presented by

HUAWEI

Implementation of Hypervisor in UEFI Firmware

Spring 2017 UEFI Seminar and Plugfest

March 27 - 31, 2017

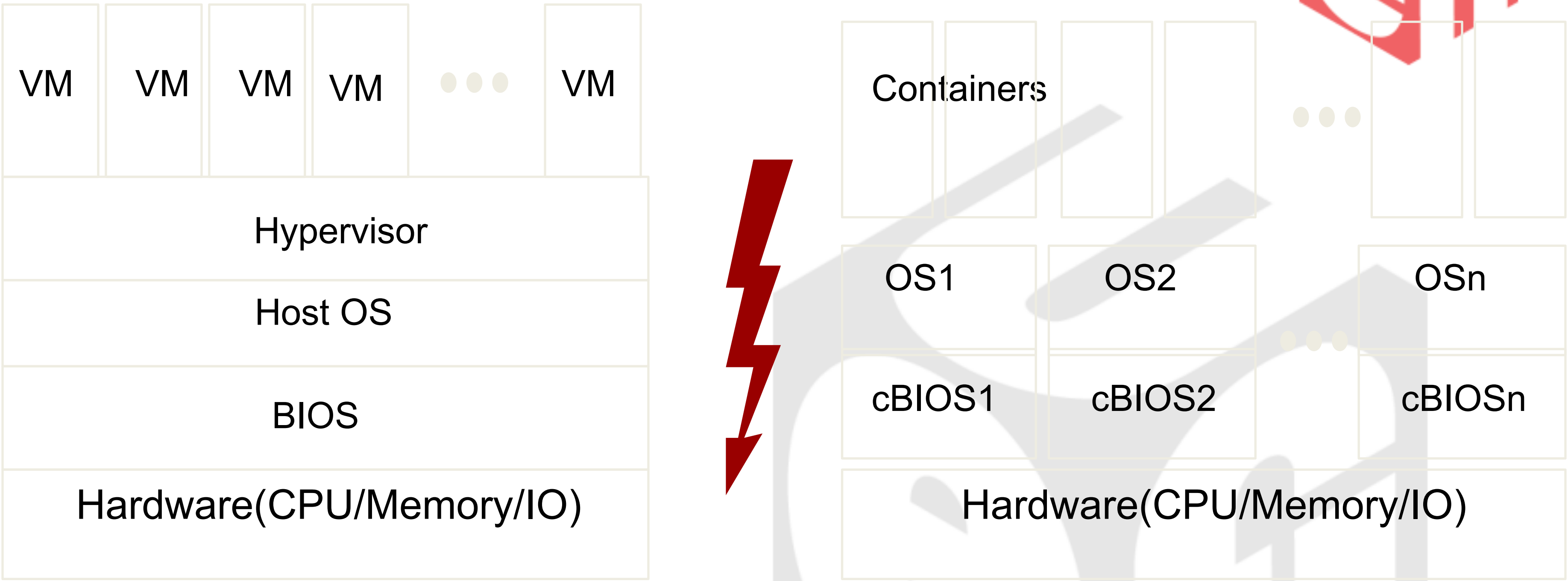
Presented by Kangkang Shen (Huawei)

Agenda

- Introduction
- Background
- Virtual UEFI (vUEFI) and Virtual ACPI (vACPI)
- Firmware Implementation
- Firmware for Cloud Computing
- Sample usage model

Introduction and motivation

Introduction (Container)



VM VM VM VM ••• VM

Hypervisor

Host OS

BIOS

Hardware(CPU/Memory/IO)

