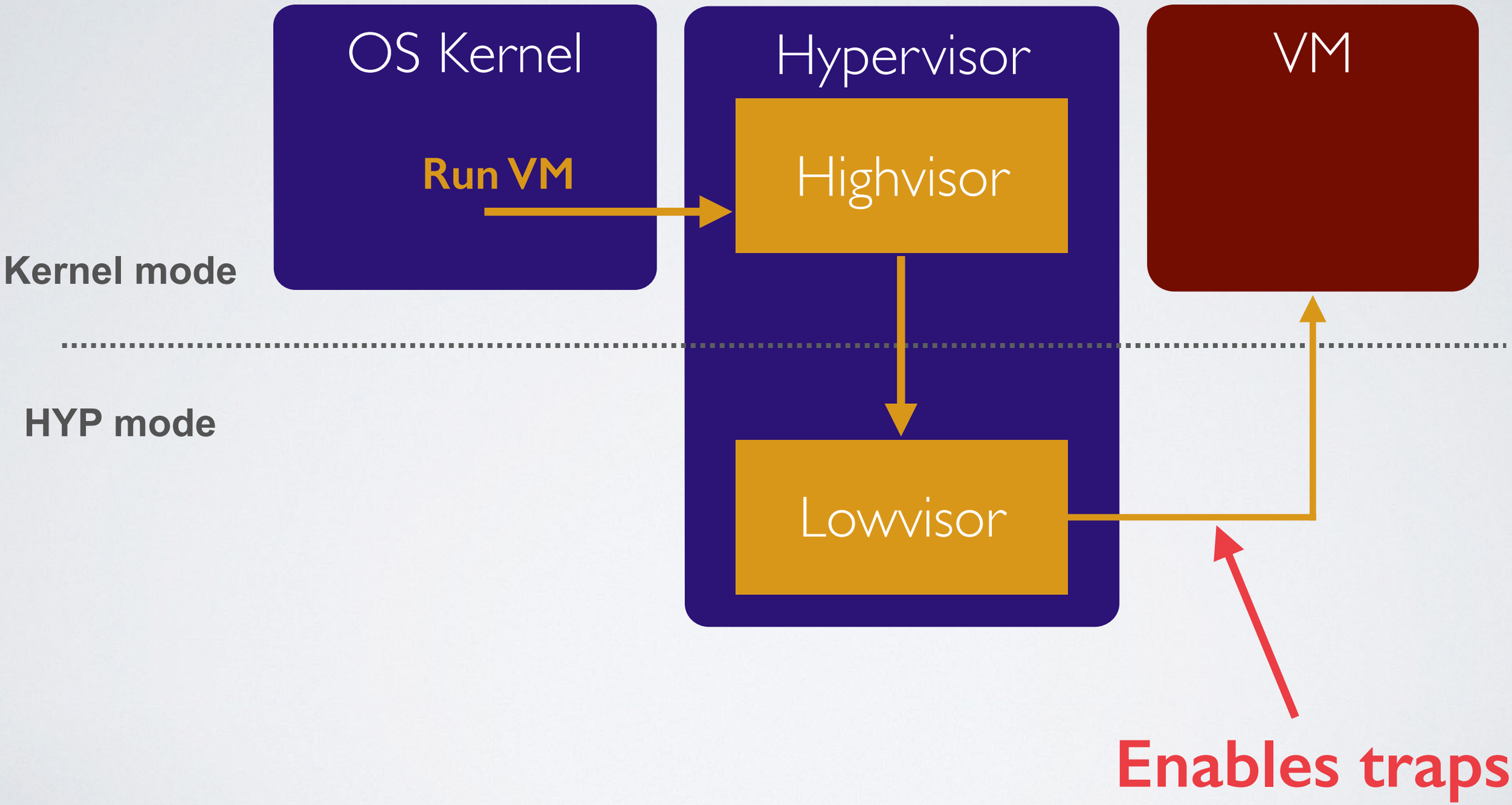
CPU Virtualization



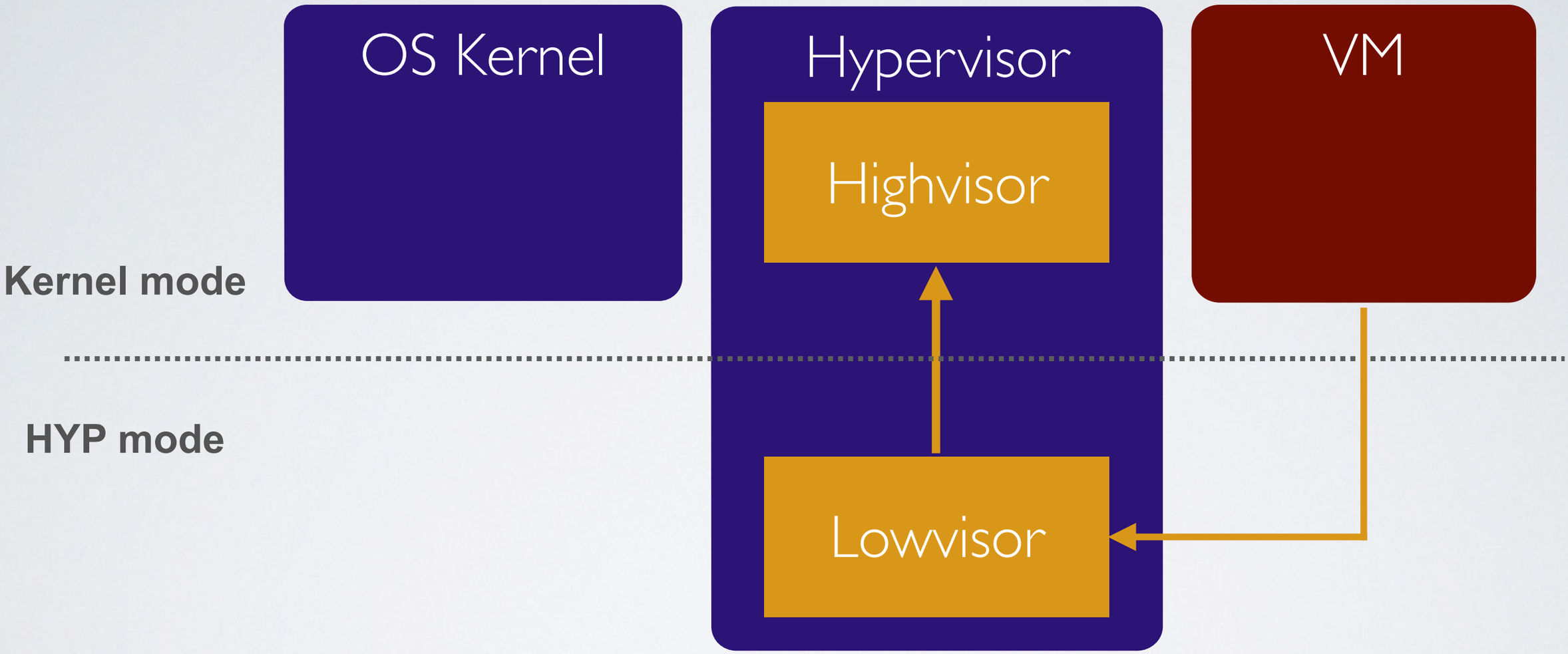
CPU Virtualization



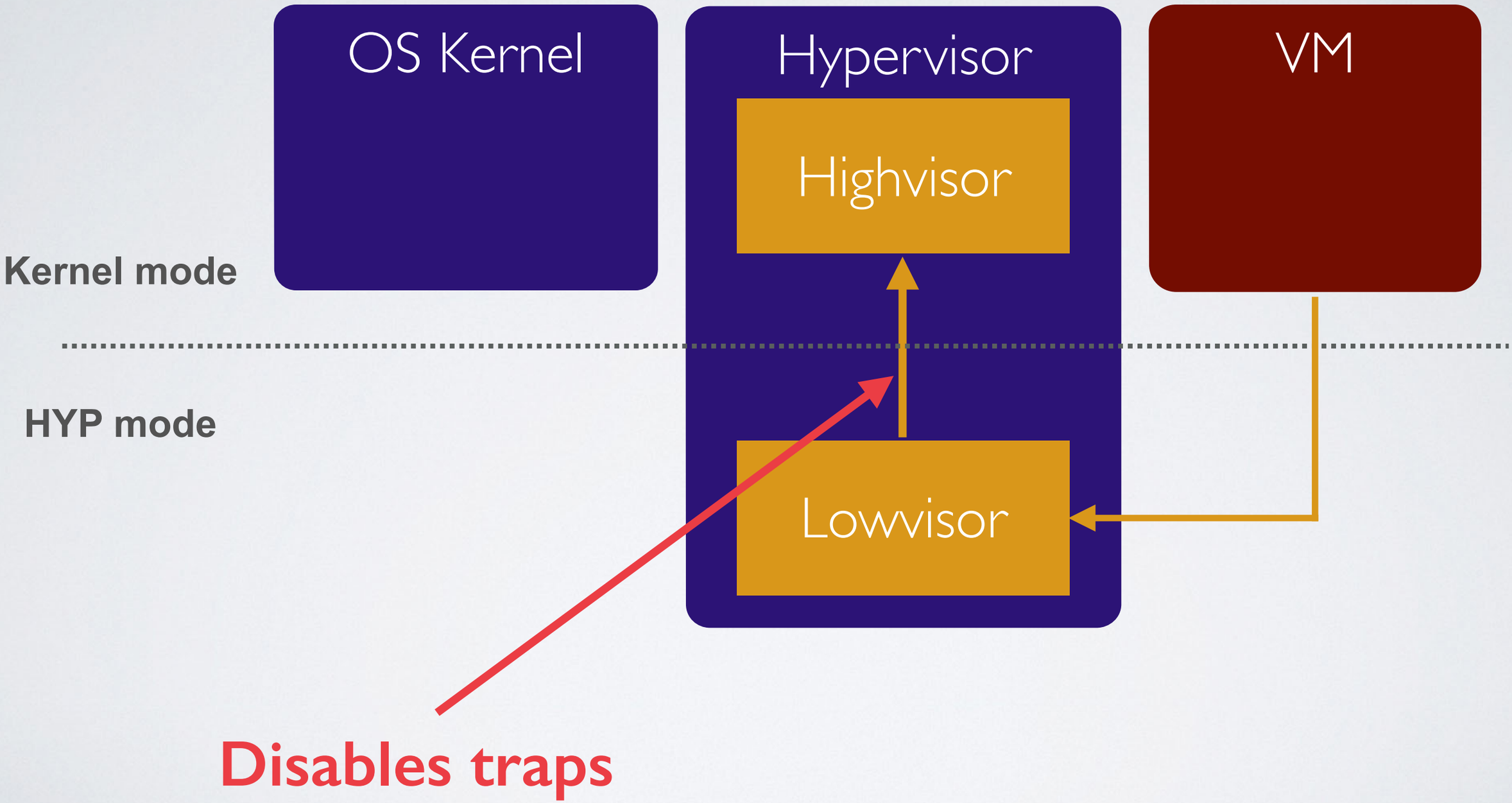
CPU Virtualization



