hp WBEM solutions

hp technical data sheet

CPU Instance Provider

Provider overview

Description

The CPU Instance Provider is a Web-Based Enterprise Management (WBEM) instance provider. It provides information about processor inventory on supported HP Integrity Servers, running HP OpenVMS. This provider is compliant with the Common Information Model (CIM) 2.7.2 Schema, proposed by the Distributed Management Task Force (DMTF). The provider requires HP WBEM Services for OpenVMS installed on the system.

You can query for information about processor inventory on a managed system using a management application that is compliant with the CIM 2.7.2 schema; for example, HP Systems Insight Manager. The Common Information Model (CIM) is an extensible, object-oriented data model that contains information about different parts of an enterprise.

The CPU Instance Provider implements the processor-related CIM classes, proposed in the DMTF CIM 2.7.2 revision. In addition to the properties that belong to the standard CIM classes, the CPU Instance Provider serves information that is specific to HP Servers, by implementing HP-specific CIM classes, derived from the standard DMTF classes.

The following Managed Object Format (MOF) classes are handled by the CPU Instance Provider:

- HP_Processor and HPVMS_Processor

  HP_Processor (subclass of CIM_Processor) and HPVMS_Processor (subclass of HP_Processor) represent “logical” information about the processors, including status, the family of the processors, clock-speeds, etc.

- HP_ProcessorChip

  HP_ProcessorChip (subclass of CIM_Chip) represents “physical” information about the processor, such as the Processor Firmware Revision, Architecture Revision, etc.

- HP_ProcessorLocation and HPVMS_ProcessorLocation

  HP_ProcessorLocation (subclass of CIM_Location) and HPVMS_ProcessorLocation (subclass of HP_ProcessorLocation) captures information about the physical location of the processor chip and the core of the processor, as seen from the physical perspective. This includes identification of which cell the processor-chip resides in, the slot numbers, cabinet numbers, etc.

- HP_ProcessorCollection

  HP_ProcessorCollection (subclass of HP_GroupSystemSpecificCollection) represents the processor-subsystem on the computer-system. The group-operational-status property of this class represents the health of the current system’s processor-subsystem.

In addition, the CPU Instance Provider also implements association classes to associate the instances of the different CIM classes mentioned above. These include:

- HP_RealizesProcessor (subclass of CIM_Realizes): This class identifies which logical Processor (HP_Processor) instance is associated to which Physical Processor (HP_ProcessorChip) instance.

- HP_ProcessorChipInLocation (subclass of CIM_PhysicalElementLocation): This class indicates the physical location (HP_ProcessorLocation) corresponding to a specific processor chip (HP_ProcessorChip).

- HP_ProcessorGroupHostedCollection (subclass of HP_GroupHostedCollection): This class represents the association between the processor subsystem (HP_ProcessorCollection) and the computer-system to which the subsystem belongs.

- HP_MemberOfProcessorCollection (subclass of CIM_MemberOfCollection): This class retrieves the "member-of" relationship between the processor-class-instances and the processor-collection (HP_ProcessorCollection). Instances of this class associate all CIMinstances that contribute to group-operational-status of the HP_ProcessorCollection, with the instance of the HP_ProcessorCollection itself.

The MOF classes mentioned above (i.e. all MOF classes prefixed with "HP_") are HP-specific extensions to the CIM Schema, and are registered in the "root/cimv2" namespace.

The following example illustrates the relationship between the MOF classes mentioned above. On an HP Itanium based server containing a single cell, with 4 processor-slots and one dual-core Itanium Processor (With Multi threading disabled), the CIM Instances returned by the CPU Instance Provider are as follows:

- 2 instances of HPVMS_Processor (one for each of the processors visible to the running OpenVMS OS kernel)
- 1 instance of HP_ProcessorChip (representing the single processor-chip FRU, i.e. the Itanium Processor-Chip)
- 1 instance of HPVMS_ProcessorLocation (representing the slot occupied by the Itanium processor-chip above)
- 2 instances of HPVMS_RealizesProcessor (each one associating one of the 2 HPVMS_Processor instances with the single HP_ProcessorChip instance)
- 1 instance of HPVMS_ProcessorChipInLocation (associating the single HPVMS_ProcessorChip instance with the HPVMS_ProcessorLocation instance (location/slot in which it rests)).
- 1 instance of HP_ProcessorCollection (representing the single instance of the processorsubsystem).
- 2 instances of HP_MemberOfProcessorCollection (one for each instance of the CIM-class that contributes to the group-operational-status of the processor-subsystem).
- 1 instance of HP_ProcessorGroupHostedCollection (associating the single processorsubsystem to the computer-system-instance).