Containers •••

OS1 OS2 ••• OSn

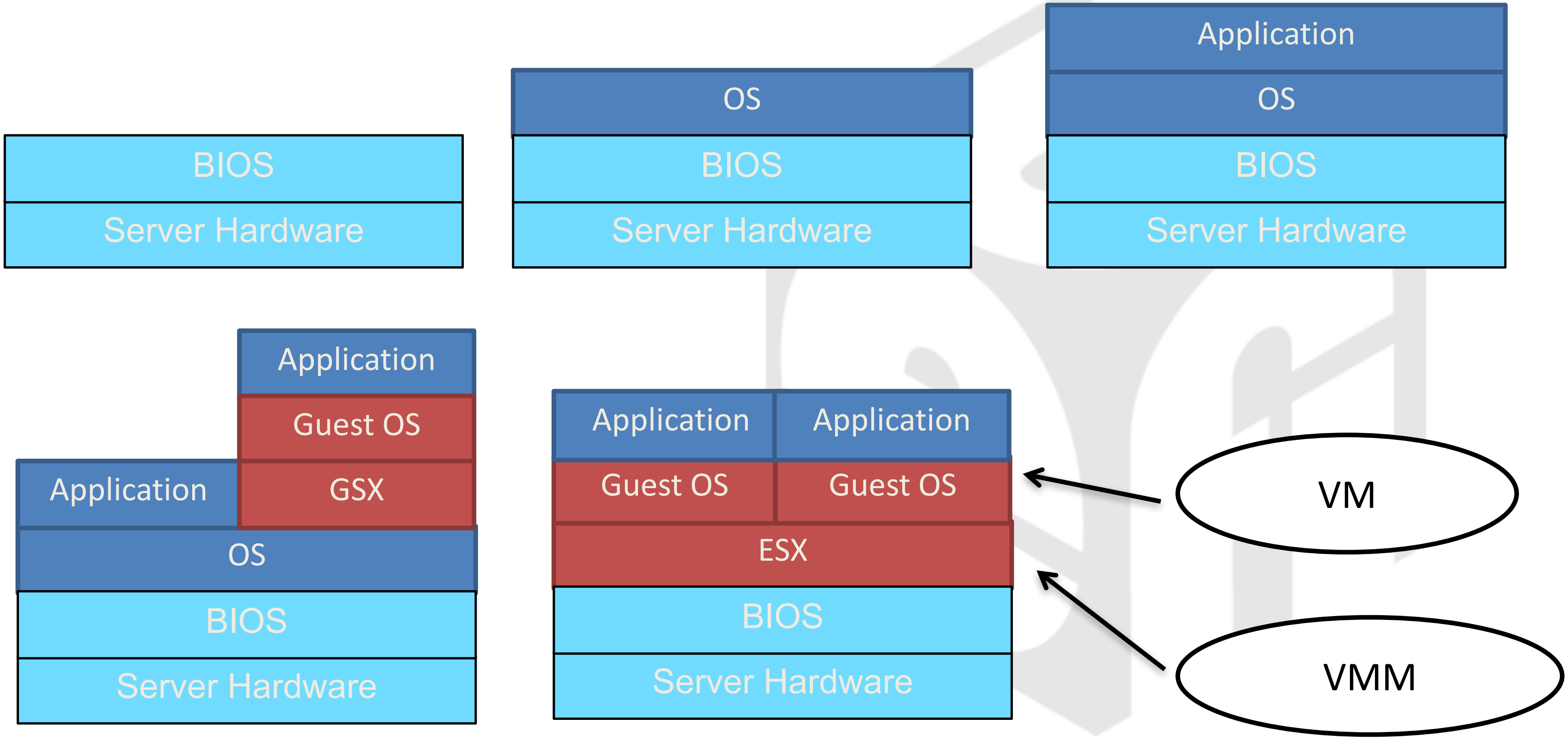
cBIOS1 cBIOS2 cBIOSn

Hardware(CPU/Memory/IO)

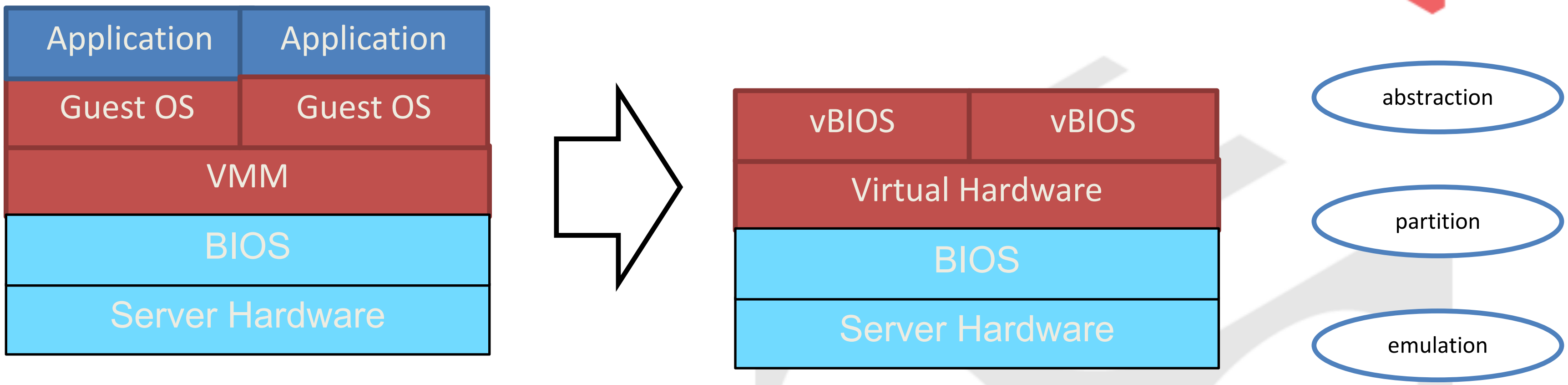
1. As the number of CPU increases the complexity of OS/Hypervisor increases dramatically.
2. Container provided an alternative solution for hypervisor, but it have security issues.
3. We can implement hardware assisted virtualization layer in firmware; generate multiple virtual BIOS interfaces for operating systems which host containers

Virtualization Background

Typical Virtualization Architecture



Standard vBIOS interface for VM



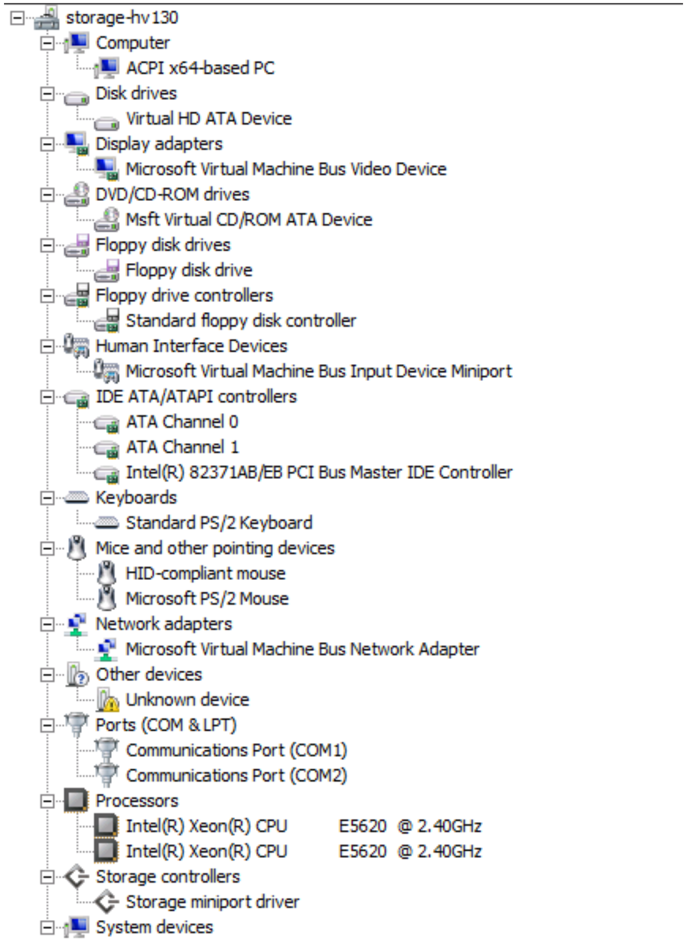
- ❑ A virtual machine (VM) is a software implementation of a machine (i.e. a computer) that executes programs like a physical machine
- ❑ The goal of vBIOS is to be able to allocate resources to build virtual machine such that the user can install an OS with vUEFI interface and the OS can configure and use the virtual hardware with vACPI.
- ❑ By introduce standards, we try to separate data and control panel.