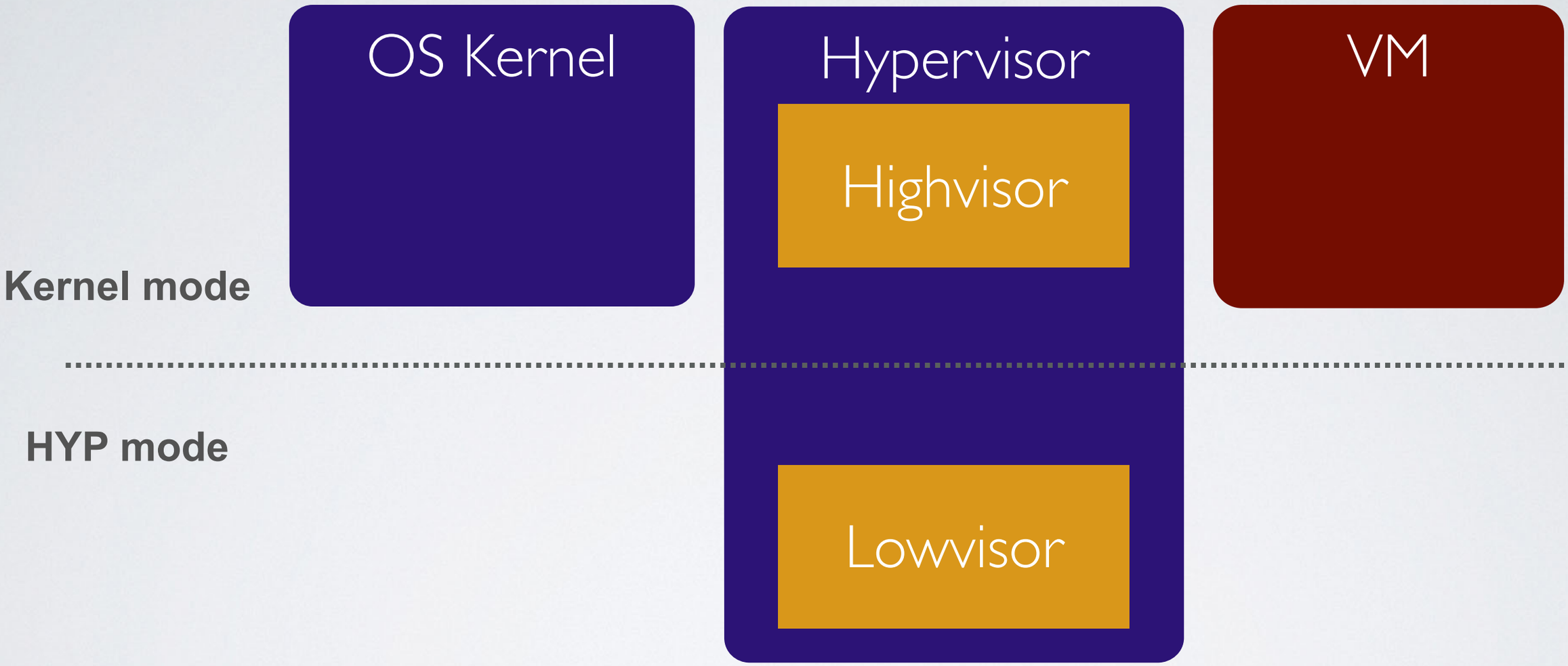
CPU Virtualization



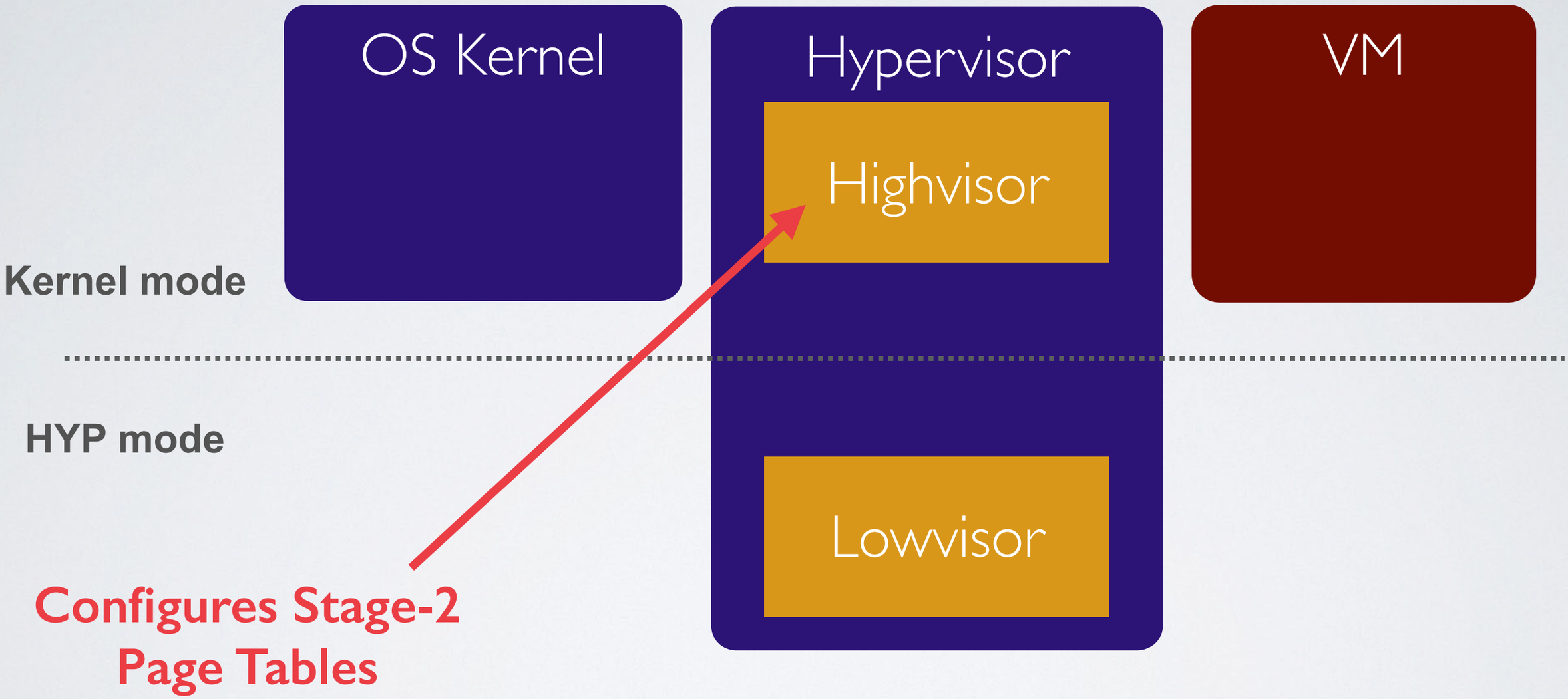
CPU Virtualization



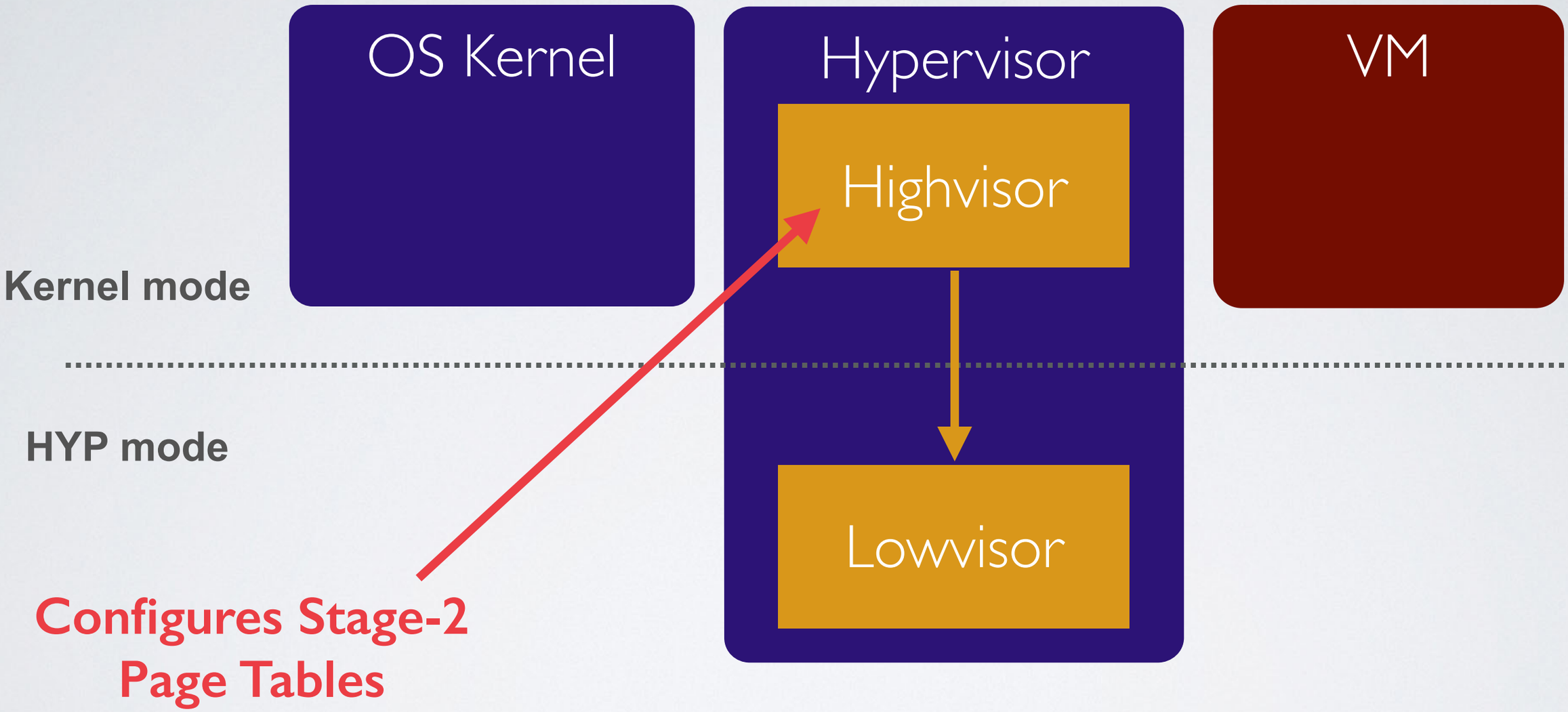
Memory Virtualization



