

November 15-18, 2010



Santa Clara Marriott  
Santa Clara, CA

# **System Virtualization 2.0: A preview**

Michael Johanssen  
IBM

## Disclaimer

- 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) website.



# Acknowledgements

- The author wishes to thank the members of the following DMTF working groups for their continued contributions:
  - SVPC
  - Profile Infrastructure
  - Physical Platform
  - WBEM Infrastructure
  - Platform Subcommittee
  - Schema Subcommittiee
  - Technical Committee



# SVPC

## Version 2 suite of profiles

- Extend the applicability of SVPC profiles
  - OVF
  - Cloud
- New concepts
  - Indications
  - Metrics
  - Standard messages
  - View classes
- Simplification of existing concepts
  - Combine the System Virtualization profile and the Virtual System profile into one profile
  - Combine the Resource Allocation profile and the Allocation Capabilities profile into one profile
  - Restructure the hierarchy of resource pools
- Based on DSP1001 (PUG) 1.1
  - Use of adaptations
  - Use of features
  - Machine readable profiles

# Profile Usage Guide 1.1: Applied concepts

- Adaptation
  - *adapts* a class defined in a schema for a particular purpose, i.e., defines a *use* of the class
  - a named profile element that defines requirements and constraints on a class
  - may be based on other class adaptations
- Feature
  - a named profile element that groups the decisions for the implementation of one or more profile elements into a single decision
  - typically used for modeling the decision to implement a concept that implies implementation requirements for a number of other elements in the profile



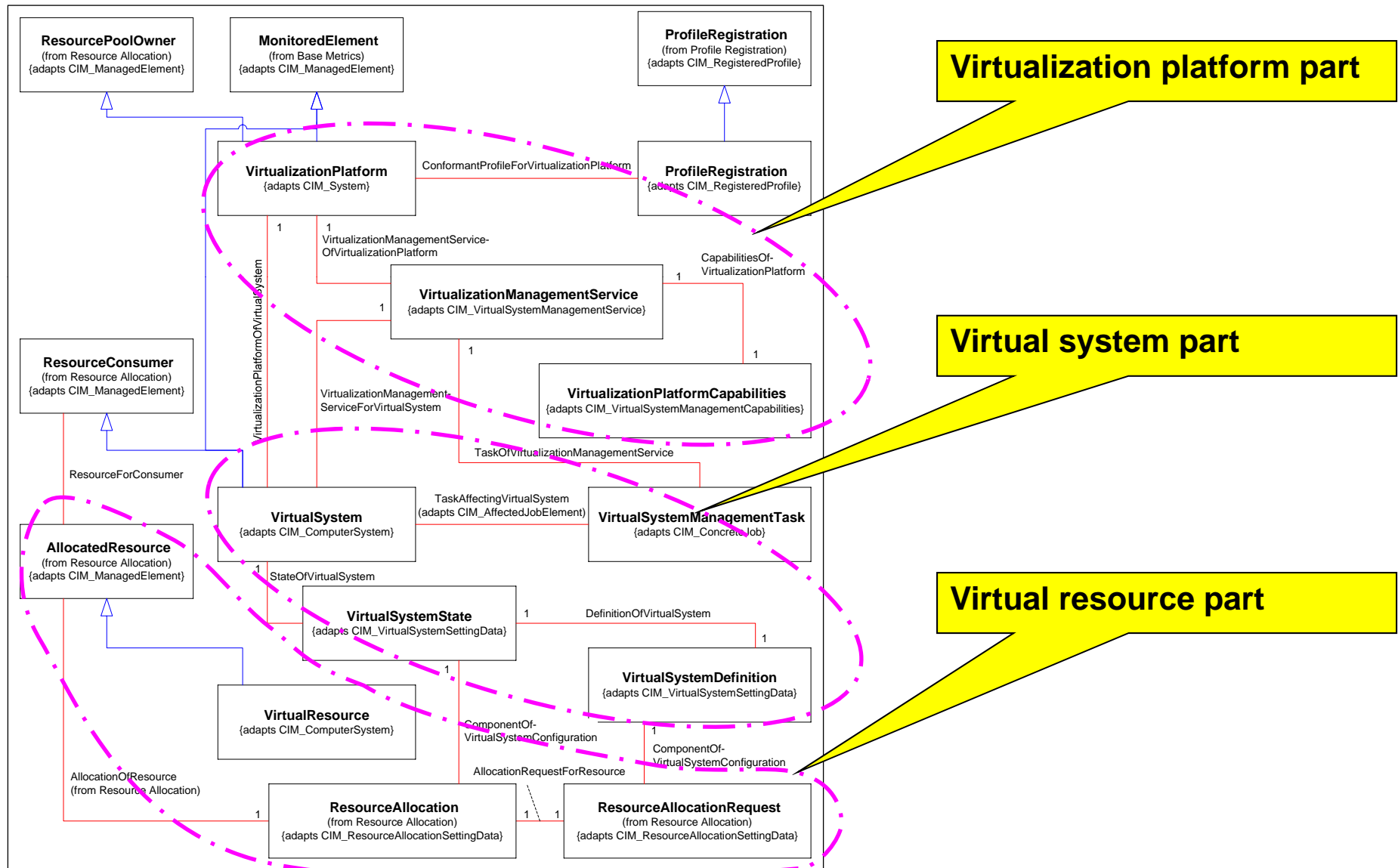
# System virtualization 2.0: Major managed object types

