CIM System Virtualization Model

Steve Hand
DMTF System Virtualization Partitioning and Clustering Workgroup (SVPC)
Symantec

Thanks to Ron Goering, Co-Chair SVPC, Distinguished Engineer
IBM
The information in this presentation represents a snapshot of work in progress within the DMTF.

This information is subject to change. The Standard Specifications remain the normative reference for all information.

For additional information, see the Distributed Management Task Force (DMTF) Web site.

http://www.dmtf.org/standards/smash
System Virtualization WG

- System Virtualization Partitioning and Clustering (SVPC) DMTF Workgroup with participation from EMC (VMware), Microsoft, IBM, HP, Sun, Novell, XenSource, Hitachi, Intel and others
  - Weekly calls (Thurs 9 PDT – details on website: http://www.dmtf.org/apps/org/workgroup/redundancy/)

- Work ongoing to produce CIM model (CIM profile and associated CIM schema changes) for virtual systems and the virtual resources which compose them.
  - Leverage SMI-S profiles for storage virtualization

- Deliverables:
  - System Virtualization Model White Paper
  - System Virtualization Overview presentation
  - CIM schema request for changes with associated MOF –
    - Changes in CIM 2.15, 2.16, small updates in 2.17
  - CIM Profiles published as draft standards
    - Resource Allocation, Resource Capabilities abstract profiles
    - System Virtualization, Virtual System
    - “Device” profiles: Generic Device,
  - CIM Profiles work in progress
    - Processor, Memory, Block back disks, file back disks, Virtual Ethernet, VHBA, and Removable Media
    - Others in second phase
Virtual System Model Requirements

- **General Requirements:**
  - Enable management applications which are unaware of virtualization to manage virtual systems, i.e. once a ComputerSystem is created most management operations (list, install, configure) are enabled without requiring the management application to understand virtualization.
    - Make sure appropriate profiles from SMWG, SMI-S and others are applicable
    - Support the symmetry inherent in multiple layers of virtualization.
    - Model should be general and flexible enough to support known virtualization systems including partitioning and containers

- **ComputerSystem**
  - **Enumerate** virtual systems, resources, relationships on a particular platform.
  - **Create** Virtual System specifying resources (CPU, disk, I/O) and attributes about those resources (shared, virtualized, based on what platform resource)
    - Provide ability for management application to introspect the system at runtime to find out virtualization capabilities and resources supported.
    - Provide appropriate defaults wherever possible
  - **Delete** virtual system and return resources to platform.
  - **Modify** the resources that compose virtual system.

- **Virtual Resources**
  - Support **creation, modification, deletion and inventory of virtual resources**
  - Enable **mapping of virtual resources** to underlying resources (through as many layers of virtualization as needed)
    - For example: Customer is notified that particular physical disk is receiving intermittent errors. Customer would like to understand which virtual machines would be effected if the disk failed.
System Virtualization Related Profiles

DMTF Management Profiles Related to System Virtualization

Component
DSP1033
Profile Registration

Abstract
DSP1052
ComputerSystem

Scoped By
DSP1027
Power State Management
(Optional)

Abstract
DSP1022
Boot Control Profile
(Optional)

Component
DSP1042
System Virtualization

Component
DSP1004
Base Server
(Optional)

Component
DSP1026
System Memory
Profile
(Optional)

Component
DSP1012
CPU Profile
(Optional)

Component
DSP1014
Ethernet Port Profile
(Optional)

Component
DSP1045
Memory Resource
Virtualization
(Optional)

Component
DSP1044
CPU Resource
Virtualization
(Optional)

Component
DSP1004
Network Port
Resource Virtualization
(Optional)

Component
DSP1049
Storage Resource
Virtualization
(Optional)

Component
DSP1059
Generic Device
Resource Virtualization
(Optional)

Component
DSP1052
Virtual System

Other Component Profiles

Other Device Profiles
Virtual System Modeling Basics

- **Host System (or Host Computer System)** – In a virtualized computer system environment the computer system that contains resources from which Virtual Systems are constructed.

- **Virtual System (or Virtual Computer System)** – Computer Systems composed of partitioned, shared or virtualized resources presented from a host system. Terms also used for this concept are Virtual Machine, Hosted Computer, Child Partition, Logical Partition, Domain, Guest.

- **HostedDependency** is used to associate Virtual System with its Host System
  – HostedDependency may be used to associate “virtual” device with “host” device.

- **LogicalIdentity** is used when simple direct host device allocation is done to Virtual System (e.g. partitioning)
Resource Virtualization

- **Basic Principle of System Virtualization modeling**
  - Devices represented by core CIM classes, additional information in associated setting data

- **Virtual System Configuration defines virtualization extensions**
  - Also used as input for Virtual System creation
  - Recorded and Active State
  - Snapshots
ResourceAllocationSettingData

- **RASD -- Key Class for describing aspects of virtualization**
  - Used on create request
  - Used to represent settings specifically related to virtualized resource
  - Resource Type -- The type of resource this allocation setting represents
  - Used in SettingDefinesCapabilities association

- PoolID – ResourcePool allocated from or to be allocated from
- ConsumerVisibility {Unknown, Passed-Through, Virtualized}
- HostResource[] – exposes specific assignment to host or underlying resource
- AllocationUnits
- VirtualQuantity
- Reservation – Amount of resource guaranteed to be available for this allocation
- Limit – Upper bound of resource that will be granted for this allocation
- Weight – relative priority for this allocation
- AutomaticAllocation/AutomaticDeallocation – whether resource is automatically allocated at power on/deallocated at power off
- Parent – Parent of resource, for example controller for port
- Connection – the thing to which this resource is connected – for example named network
- Address – for example MAC address
- MappingBehavior – How this resource maps to underlying resources {Dedicated, Soft Affinity, Hard Affinity, Not supported}
System Configuration