Virtual UEFI (vUEFI) and Virtual ACPI (vACPI)

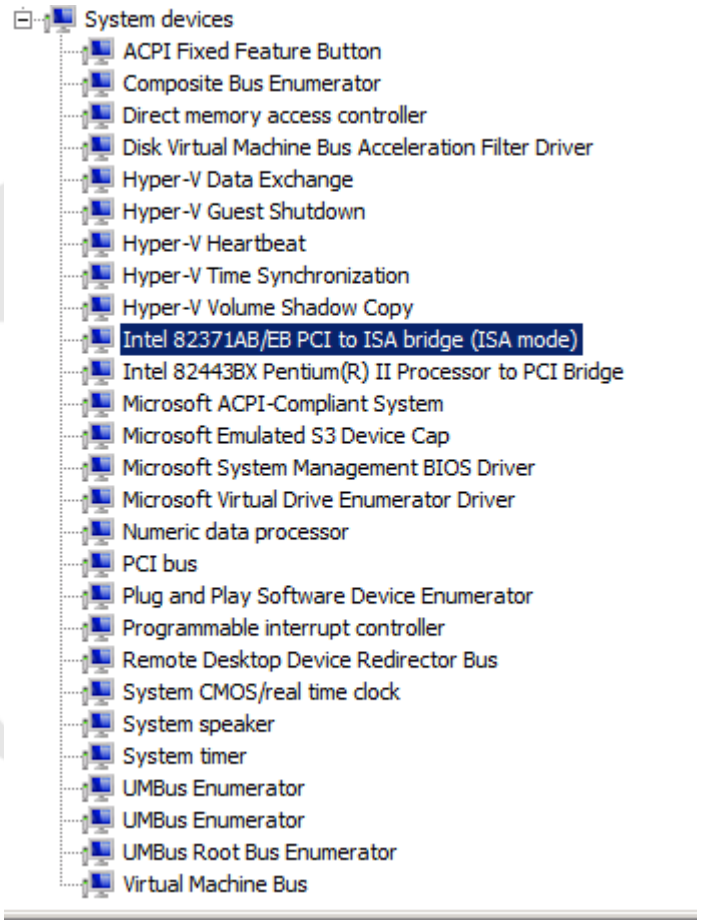
Virtual UEFI

- Based on customer defined profile, VMM allocate resources for vBIOS
- Specify Initial Power On CPU
- Define virtual memory instead of detecting physical memory
- Connect to virtual network and storage device
- Current UEFI specification can basically meet these requirements. An open source project OVMF from Tianocore.org provides
  - Libraries and drivers related to virtual machines
  - An entire firmware implementation with supports UEFI on open source virtual machines.

Device Manager in Guest Window OS



- storage-hv130
  - Computer
    - ACPI x64-based PC
  - Disk drives
    - Virtual HD ATA Device
  - Display adapters
    - Microsoft Virtual Machine Bus Video Device
  - DVD/CD-ROM drives
    - Msft Virtual CD/ROM ATA Device
  - Floppy disk drives
    - Floppy disk drive
  - Floppy drive controllers
    - Standard floppy disk controller
  - Human Interface Devices
    - Microsoft Virtual Machine Bus Input Device Miniport
  - IDE ATA/ATAPI controllers
    - ATA Channel 0
    - ATA Channel 1
    - Intel(R) 82371AB/EB PCI Bus Master IDE Controller
  - Keyboards
    - Standard PS/2 Keyboard
  - Mice and other pointing devices
    - HID-compliant mouse
    - Microsoft PS/2 Mouse
  - Network adapters
    - Microsoft Virtual Machine Bus Network Adapter
  - Other devices
    - Unknown device
  - Ports (COM & LPT)
    - Communications Port (COM1)
    - Communications Port (COM2)
  - Processors
    - Intel(R) Xeon(R) CPU E5620 @ 2.40GHz
    - Intel(R) Xeon(R) CPU E5620 @ 2.40GHz
  - Storage controllers
    - Storage miniport driver
  - System devices



- System devices
  - ACPI Fixed Feature Button
  - Composite Bus Enumerator
  - Direct memory access controller
  - Disk Virtual Machine Bus Acceleration Filter Driver
  - Hyper-V Data Exchange
  - Hyper-V Guest Shutdown
  - Hyper-V Heartbeat
  - Hyper-V Time Synchronization
  - Hyper-V Volume Shadow Copy
  - Intel 82371AB/EB PCI to ISA bridge (ISA mode)
  - Intel 82443BX Pentium(R) II Processor to PCI Bridge
  - Microsoft ACPI-Compliant System
  - Microsoft Emulated S3 Device Cap
  - Microsoft System Management BIOS Driver
  - Microsoft Virtual Drive Enumerator Driver
  - Numeric data processor
  - PCI bus
  - Plug and Play Software Device Enumerator
  - Programmable interrupt controller
  - Remote Desktop Device Redirector Bus
  - System CMOS/real time clock
  - System speaker
  - System timer
  - UMBus Enumerator
  - UMBus Enumerator
  - UMBus Root Bus Enumerator
  - Virtual Machine Bus

Virtual ACPI

❑ACPI specification was originally proposed by Microsoft and owned by a small group of companies such that only limited virtualization information is available through Intel, Microsoft and HP.