- Virtualization platform
  - virtualizing infrastructure residing at a host system that enables the provision and management of virtual systems
- Virtual system
  - emulated system containing virtual resources
- Resource pool
  - entity representing resources available for and/or in use for the purpose of allocating and exposing allocated resources to consumers
- Virtual resource
  - resource exposed to a consumer as a logical device based on one or more allocated resources
- Host resource
  - device or other kind of resource contained by a host system that may be allocated with either exclusive or shared access via the host system to provide resources to a resource pool or consumer
- Resource allocation
  - the result of assigning, separating, reserving, granting share of or emulating resources for use by a consumer. It describes the amount of resource allocated from a resource pool, including required host resource(s)
- Resource allocation request
  - a request for resource allocation
- Resource mutability
  - a description of the potentially possible modifications of a resource
- Virtual system migration
  - the process of moving an active virtual system from one virtualization platform to another virtualization platform without operational disruption of the virtual system

# System Virtualization profile

- Integration of the Virtual System profile
- Central class adaptation:  
VirtualizationPlatform
  - models virtualization platforms
    - i.e., the virtualizing infrastructure residing at a host system that enables the provision and management of virtual systems
  - adapts the CIM\_System class
  - based on
    - ProfileRegistration::CentralElement
    - ResourceAllocation::ResourcePoolOwner
    - BaseMetric::MonitoredElement