For all the MOF classes mentioned above, the CPU Instance Provider supports the following standard CIM Operations:

- enumerateInstanceNames()
- enumerateInstances()
- getInstance()

The following CIM operations are not supported by the CPU Instance Provider:

- createInstance()
- deleteInstance()
- modifyInstance()

The CPU Instance Provider is not a CIM Method Provider, and does not support extrinsic method invocation on instances on any of the MOF classes mentioned above. The invocation of any of these methods will result in a CIM_ERR_NOT_SUPPORTED exception.

Requirements

The Provider requires HP WBEM Services for OpenVMS.

Release history

This provider will be available via OpenVMS Version 8.3-1H1 release.

Supported managed resources

This provider provides "logical" information about system CPU's, "physical" attributes of the Processor-Chip, and details of the physical-location of the processor.

Note that the CPU Instance Provider provides only the information about the above resources. It does not provide any management, diagnostic or configuration capabilities for the above resources.

Setting up this provider

Installing this provider

The installation of HP WBEM Providers will set up this provider. Ensure HP WBEM Services is already installed.

On installation, executable binaries, configuration files and MOF definition and registration files will be available in their respective directory, as follows:

- The CIM MOF files, containing the definitions of the HP-specific MOF classes, (namely HPProcessor.mof & HP_ProcessorStatus.mof) will be available in SYS$COMMON:[WBEMPROVIDERS.MOF]. This directory will also include the provider registration file, namely CPUINSTANCEPROVIDERR.MOF. Note: All the HP-specific MOF classes will be registered under the "root/cimv2" namespace.

- The SYS$SPECIFIC:[WBEMPROVIDERS] directory will contain the (XML) configuration files of the WBEM Providers Product.

- The WBEM Services SYS$SPECIFIC:[WBEM_Services]CIMSERVER_STARTUP.LOG log file will contain logs generated during the execution of the CPU Instance Provider.

Configuring this provider

This provider does not accept specific configuration adjustments (beyond standard WBEM support).

Using this provider

Schema supported by this provider

The "Description" section explains in brief the different MOF classes supported by the CPU Instance Provider. The following tables list all the supported properties corresponding to these MOF classes, along with the properties inherited from the standard CIM MOF classes, as per CIM 2.7.2 schema specifications.

Note: All key properties corresponding to the CIM classes are supported by the CPU Instance Provider. The few non-key properties not supported (currently) by the CPU Instance Provider are not listed below.

Note:

1. All key properties corresponding to the CIM classes are supported by the CPU Instance Provider.
2. The non-key properties that are not supported by the CPU Instance Provider are not listed below.

Table 1: HP_Processor and HPVMS_Processor Properties (Logical Processor Information):
Table 1 describes the properties of the HP_Processor and HPVMS_Processor CIM classes. It has three columns. The first is the property name (including type and units), the second is the property inheritance (indicating which class or superclass defines the property), and the third is the property's value and data source. Each row describes a property.

Property name	***Property inheritance***	***Property value (and data source)***
string Caption	Inherited from CIM_ManagedElement	This value is always returned as "Processor (SPU)".
string Description	Inherited from CIM_ManagedElement	For HP Integrity Servers, this string is set to "This is an Intel ® Itanium ® 2 Processor (Model: Intel (R) Itanium 2 Processor (Family F Model M Stepping S)), with the following details:", where: F: identifies the family of the processor (E.g. 31 for Itanium 2) M: indicates the model of the processor S: indicates the stepping string of the processor Followed by this is the location-details identified on the system hardware for the processor. Location details would include the following, which ever is applicable: 1. Cabinet Number 2. Card Cage Number 3. Backplane Number 4. Cell Slot number 5. Slot number 6. SPU Number (as seen by the OS instance)
string ElementName	Inherited from CIM_ManagedElement	For HP Integrity Servers, this string is set to "Intel ® Itanium ® 2 Processor".