❑ACPI has been widely adopted by the IT industry. In October 2013, the governance of ACPI has been changed and the specification is now managed by UEFI Forum.

❑In addition to Windows, many other Operating Systems and Hypervisors are now supporting ACPI and vACPI.

ACPI described virtual hardware hierarchy

- Instead of describe real hardware we need vACPI to describe customer virtual hardware.
- For PCI devices supporting SR-IOV, we should use VF instead of PF
- For multiple CPU architecture, we would like to see ACPI provides CPU topology information such that guest OS can control the CPU directly assuming the virtual machine only using CPU partition

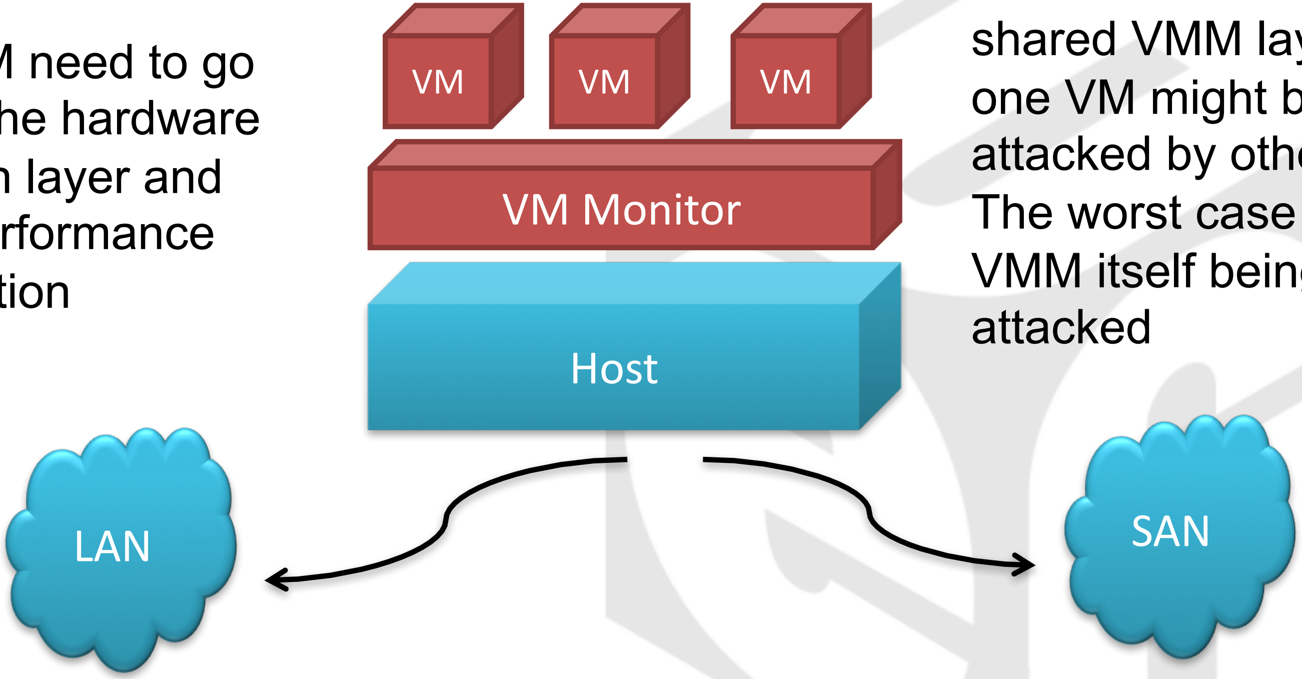
ACPI Power States

- S3
  - May not be able to put the CPU in power saving mode if the physical CPU is shared by multiple VMs. But we can avoid this problem if we use CPU partition. It is useful for migration
- S4
  - Good for quick boot
- C states: exit virtual processor from scheduler and wake back
- P states: voltage ignored, but frequency will affect scheduler
- D states: device control method can be encapsulated within ACPI AML code.