# System Virtualization profile Overview



# Maintain virtual system representation & configuration

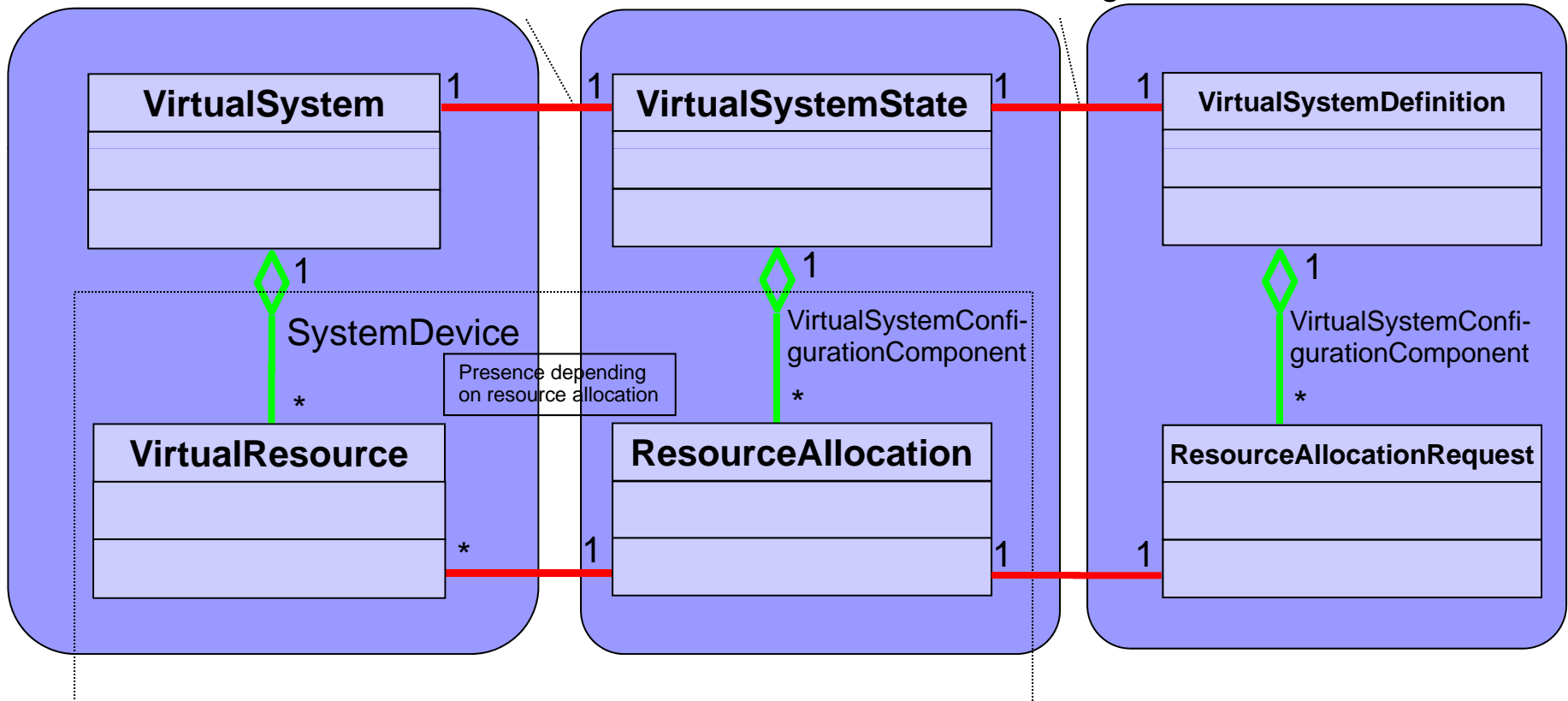
## Virtual System Representation

## Virtual System State Configuration

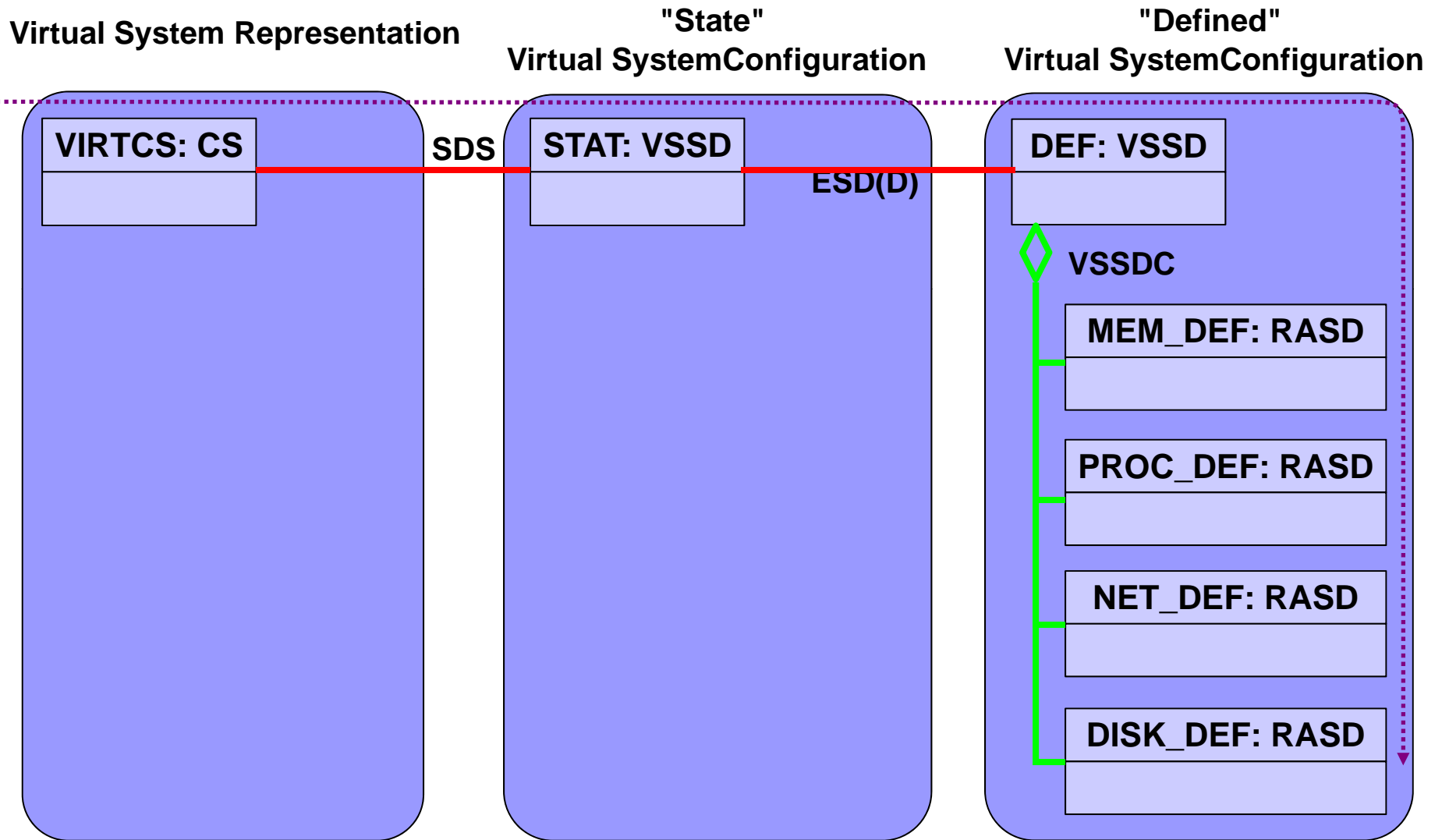
## Virtual System Defined Configuration

SettingsDefineState

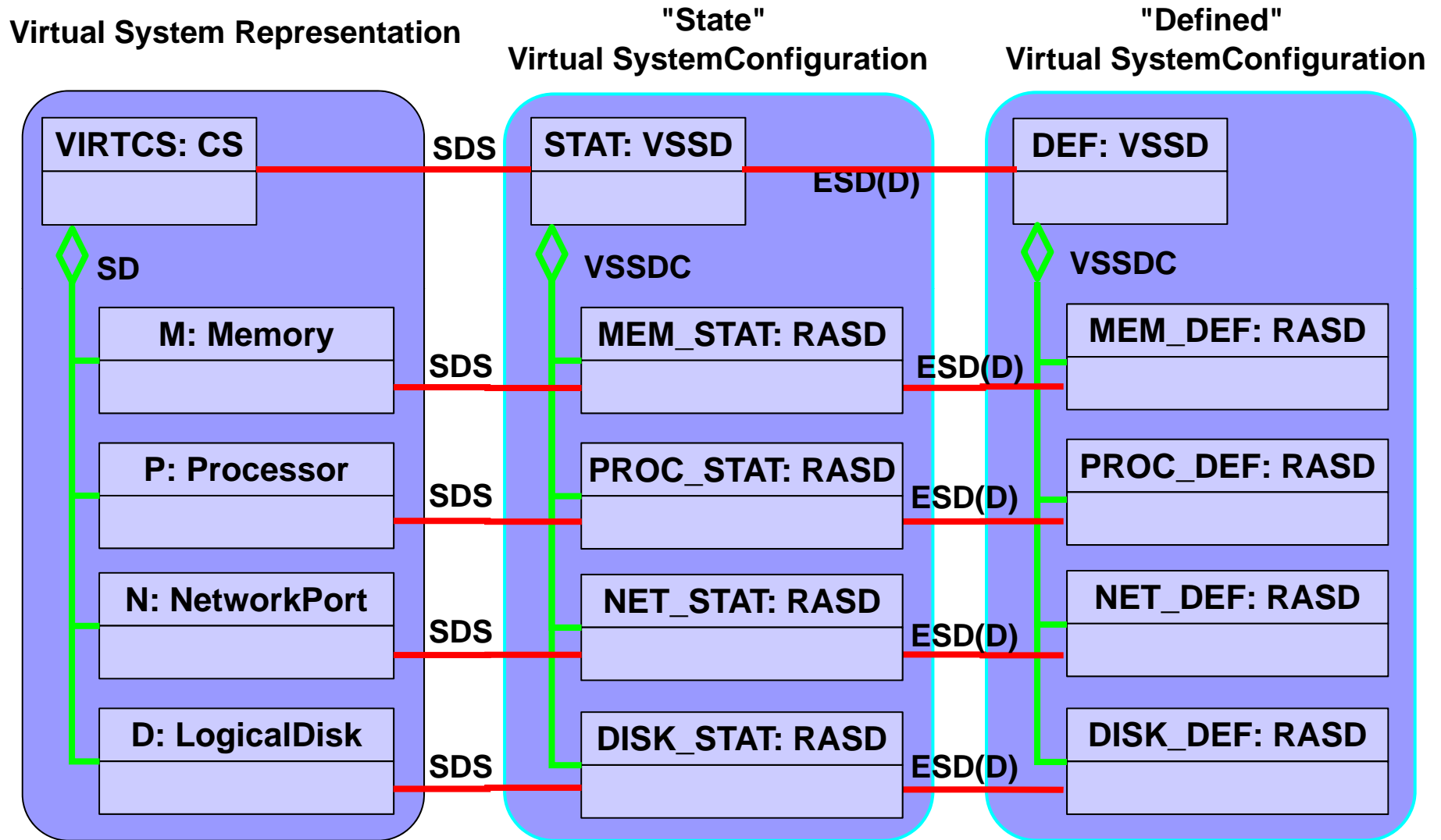
ElementSettingData



# "Defined" virtual system configuration



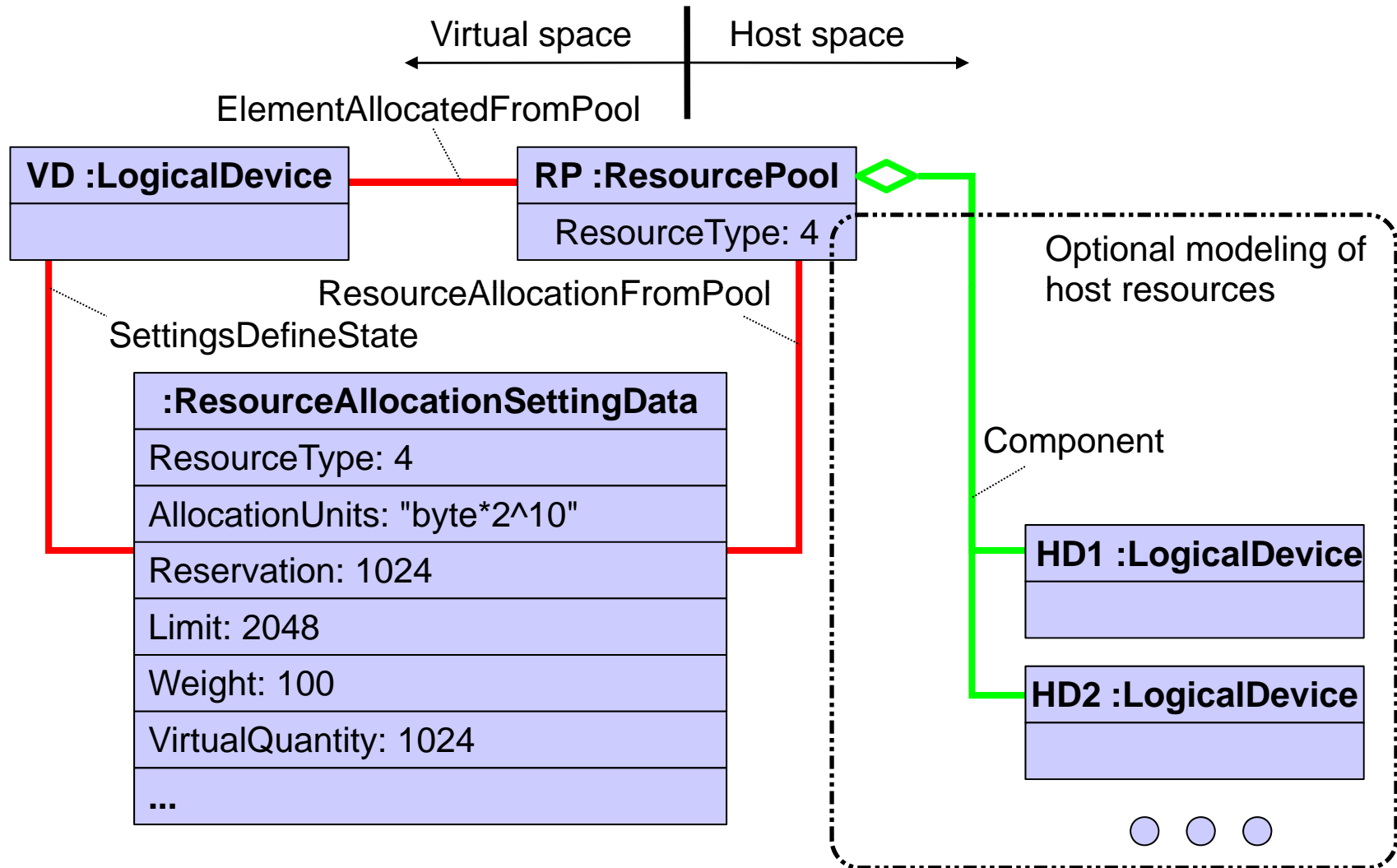
# Virtual system representation and "State" virtual system configuration



# Resource Allocation profile

- Central class adaptation: ResourcePool
  - models resource pools
    - entities representing resources available for and/or in use for the purpose of allocating and exposing allocated resources to consumers
  - adapts the CIM\_ResourcePool class
  - based on
    - ProfileRegistration::CentralElement
    - BaseMetric::MonitoredElement
- Integration of the Allocation Capabilities profile

# Maintain the concept of resource allocation



# Resource pool changes

- Motivation
  - Support for mobility
    - Resource requirements need to be expressed in more abstract terms, rather than in terms of required host resources, i.e.,
      - for file systems, use a global name rather than a local path
      - for networks, use a global network name rather than a local network adapter address
  - Support for multi-tenancy
    - Decoupling of interdependencies of resource requirements for different tenants
    - Addressed through intermediate child pools for allocations to resources belonging to a tenant
      - All resource allocations for a tenant remain within the amount of resource allocated to the tenant specific child resource pool
    - May be used for monitoring resource consumption by tenants