String Name	Inherited from CIM_ManagedSystemElement	For HP Integrity Servers, this string is set to “Intel ® Itanium ® 2 Processor”.
uint16 OperationalStatus []	Inherited from CIM_ManagedSystemElement	The Value-Map associated with this property (as per the CIM 2.7.2 schema specification) is as follows: ValueMap {"0", "1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15", "16", "17"}, Values {"Unknown", "Other", "OK", "Degraded", "Stressed", "Predictive Failure", "Error", "Non-Recoverable Error", "Starting", "Stopping", "Stopped", "In Service", "No Contact", "Lost Communication", "Aborted", "Dormant", "Supporting Entity in Error", "Completed"} The OperationalStatus array contains multiple values, to indicate the different aspects of the CPU’s status. The values returned in different scenarios are as follows: 1. If the processor is configured and active, the first element is 2 (for “OK”). 2. If the processor is Configured and Inactive, then the first element is 2 (for “OK”) and the second element is 15(for “Dormant”) 3. If the processor is Marked for configuration, then the first element is set to 2 (for “OK”) and the second element is set to 8 (for “Starting”) 4. If the processor is Marked for Deconfiguration and Active, then the first element is set to 2 (for “OK”) and the second element is set to 9 (for “Stopping”). 5. If the processor is Marked for Deconfiguration and Inactive, then the first element is set to 2(for “OK”), the second element is set to 15 (for “Dormant”) and the third element is set to 9 (for “Stopping”). 6. If either the Deconfiguration status, or the Active status of the processor is unknown, or not retrievable, this array will contain a single element 0, indicating an “Unknown” status
string StatusDescriptions[]	Inherited from CIM_ManagedSystemElement	This contains string descriptions for the status values returned in the Operational Status array described above. Each value in the StatusDescriptions array corresponds to the (localized) verbose status description for the value at the same index in the OperationalStatus array. The Strings describing the different values in the OperationalStatus Array are as follows: 1. The Description string corresponding to a value of “2” in the OperationalStatus array is: “Processor is OK”. 2. The Description string corresponding to a value of “15” in the OperationalStatus array is: “Processor is idle. There are no processes scheduled on this processor.” 3. The Description string corresponding to a value of “8” in the OperationalStatus array is: “Processor is currently deconfigured, and will be configured in the next reboot.” 4. The Description string corresponding to a value of “9” in the OperationalStatus array is:

		“Processor is currently configured, and will be deconfigured in the next reboot.” 5. The Description string corresponding to a value of “6” in the OperationalStatus array is: “Processor is in Error”. 6. The Description string corresponding to a value of “0” in the OperationalStatus array is: “Processor is in an UNKNOWN state”. For example, consider that the processor is Marked for Deconfiguration, and Active. The OperationalStatus array will be set such that: 1. OperationalStatus[0] = 2 2. OperationalStatus[1] = 9 Then, the StatusDescriptions array will correspondingly be set such that: 1. StatusDescriptions[0] = “Processor is OK” 2. StatusDescriptions[1] = “Processor is currently configured, and will be deconfigured in the next reboot.”
string SystemCreationClassName [Key]	Inherited from CIM_LogicalDevice	Fixed string “CIM_ComputerSystem”
string SystemName [Key]	Inherited from CIM_LogicalDevice	The host name of the server.
string CreationClassName [Key]	Inherited from CIM_LogicalDevice	This is set to the name of the instantiated sub-class, i.e. “HPVMS_Processor”.
string DeviceID [Key]	Inherited from CIM_LogicalDevice	This is set to the LID of the processor.
Uint16 Family	Inherited from CIM_Processor	The Value-Map for this property looks as follows: ValueMap {"1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15", "16", "17", "18", "19", "24", "25", "26", "27", "28", "29", "30", "31", "32", "33", "34", "35", "36", "37", "38", "39", "48", "49", "50", "51", "52", "53", "54", "55", "64", "65", "66", "67", "68", "69", "80", "81", "82", "83", "84", "85", "86", "87", "88", "96", "97", "98", "99", "100", "101", "112", "120", "121", "128", "130", "144", "145", "146", "147", "148", "149", "150", "160", "176", "177", "178", "179", "180", "181", "182", "183", "184", "190", "200", "201", "202", "250", "251", "260", "261", "280", "281", "300", "301", "302", "320", "350", "500"} Values {"Other", "Unknown", "8086", "80286", "80386", "80486","8087", "80287", "80387", "80487", // 11 "Pentium(R) brand", "Pentium(R) Pro", "Pentium(R) II", "Pentium(R) processor with MMX(TM) technology", "Celeron(TM)", "Pentium(R) II Xeon(TM)", "Pentium(R) III", "M1 Family", "M2 Family", //24 "K5 Family", "K6 Family", "K6-2", "K6-3", "AMD Athlon(TM) Processor Family", "AMD(R) Duron(TM) Processor", "AMD29000 Family", //31 "K6-2+", "Power PC Family", "Power PC 601", "Power PC 603", "Power PC 603+", "Power PC 604", "Power