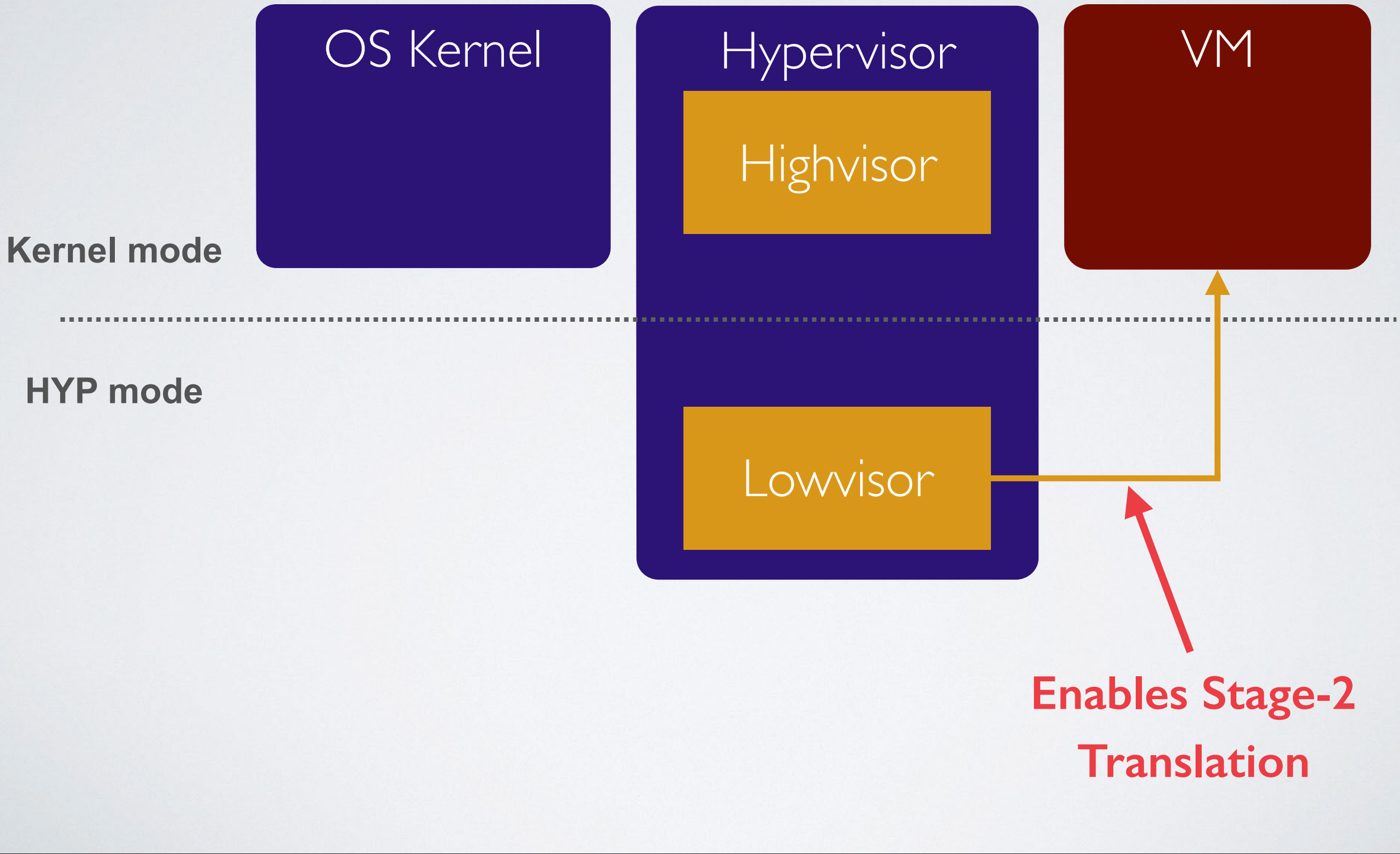
Memory Virtualization



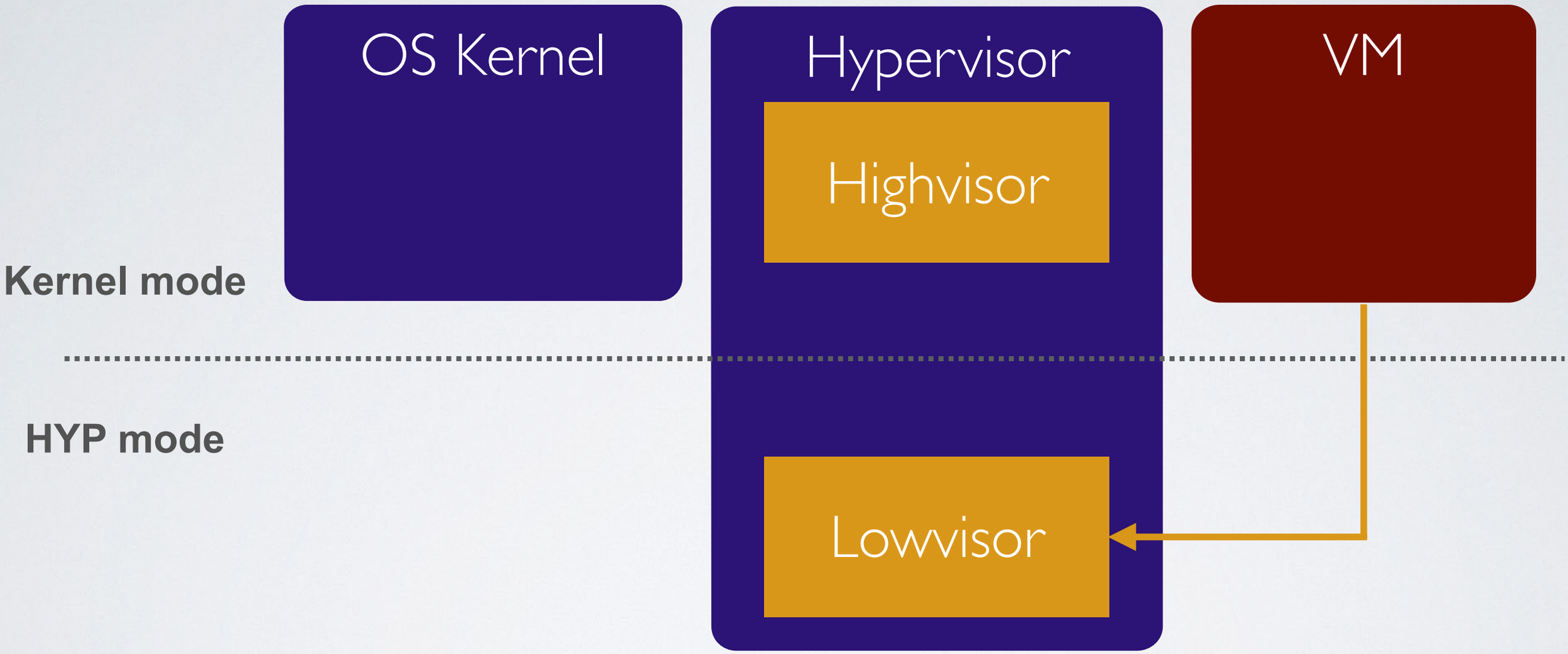
Memory Virtualization



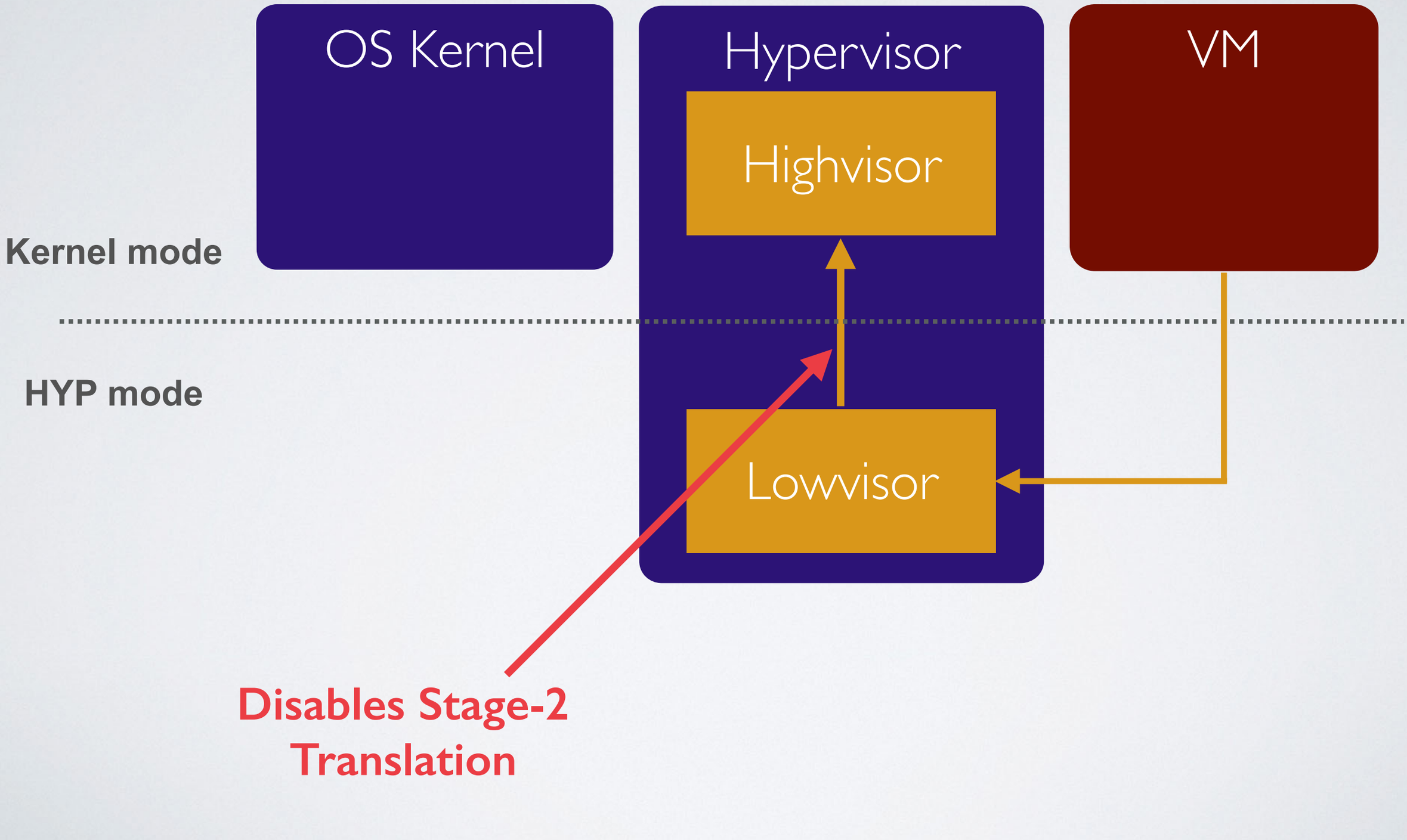
Memory Virtualization