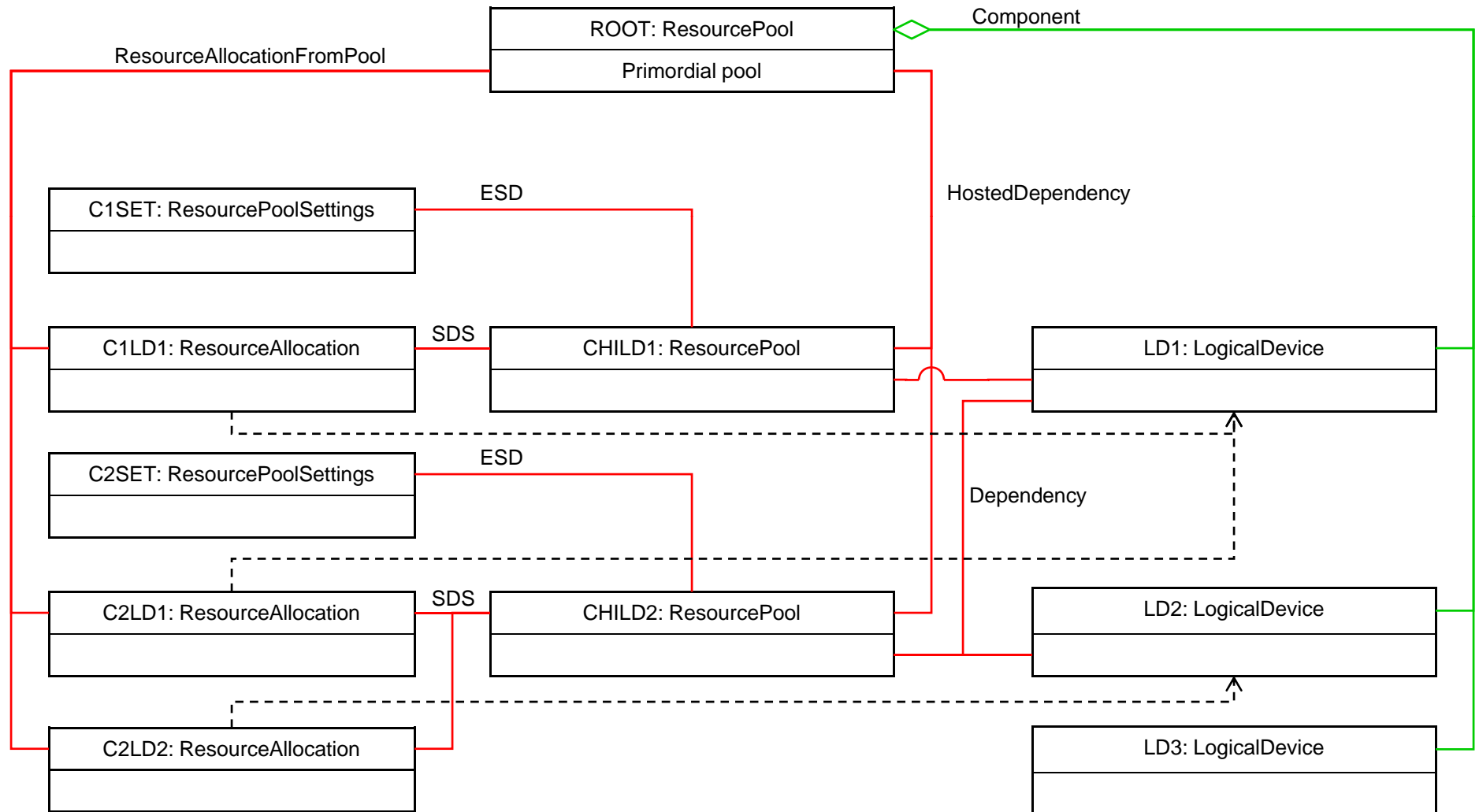
## Resource pool changes (II)

- Maintain concept of primordial resource pools
  - Each resource type will continue to have a single primordial pool into which all host resources of that resource type are initially aggregated
    - typically upon installation of the platform
    - may occur as part of dynamic discovery of resources
  - All compatible host resources are aggregated into the primordial resource pool
    - Applicable to logical devices
    - Applicable to system like resources, such as switches
  - Resources may be allocated from the primordial resource pool to
    - virtual systems
    - child resource pools

## Resource pool changes (III)

- Revised the concept of child resource pools
  - Resources can be allocated to child resource pools just as they can be allocated to other consumers (such as virtual systems)
    - an administrative activity
  - Child pools may depend on host resources contained in the primordial pool
    - Complete resources or fragments
  - Additional configurable capabilities
    - load balancing behavior
    - mapping order for remapping of resources

# Pool resource dependencies



# Indications overview

- Indication
  - a notification about an event
  - the event reported by an indication is defined by an event definition query statement, usually as part of the indication definition
  - indications do not (unlike instances of ordinary CIM classes) exist in a namespace and represent an aspect of a managed object
    - indications are transitional objects that convey the notification about an event from an implementation to a listener
    - however, indications are required to have an *origin namespace* (detailed later)
- Alert indication
  - a notification about root events
  - may be related to a managed objects that may or may not be represented in CIM
- Lifecycle indication
  - a notification about events in a CIM model (always secondary events)
  - always relates to the CIM representation of managed objects