Firmware Implementation

Virtual Machine Challenges

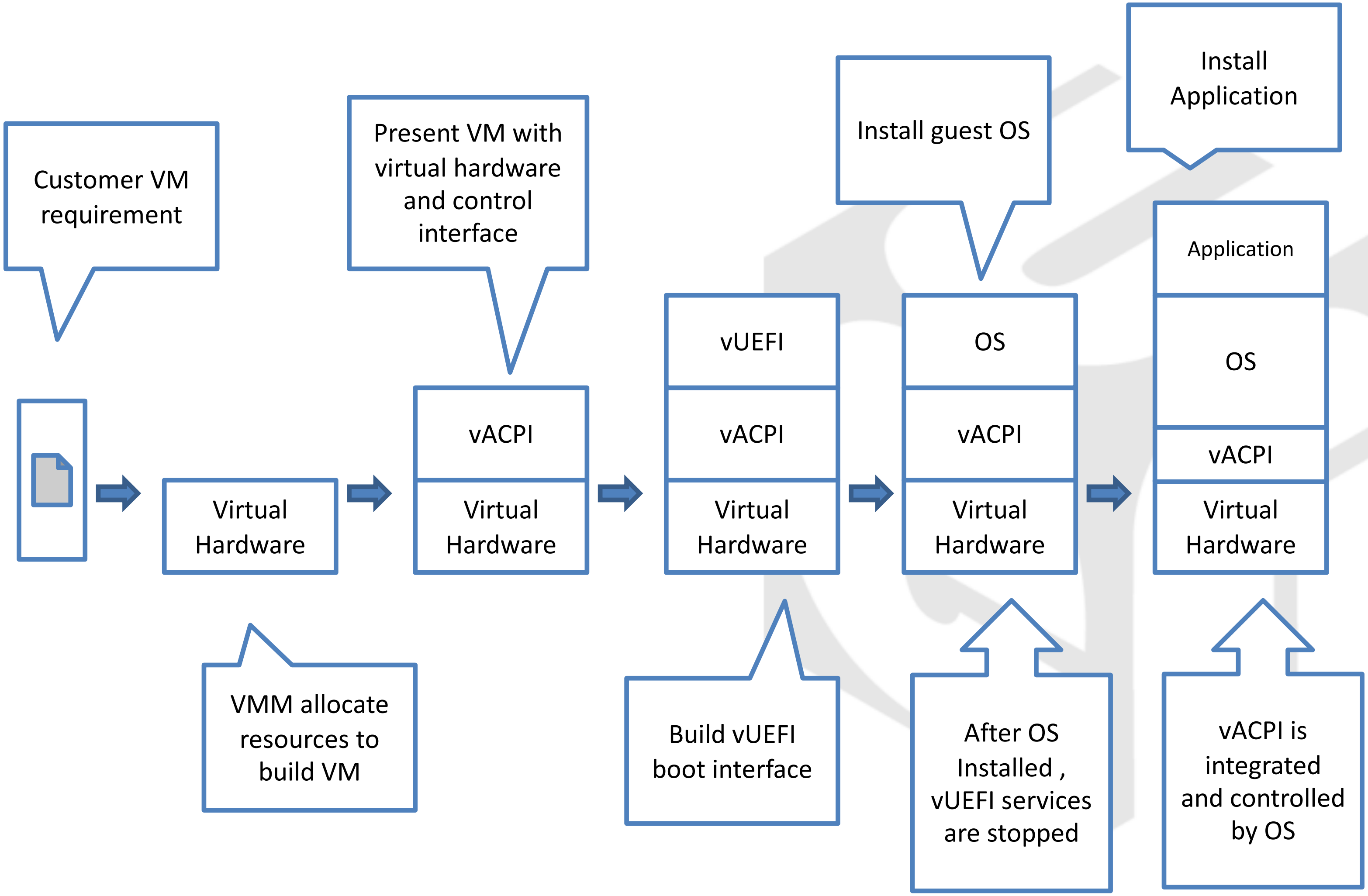
When VM need to go through the hardware emulation layer and cause performance deterioration



Because of the shared VMM layer, one VM might be attacked by other VM. The worst case is that VMM itself being attacked

If we can avoid enter VMM during VM operation, we should be able to solve both performance and security problem