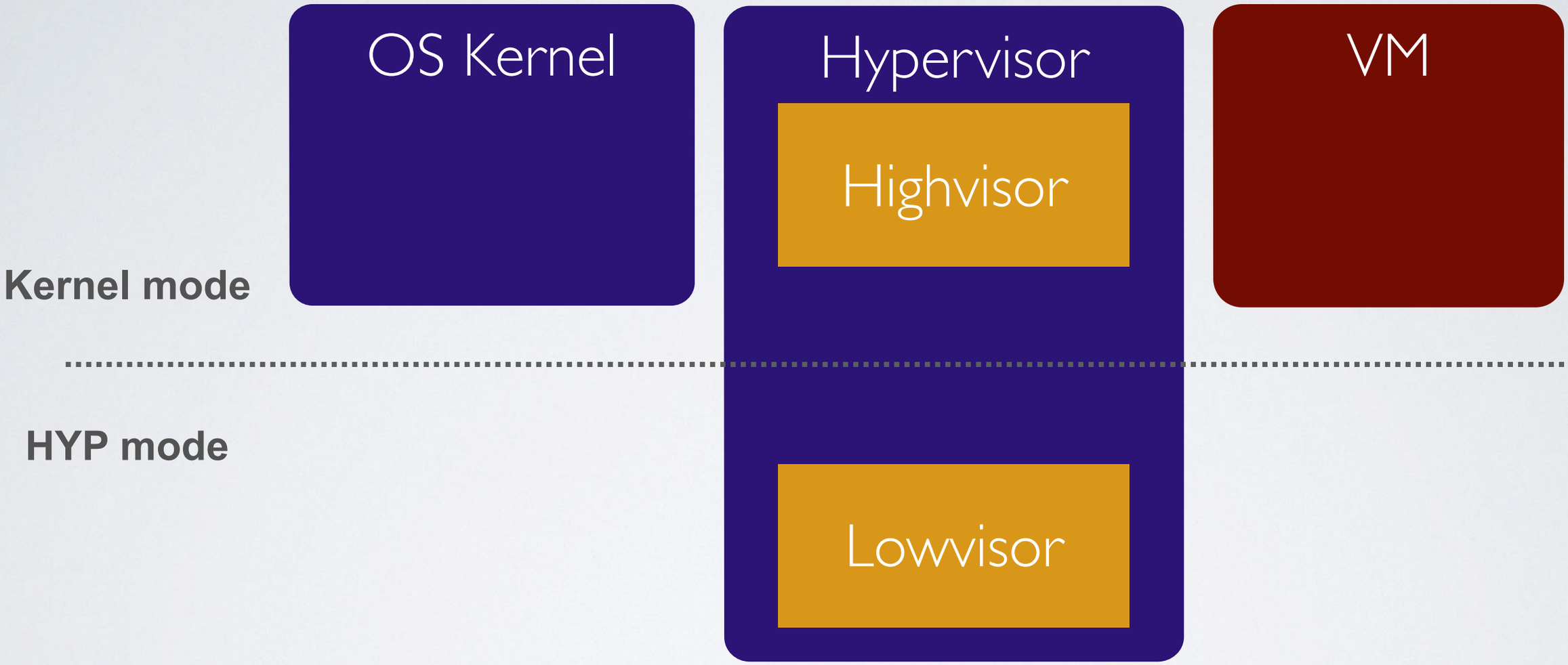
Memory Virtualization



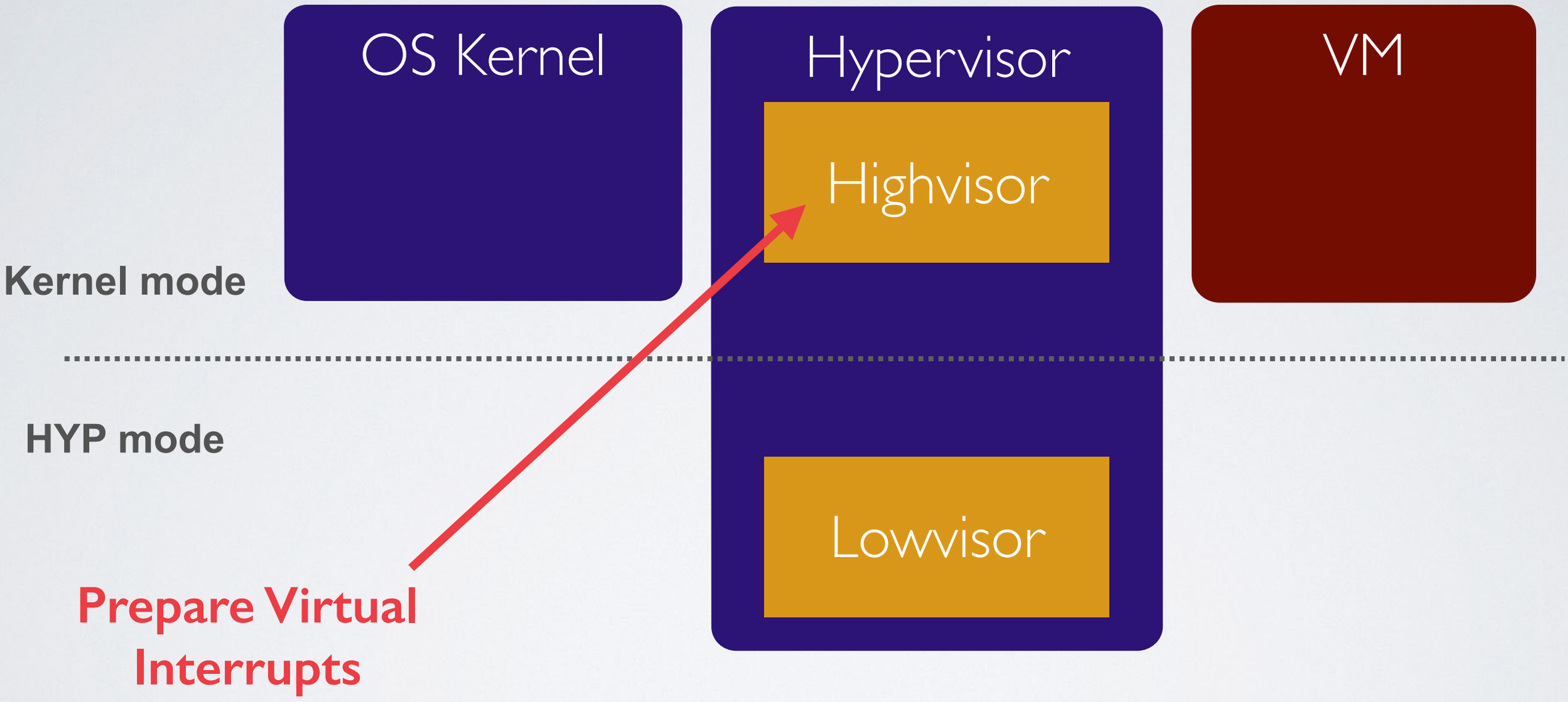
Memory Virtualization



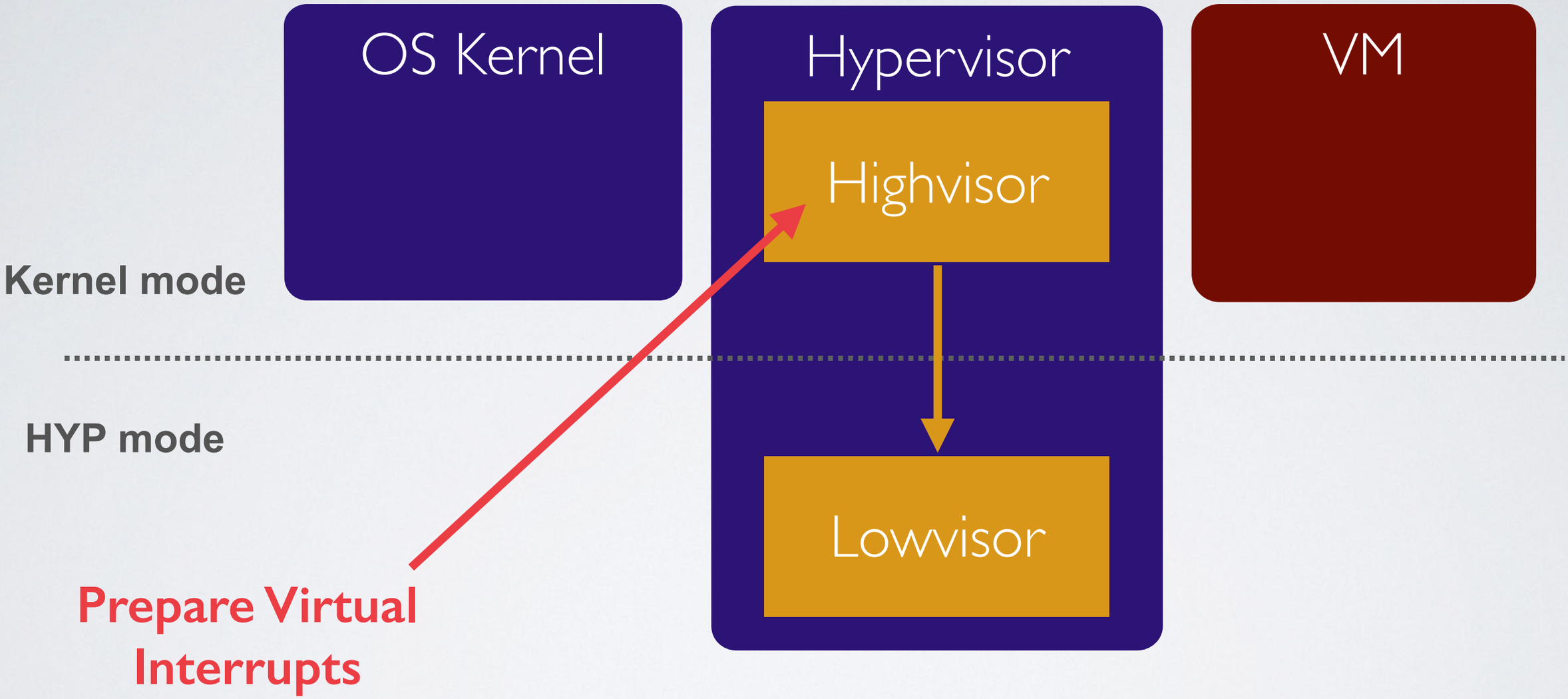
Interrupt Virtualization