## Virtualization management V2 general decisions wrt. indications

- Model alert indications
  - based on the *AlertIndication* adaptation of version 1.2 of the *Indications* profile
  - extend the *SVPC* message registry with respective messages



# Indications profile 1.2

## Applied concepts

- Use of class adaptations
- Referencing profiles define indication adaptations based on basic indication adaptations defined in the Indications profile
- Particularly easy for alert indications
  - Use of the AlertIndication adaptation
  - based on messages defined in a message registry
- No need to define filters & filter collections in the referencing profile
- All indication delivery related tasks are taken care of by the Indication profile implementation within a WBEM server

# SVP Indications

- VM Lifecycle:
  - VM create / delete
  - VM activate / deactivate
  - VM suspend / resume
- VM setting changes (name, etc.)
- VM snapshot
  - capture / apply
- VM migration
  - Using indications of the Job Control profile
    - Create / delete / change / % complete of migration job (lifecycle indications)
    - Alternatively, define respective alert indications (for consistency)
- Resource management
  - Add / remove / modify resource (allocation)
  - Possible for the active and for the defined configuration

# RAP Indications

- Resource pool indications
  - Utilization threshold – per pool
  - Percent capacity reserved
  - Pool creation / deletion
  - Host Resource add/remove from pool
  - Sub-pool resource allocation changes
  - probably more pool management related indications ...

- General decisions:
  - Model CPU and Memory metrics
    - Virtualization platform
    - Virtual system
    - Resource pool
  - based on the MonitoredElement adaptation of version 1.1 of the Base Metrics profile
  - Work on a SVPC metrics registry is already in progress

# Metrics

## Related standards

---

- Metric Registry XML schema (DSP8020)
- System Virtualization Performance Metrics Registry (DSP8048)
- Base Metrics profile (DSP1053)
- Capacity Metrics profile (DSP1073)
- CIM schema
- some XSLs to transform metric registries into different formats



# Metrics

## Basic attributes

- **METRICID:**
  - programmatically useable metric ID, globally unique, contains owning organization and reference name
  - Example: "DMTF:DSP8048:SVPNumberOfPhysicalCPUsUtilized"
- **LOCALID:**
  - metric ID unique within the metric registry, such as "SVPNumberOfPhysicalCPUsUtilized"
  - identifier, can be used as a property name
- **NAME:**
  - metric name for human consumption
- **METRIC\_DESCRIPTION:**
  - a comprehensive description of the metric
- **DATATYPE:**
  - metric data type, like uint64 or real32 or boolean
- **PROGRAMMATIC\_UNITS:**
  - programmatically useable unit of the metric
- **CHANGE\_TYPE:**
  - Counter or Gauge
- **TIMESCOPE:**
  - Point, Interval or StartupInterval
- **GATHERING\_TYPE:**
  - OnChange, Periodic or OnRequest

# Virtualization platform metrics (selection)

- **CPU**

- **SVPNumberOfPhysicalCPUsUtilized:** Equivalency of physical CPU cores utilized on virtualization platform level
- **SVPAverageProcessorFrequency:** Average frequency processor cores attached to the virtualization platform are running at
- **SVPAverageProcessorFrequencyPercentage:** Max. frequency divided by actual frequency processor cores are running at
- **SVPTotalCPUTime:** CPU time in  $\mu$ s within the measurement interval, aggregating CPU times from all CPU cores used by the associated virtualization platform
- **SVPManagementTime:** Management overhead time caused by the virtualization platform

- **Memory**

- **SVPMemoryAllocatedToVirtualSystems:** Sum of real memory currently used by virtual systems not including unallocated physical memory or memory used by the virtualization platform
- **SVPMemory:** Memory used by the virtualization platform for itself
- **SVPActiveMemory:** Memory used by the virtualization platform which has been referenced recently
- **SVPPageInRate:** Page-in rate, virtualization platform level
- **SVPMemoryAvailableForVirtualServerCreation:** Virtualization platform physical memory available for virtual server creation. This is an estimate on what amount of memory is free plus the memory which can be freed by the virtualization platform without causing performance problems because of a high page rate.
- **SVPFreePhysicalMemory:** Virtualization platform real memory currently not used.