Process to run an application



Customer VM requirement

Virtual Hardware

VMM allocate resources to build VM

Present VM with virtual hardware and control interface

vACPI

Virtual Hardware

vUEFI

vACPI

Virtual Hardware

Build vUEFI boot interface

Install guest OS

OS

vACPI

Virtual Hardware

After OS Installed , vUEFI services are stopped

Install Application

Application

OS

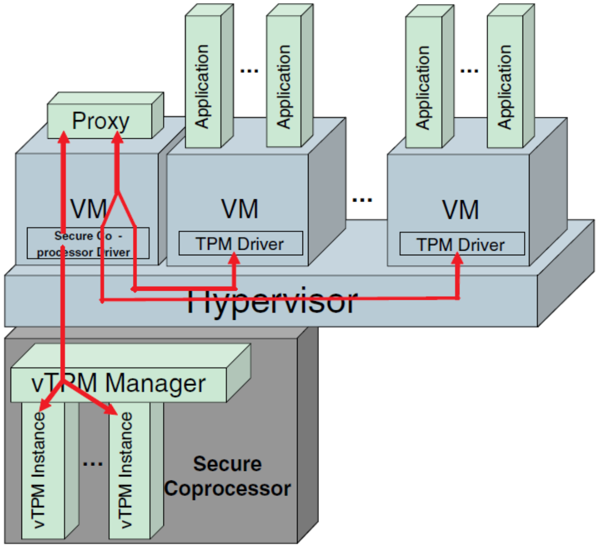
vACPI

Virtual Hardware

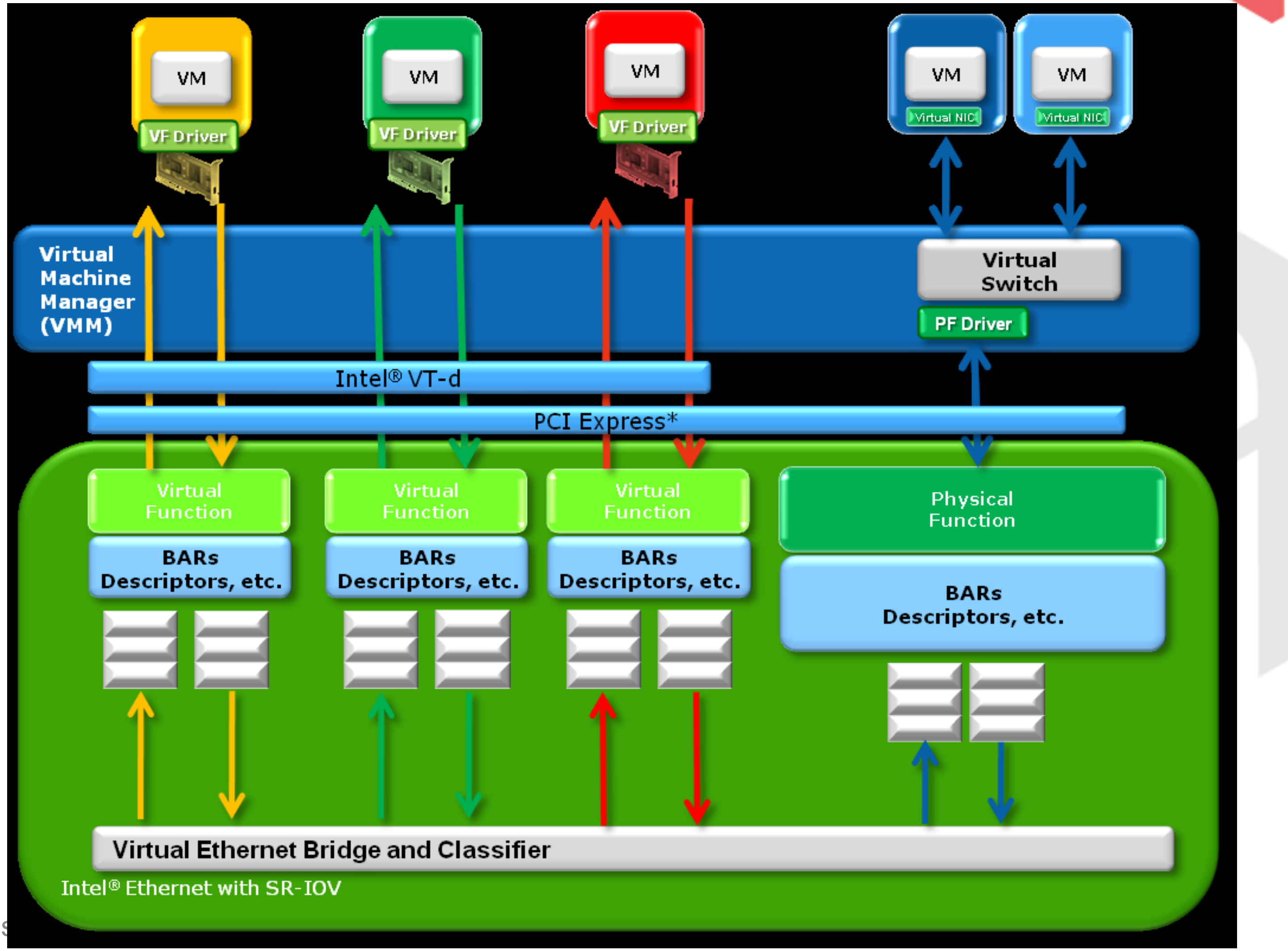
vACPI is integrated and controlled by OS

Industry solutions for Virtualization

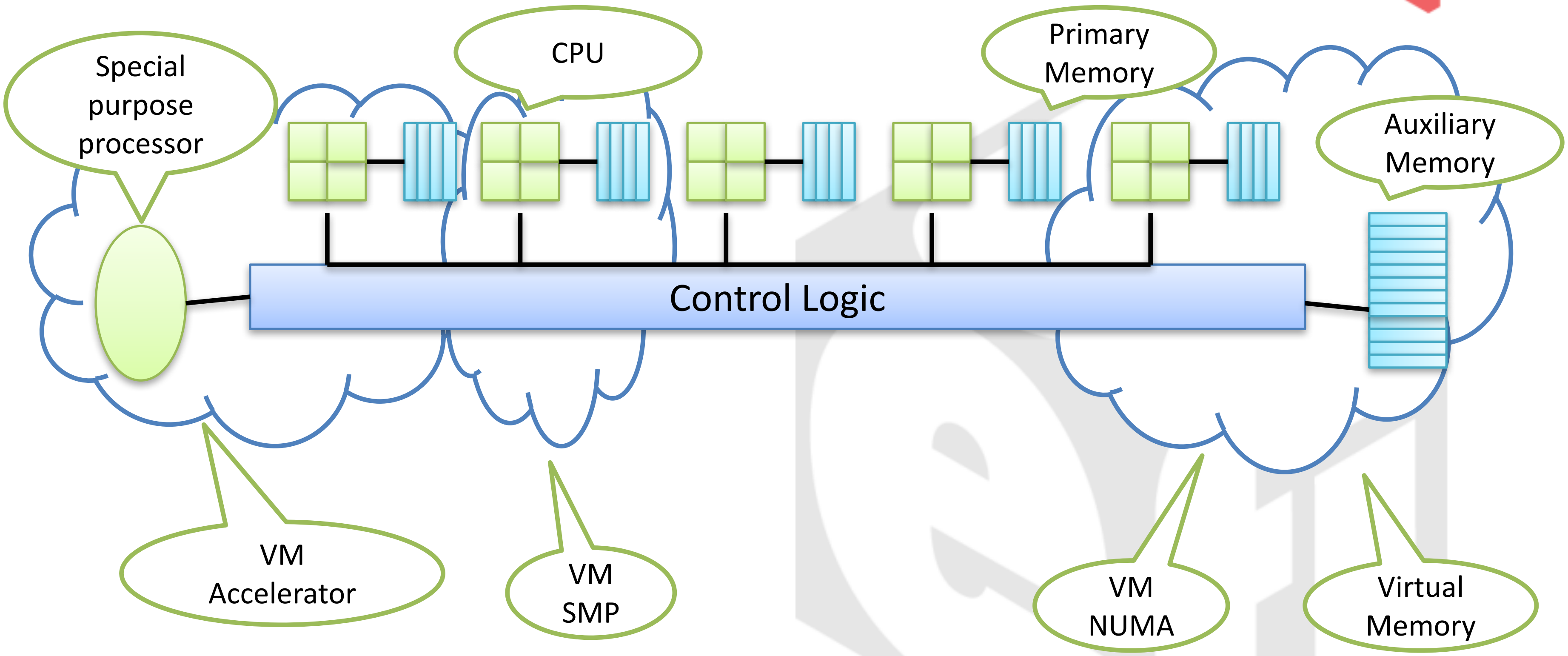
- Hardware assisted virtualization
  - Intel VT-X
  - Intel VT-d
  - Intel VT-C
- Trusted Computing
  - IBM vTPM
- Standards
  - SR-IOV
  - MR-IOV