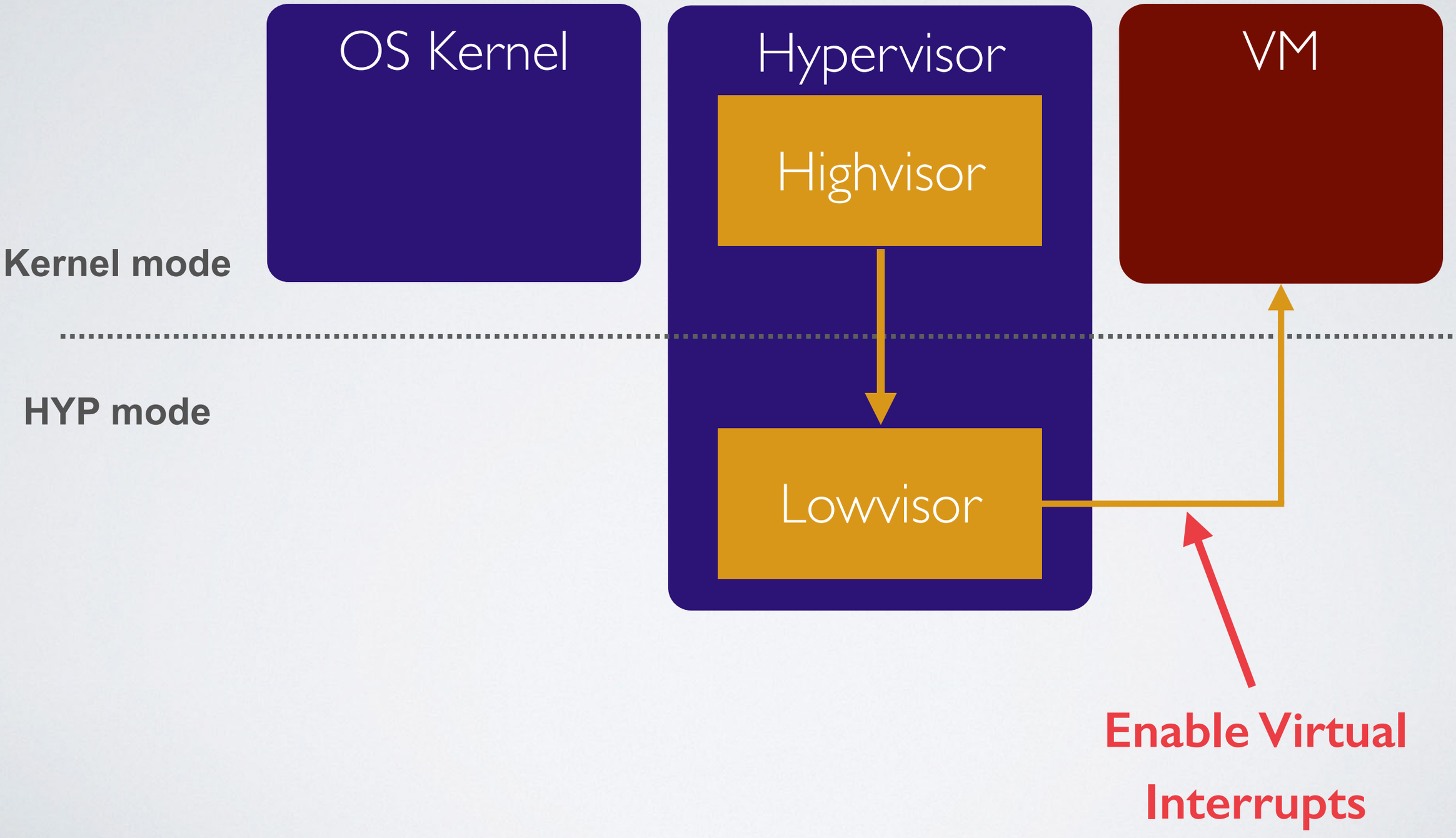
Interrupt Virtualization



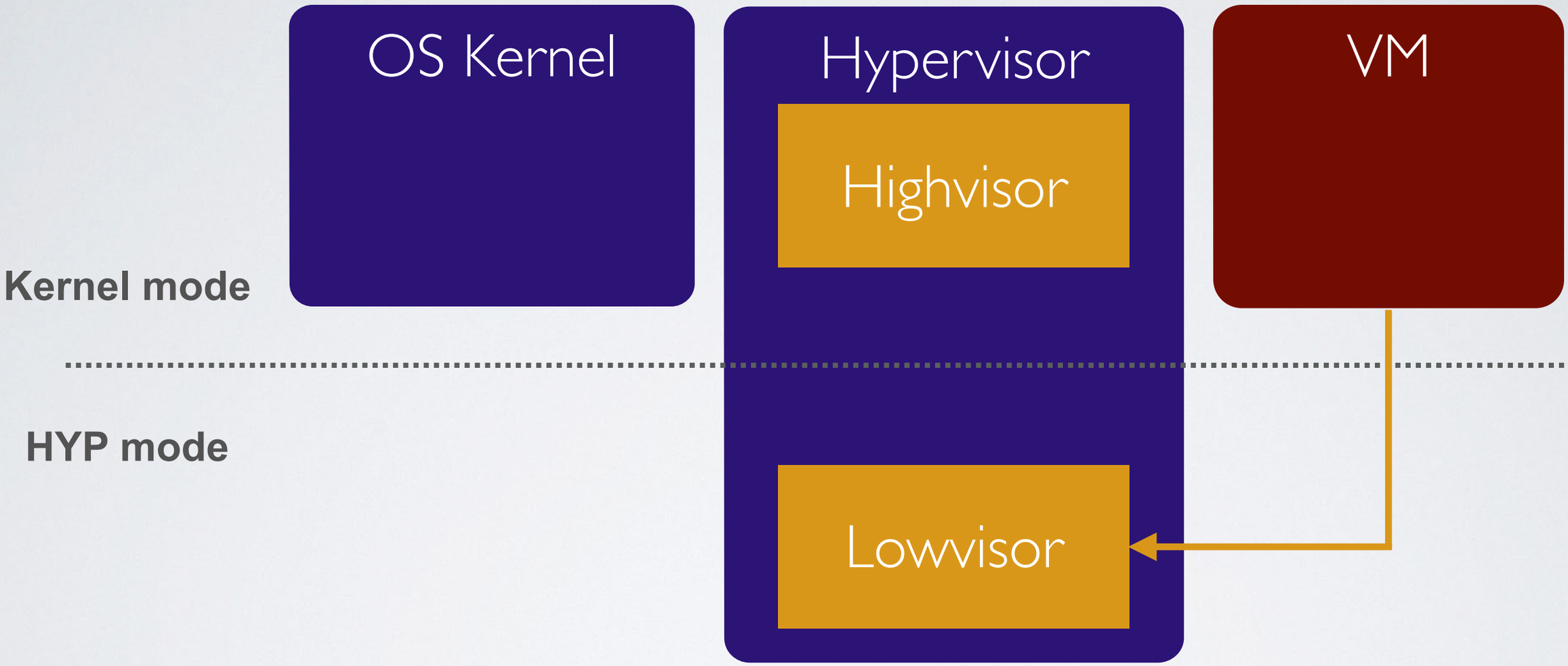
Interrupt Virtualization



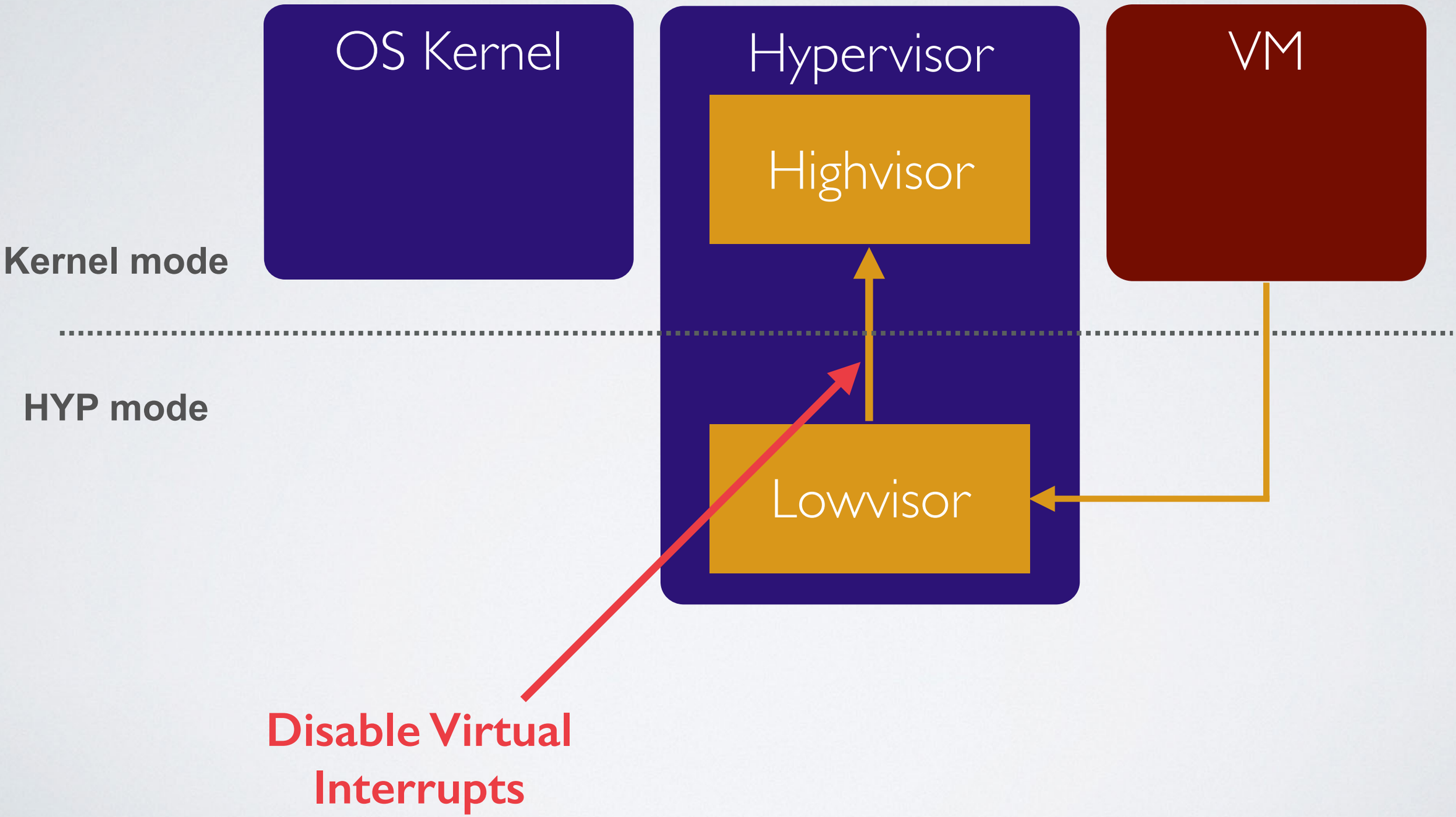