Virtual System Configuration ("Defined")

Virtual System Configuration ("State")

Concrete Component

Processor 1: ResourceAllocationSettingData
- InstanceID = "FE24AC09300E4A0A"
- ResourceType = 3 (Processor)
- AllocationUnits = "Processor"
- VirtualQuantity = 2
- Reservation = 2
- Limit = 2
- Weight = 100
- AutomaticAllocation = True

Memory: ResourceAllocationSettingData

Disk 1: ResourceAllocationSettingData

Port 1: ResourceAllocationSettingData

Concrete Component

Processor 1: LogicalDevice

Memory: LogicalDevice

Disk 1: LogicalDevice

Port 1: LogicalDevice
ResourcePool

- Key Properties
- Primordial – Always exists and aggregates host resources
- Capacity
- Reserved
- ResourceType, subtype, units
Capabilities and Settings

Host System: ComputerSystem

AllocationCapabilities
ResourceType: CPU

ElementCapabilities

HostedResourcePool

:ResourcePool
ResourceType: CPU

AllocationCapabilities
ResourceType: CPU

RASD6: ResourceAllocationSettingData
VirtualQuantity: 4

RASD7: ResourceAllocationSettingData
VirtualQuantity: 1

RASD8: ResourceAllocationSettingData
VirtualQuantity: 1

RASD9: ResourceAllocationSettingData
VirtualQuantity: 1

RASD1: ResourceAllocationSettingData
VirtualQuantity: 2

RASD2: ResourceAllocationSettingData
VirtualQuantity: 1

RASD3: ResourceAllocationSettingData
VirtualQuantity: 1

RASD4: ResourceAllocationSettingData
VirtualQuantity: 1

SettingsDefine Capabilities
ValueRange: Maximums
SettingsDefine Capabilities
ValueRange: Minimums
SettingsDefine Capabilities
ValueRange: Increments
SettingsDefine Capabilities
ValueRole: Default
Total virtualization model - summary

Platform Virtualization Pools and Capabilities
Tells client what and how this platform virtualizes resources. Conforms to System Virtualization Profile

Virtual System Configuration

Resource Allocation SettingData

ConcreteComponent

VirtualSystem Setting Data

ElementSetting Data

Resource Allocation SettingData

Resource Allocation SettingData

SettingsDefine

Virtual System

Basic CS and devices
Conform Virtual System
and SMWG profiles

Resource Pool

Hosted Resource Pool

Host Platform Resources
Explicit Platform Objects conform to SMWG profiles

ComputerSystem (Hypervisor)

Host Devices

System Device

IsMember OfPool

Virtual Device

"Virtual" Device

SettingDefines Capability (Min)

SettingDefines Capability (Max)

AllocationCapability

SettingDefines Capability (Increment)

Resource Allocation SettingData

Resource Allocation SettingData

Resource Allocation SettingData

Copyright (c) 2007 DMTF. All rights reserved.
Virtual System Management Service

- DefineSystem() -- Defines a virtual system. Input that is not completely specified will be filled out with default values
  - Embedded instance of class CIM_VirtualSystemSettingData that is used to define attributes of the virtual system to be defined
  - Array of embedded instance of RASD that describe desired resources
  - Reference to VSSD that refers to virtual system configuration used to complement the configuration of new virtual system if parameters in VSSD and RASD are not provided.

- DestroySystem() – Destroys Virtual System
  - Input is reference to CS instance

- AddResourceSettings() – Adds resource to virtual system configuration – if virtual system is active adds to virtual system
  - Array of embedded instance of class CIM_ResourceAllocationSettingData (RASD) that describes resources to be added to the virtual system

- ModifyResourceSettings()
  - Array of embedded RASD instance for each resource to be modified

- ModifySystemSettings() -- Modifies virtual system settings
  - Input is ref of VSSD to be modified and instance for modified values

- RemoveResourceSettings() -- Removes virtual resource settings from virtual system
  - Input is array of references to RASD representing resources to be removed

- All methods have capability of returning job if long running
Virtual System State

- **“Defined” State**
  - Virtual computer system is defined at the virtualization platform, but not yet instantiated.
  - There is an instance of class CIM_ComputerSystem in this state.
  - A virtual system in the “Defined” state is not enabled to perform tasks.
  - Typically in this state the virtual system does not consume any resources.

- **“Active” State**
  - Virtual computer system is instantiated at the virtualization platform and its resources are performing tasks.

- **“Paused” State – Optional**
  - Host resources remain allocated
  - Virtual system not enabled to perform tasks.

- **“Suspended” State – Optional**
  - Virtual resource persisted
  - Virtual resources represented by device instances but host resources may have been deallocated.
Virtual Storage and Adapter Modeling

- Block and File based Virtual Disks
- Leverage SNIA profiles and packages to provide management application consistency for storage management
Ongoing Work

- Profiles for Additional Virtual Devices
  - Keyboard/Mouse/Video
  - CD-ROM, Floppy – want to model both the (virtual) device, and ability to map to (virtual) media (ISO image, file, real media…)
  - Sound, Video, Serial, USB

- Image Formats (OVF) and Image Management
Summary

- Flexible Model for Virtualization & Partitioning
- Schema Changes part of CIM 2.16
- Profiles available for public comment and feedback per DMTF process
- Expect multiple implementations available in 2007
Questions?

DMTF:  http://www.dmtf.org/

EMAIL:  steve_hand@symantec.com