Industry efforts to improve performance (PCI-SIG)



CPU and Memory Topology



Various choice to build a VM with 4 CPUs

VM Migration

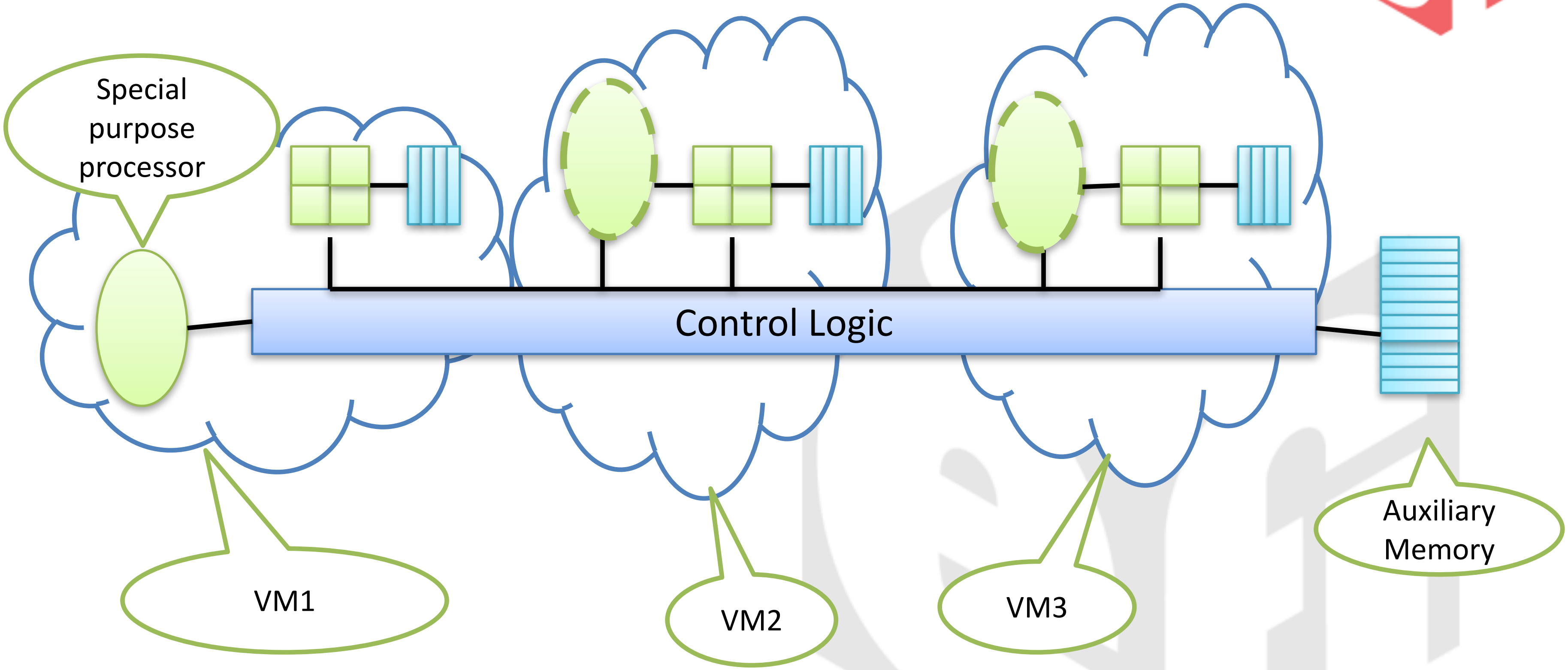
- Emulated device
- Para virtualized
- Physical
  - –Build VM with additional hardware resources
  - –Using existing Server RAS features such as hot plug, error reporting, etc to facility migration

Hot Plug

- ACPI
  - Well defined and exposed to OS
- PCI-SIG
  - SHPC – Standard Hot plug Controller- complicated
- Vendor Specific not good for various OS