Interrupt Virtualization



Interrupt Virtualization



Interrupt Virtualization



Split-Mode Virtualization

Pros

- Existing OS functionality
- Existing hardware support
- HYP mode for virtualization

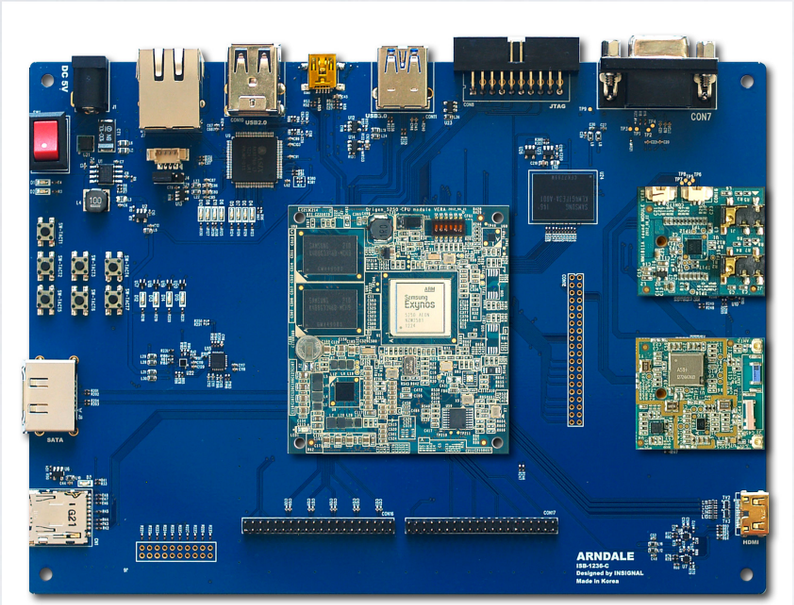
Cons

- Double-trap on world-switch

Implementation

- Implemented KVM/ARM using split-mode virtualization
- Merged into mainline Linux v3.9