# Virtual system metrics

- **CPU**
  - **VSCPUReserved**: CPU reserved by the associated virtual server; same unit as the related processor resource pool
  - **VSCPUUtilized**: CPU used by the associated virtual server; same unit as the related processor resource pool
  - **VSNumberOfPhysicalCPUsUtilized**: equivalency of physical CPU cores utilized by the associated virtual server
  - **VSTotalCPUTime**: CPU time in  $\mu$ s within the measurement interval, aggregating CPU times from all CPU cores used by the associated virtual server
- **Memory**
  - **VSRealMemoryMapped**: Real memory mapped to virtual system . This is also known as resident set size.
  - **VSWorkingSetSize**: Estimated minimal amount of real memory required to run this virtual system without any significant paging activity.
  - **VSActiveMemory**: Memory used by the associated virtual server which has been referenced recently
  - **VSSharedMemory**: Amount of shareable memory used by the associated virtual server.
  - **VSTargetMemory**: Target memory size

# Resource pool metrics

- **CPU**
  - **CPURPReserved**: CPU capacity from the associated processor resource pool that is reserved by some virtual servers, similar to the CIM\_ResourcePool.Reserved property for processor resource pools.
  - **CPURPCapacity**: Size of the associated processor resource pool, similar to CIM\_ResourcePool.Capacity property for processor resource pools.
    - This metric might have values like “320 000 MHz”, or “6400 entitlements”, where MHz corresponds to programmatic unit “hertz \* 10<sup>6</sup>” and entitlement corresponds to programmatic unit “count \* 0.01” and count is meant to count processor cores.
  - **CPURPUtilized**: Amount of CPU resource from the associated processor resource pool that is actually utilized.
- **Memory**
  - **MemoryRPAAllocated**: Memory from the associated resource pool which is currently allocated
  - **MemoryRPCapacity**: Memory resource pool capacity including space on paging device
  - **MemoryRPRealMemorySize**: Memory resource pool real memory size

# System Virtualization profile

## MRP

- e mrp:Elements
  - + e mrp:ClassAdaptation name=VirtualizationPlatform
  - e mrp:ClassAdaptation name=VirtualSystem
    - e mrp:BaseAdaptation name=ResourceConsumer
    - e mrp:BaseAdaptation name=MonitoredElement
    - e mrp:Requirement level=Mandatory
    - + e mrp:Description
    - + e mrp:MetricReference localId=VSCPUReserved
    - e mrp:MetricReference localId=VSCPUUtilized
      - e mrp:Description
      - e mrp:Requirement level=Optional
    - e mrp:MetricReference localId=VSNumberOfPhysicalCPUsUtilized
      - e mrp:Description
      - e mrp:Requirement level=Optional
    - e mrp:MetricReference localId=VSTotalCPUTime
      - e mrp:Description
      - e mrp:Requirement level=Optional
    - + e mrp:Property name=EnabledState
    - + e mrp:Method name=RequestStateChange
    - + e mrp:Operation name=Associators
    - + e mrp:Operation name=AssociatorNames
    - + e mrp:Operation name=References
    - + e mrp:Operation name=ReferenceNames
    - + e mrp:Operation name=ModifyInstance
  - + e mrp:ClassAdaptation name=VirtualizationPlatformOfVirtualSystem
  - + e mrp:ClassAdaptation name=VirtualSystemState

**All adaptation related requirements in one place**

**Base adaptation: Monitored Element  
Base adaptations have individual requirement levels**

**Metric requirements:  
Apply in context of the  
MonitoredElement base adaptation**

**Property requirement**

**Method requirement**

**Operation requirement**

**Table: Element requirements for VirtualSystem adaptation**

Element	Requirement	Description
<b>Properties</b>		
EnabledState	Mandatory	See CIM schema definition.
<b>Methods</b>		
RequestStateChange( )	Optional	See CIM schema definition.
<b>Operations</b>		
Associators( )	Mandatory	See <a href="#">DSP0200</a> .
AssociatorNames( )	Mandatory	See <a href="#">DSP0200</a> .
References( )	Mandatory	See <a href="#">DSP0200</a> .
ReferenceNames( )	Mandatory	See <a href="#">DSP0200</a> .
ModifyInstance( )	Optional	See <a href="#">DSP0200</a> .

EXPERIMENTAL: The following table lists the metric definitions for this adaptation. For each metric definition in this table, one instance of this adaptation shall exist that conforms to that metric definition.

**EXPERIMENTAL: Table: Metric definitions**

Metric ID (LOCALID)	Metric display name	Defining metric registry
VSCPURreserved	See defining message registry	<a href="#">DMTF DSP8048</a>
VSCPUUtilized	See defining message registry	<a href="#">DMTF DSP8048</a>
VSNumberOfPhysicalCPUsUtilized	See defining message registry	<a href="#">DMTF DSP8048</a>
VSTotalCPUTime	See defining message registry	<a href="#">DMTF DSP8048</a>

## Status

- Work-group internal MRP version of the System Virtualization profile under discussion
  - WIP releases expected in 2011
- Extensions to the System Virtualization message registry under discussion
  - WIP release expected in 2011
- First version of the System Virtualization Performance metrics registry in work
  - WIP release expected in 2010