Firmware for Cloud Computing - Cloud BIOS

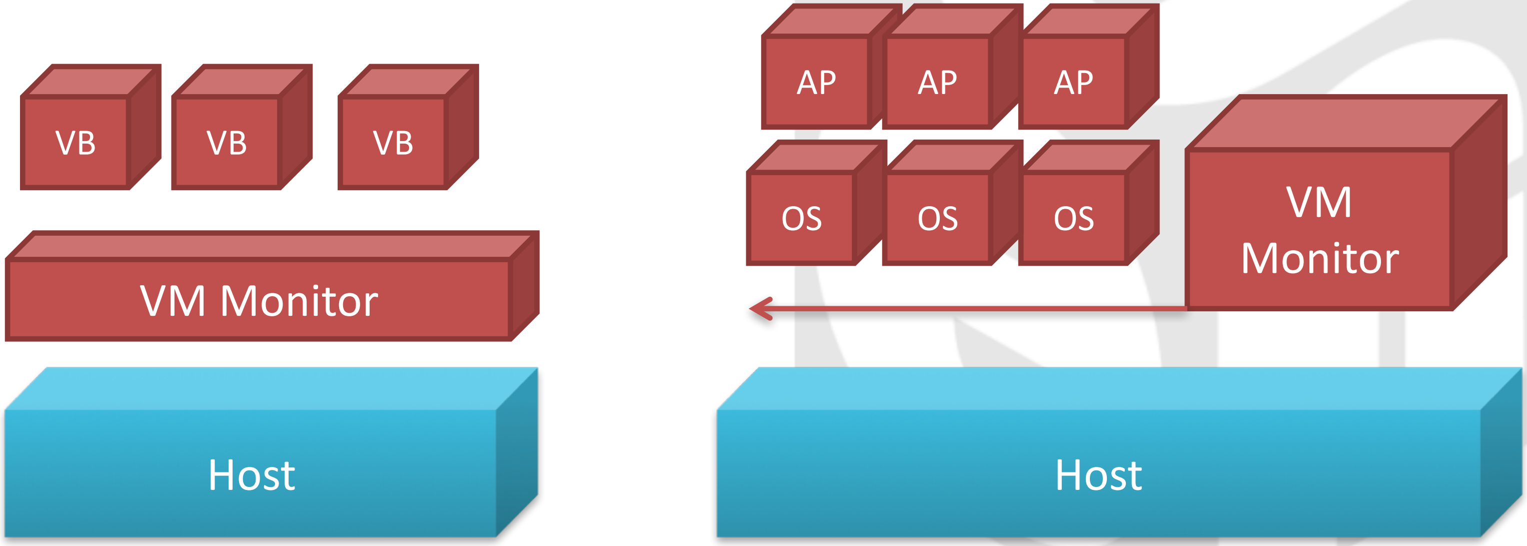
Accelerator Virtualization



Accelerator is used on task intensive purpose. There is little meaning to have virtual accelerator. Instead we can realize accelerator virtualization through hot plug and migration

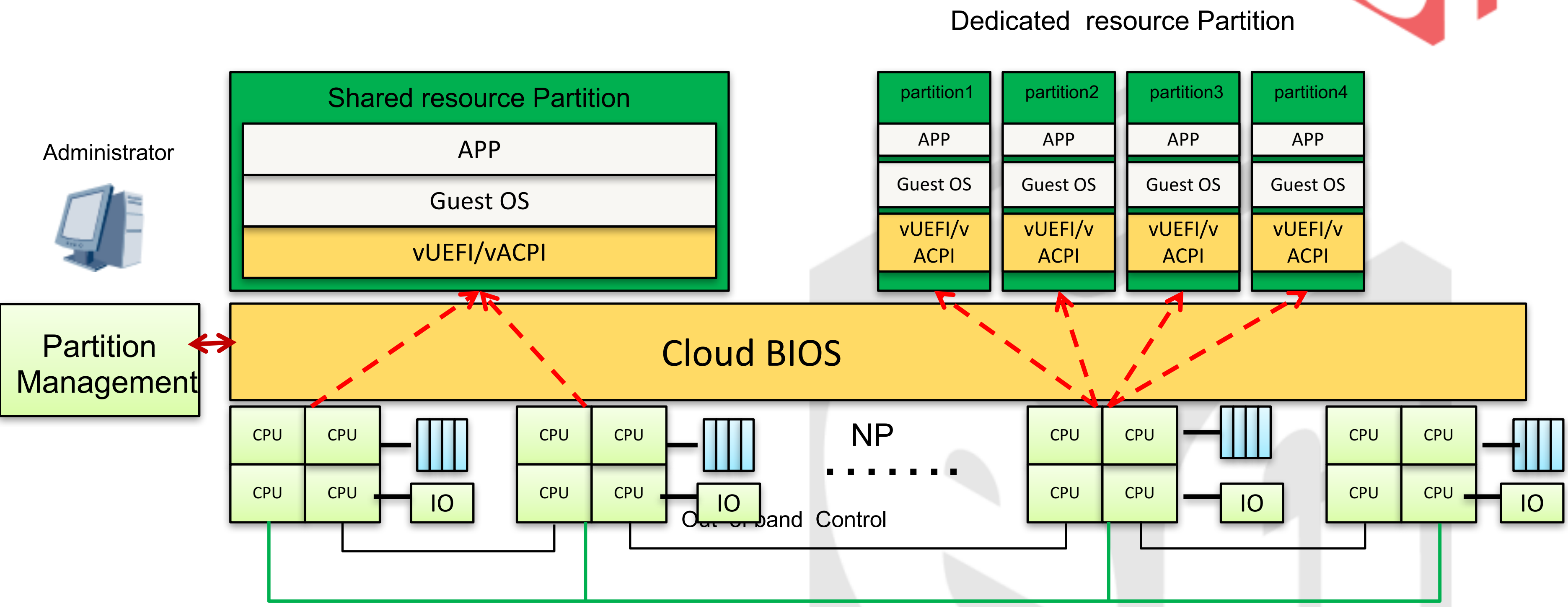
Virtualization without Hypervisor

- vBIOS is used to install OS
- vACPI pass virtual hardware hierarchy and control methods to OS
- Guest OS operates through hardware or firmware assisted virtualization



Sample usage model

Combine multi processor to provide high performance



Dedicated resource Partition

Administrator

Shared resource Partition

APP

Guest OS

vUEFI/vACPI

partition1

partition2

partition3

partition4

APP

Guest OS

vUEFI/vACPI

Partition Management

Cloud BIOS

CPU

IO

NP

Out-of-band Control

Benefits and advantages

1、Save hypervisor software cost and make the product more competitive

2、Product can be flexibly arranged based on usage cases

3、Physical partition and hardware assisted virtualization increase reliability, availability and security

4、Bare metal，avoid performance penalty.

Thanks for attending the Spring 2017 UEFI Seminar and Plugfest

For more information on the UEFI Forum and UEFI Specifications, visit http://www.uefi.org

presented by

HUAWEI