Experimental Setup



Arndale Board - Exynos 5250
Dual-Core Cortex A-15 @ 1.4 GHz
eSATA SSD

MacBook Air
Dual-Core Intel i7 @1.8 GHz
SATA SSD

Experimental Setup

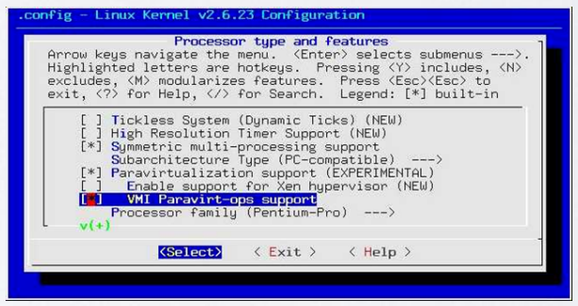
ubuntu

version 12.10
(Quantal Quetzal)

Linux™

Kernel version 3.10

Same amount of memory
across all runs

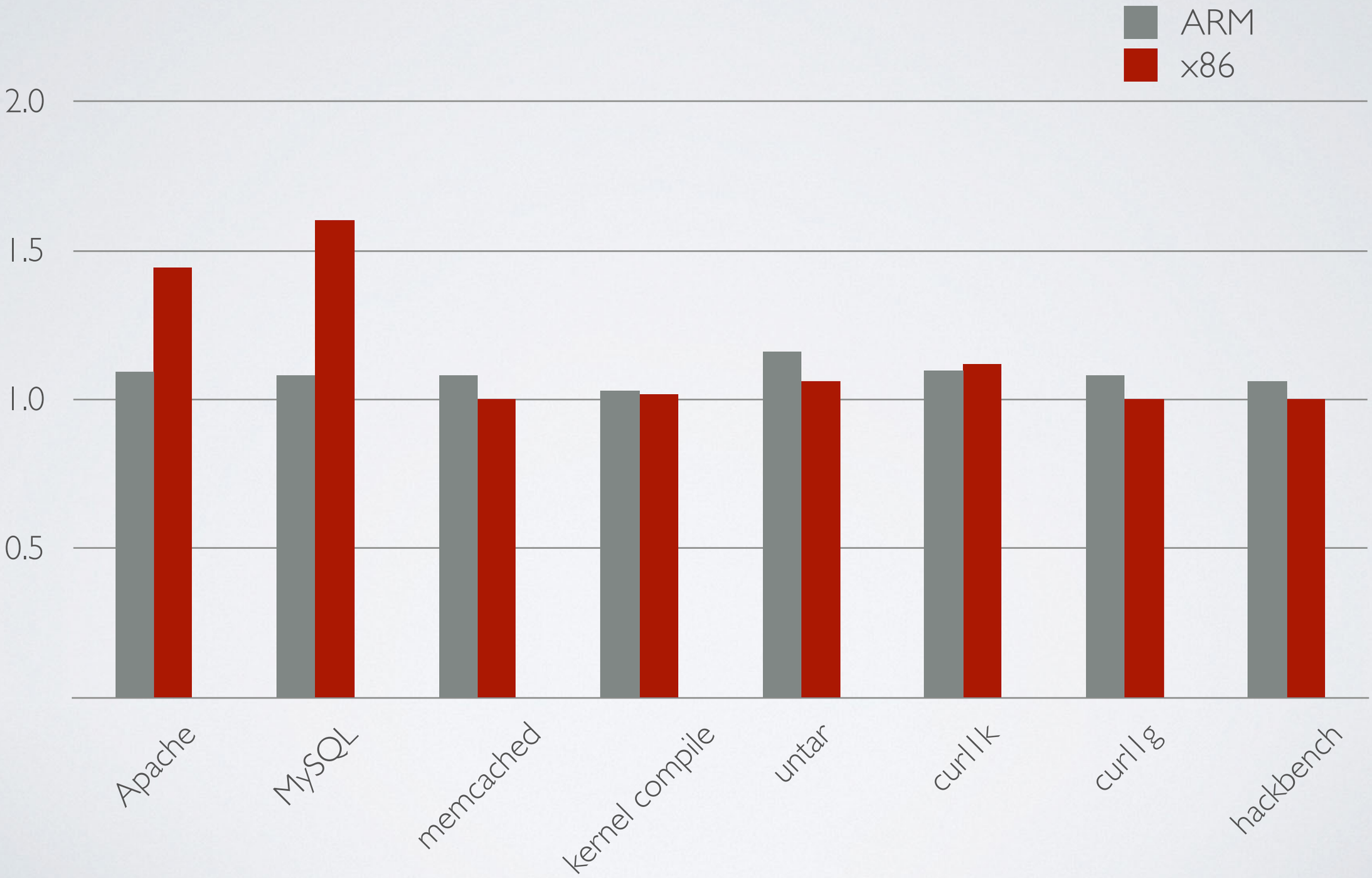


Same kernel config

Micro Benchmarks

Test	ARM (cycles)	x86 (cycles)
Trap	27	632
Hypercall	5,326	1,336
Send virtual IPI	14,366	17,138
Ack virtual IPI	427	2,043

Macro Benchmarks



Outline

- ARM Virtualization Extensions
- Comparison to x86
- Split-Mode Virtualization
- Results
- **Experiences**

Maintainability is key

Maintainability is key

Genuine fear of “Dump and Run”

arch/arm

mach-aaec2000
mach-at91
mach-bcm
mach-bcmring
mach-clps711x
mach-davinci
mach-dove
mach-ebsa110
mach-ep93xx
mach-footbridge
mach-gemini
mach-h720x
mach-highbank
mach-imx
mach-integrator
mach-iop13xx
mach-iop32x
mach-iop33x
mach-ixp2000
mach-ixp23xx
mach-ixp4xx
mach-keystone
mach-kirkwood
mach-ks8695
mach-l7200
mach-lh7a40x
mach-loki
mach-mmp
mach-msm
mach-mv78xx0
mach-mvebu
mach-mx1
mach-mx2
mach-mx25
mach-mx3
mach-mxc91231
mach-netx
mach-nomadik
mach-ns9xxx
mach-omap1
mach-omap2
mach-orion5x
mach-picoxcell
mach-pnx4008
mach-prima2
mach-pxa
mach-realview
mach-rockchip
mach-rpc
mach-s3c2400
mach-s3c2410
mach-s3c2412
mach-s3c2440
mach-s3c2442
mach-s3c2443
mach-s3c24a0
mach-s3c6400
mach-s3c6410
mach-s5pc100
mach-sa1100
mach-shark
mach-shmobile
mach-socfpga
mach-spear
mach-sti
mach-stmp378x
mach-stmp37xx
mach-sunxi
mach-tegra
mach-u300
mach-ux500
mach-versatile
mach-vexpress
mach-virt
mach-vt8500
mach-w90x900
mach-zynq

Trust

Trust

It's about the people

Linus Torvalds

Russell King
arch/arm

Avi Kivity
virt/kvm

...

Will Deacon
(ARM Ltd.)

Marc Zyngier
(ARM Ltd.)

...

Conclusion

- KVM/ARM was the first system to run unmodified guests on hardware
- Facilitated by split-mode virtualization
- Performance within 10% of native execution
- Implemented KVM/ARM and upstreamed to Linux
- Commercially supported and maintained

Use and Contribute

- Web: http://systems.cs.columbia.edu/projects/kvm-arm
- Code: https://github.com/torvalds/linux
- Mailing list: kvmarm@lists.cs.columbia.edu
- IRC: #kvm-arm on freenode