		PC 620", "Power PC X704", "Power PC 750", // 48 "Alpha Family", "Alpha 21064", "Alpha 21066", "Alpha 21164", "Alpha 21164PC", "Alpha 21164a", "Alpha 21264", "Alpha 21364", // 64 "MIPS Family", "MIPS R4000", "MIPS R4200", "MIPS R4400", "MIPS R4600", "MIPS R10000", // 80 "SPARC Family", "SuperSPARC","microSPARC II", "microSPARC IIep", "UltraSPARC", "UltraSPARC II", "UltraSPARC IIi", "UltraSPARC III", "UltraSPARC IIIi", // 96 "68040", "68xxx Family", "68000", "68010", "68020", "68030", // 112 "Hobbit Family", "Crusoe(TM) TM5000 Family", "Crusoe(TM) TM3000 Family", "Weitek", "Itanium(TM) Processor", // 144 "PA-RISC Family", "PA-RISC 8500", "PA-RISC 8000", "PA-RISC 7300LC", "PA-RISC 7200", "PA-RISC 7100LC", "PA-RISC 7100", // 160 "V30 Family", "Pentium(R) III Xeon(TM)", "Pentium(R) III Processor with Intel(R) SpeedStep(TM) ""Technology", "Pentium(R) 4", "Intel(R) Xeon(TM)", // 180 "AS400 Family", "Intel(R) Xeon(TM) processor MP", "AMD AthlonXP(TM) Family", "AMD AthlonMP(TM) Family", "Intel(R) Itanium(R) 2", // 190 "K7", // 200 "IBM390 Family", "G4", "G5", // 250 "i860", "i960", "SH-3", "SH-4", "ARM", "StrongARM", // 300 "6x86", "MediaGX", "MII", "WinChip", "DSP", "Video Processor"}, For HP Integrity Servers, this property is set to “184”, to indicate “Intel ® Itanium ® 2”.
Uint32 CurrentClockSpeed	Inherited from CIM_Processor	The clock speed of the processor, in MHz.
Uint16 DataWidth	Inherited from CIM_Processor	Width of the data-bus of the processor.
Uint16 LoadPercentage	Inherited from CIM_Processor	Loading of the processor, averaged over one minute, in percentage.
Uint16 SpuId	Inherited from HPVMS_Processor	This is an ID of the processor as seen by the OS.

Table 2: HP_ProcessorChip properties

Table 2 describes the properties of the HP_ProcessorChip class. It has three columns. The first is the property name (including type and units), the second is the property inheritance (indicating which class or superclass defines the property), and the third is the property's value and data source. Each row describes a property.

Property name	***Property inheritance***	***Property value (and data source)***
string Caption	Inherited from CIM_ManagedElement.	This value is always returned as "Processor-Module".
string Description	Inherited from CIM_ManagedElement.	For HP Integrity Servers, this string is set to "This is an Intel ® Itanium ® 2 Processor-module (Model: Intel (R) Itanium 2 Processor (Family F Model M Stepping S)), with the following details:", where: F: identifies the family of the processor (E.g. 31 for Itanium 2) M: indicates the model of the processor-chip S: indicates the stepping string of the processor-chip followed by the location-details for the processor. Location details include (where available): 1. Cabinet Number 2. Card Cage Number 3. Backplane Number 4. Cell Slot number 5. Slot number
string ElementName	Inherited from CIM_ManagedElement	For HP Integrity Servers, this string is set to "Intel ® Itanium ® 2 Processor-Module".
String Name	Inherited from CIM_ManagedSystemElement	For HP Integrity Servers, this string is set to "Intel ® Itanium ® 2 Processor-Module".
Uint16 OperationalStatus	Inherited from CIM_ManagedSystemElement.	The Value-Map associated with this property (as per CIM 2.7.2 Schema specifications) is as follows: ValueMap {"0", "1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15", "16", "17"}, Values {"Unknown", "Other", "OK", "Degraded", "Stressed", "Predictive Failure", "Error", "Non-Recoverable Error", "Starting", "Stopping", "Stopped", "In Service", "No Contact", "Lost Communication", "Aborted", "Dormant", "Supporting Entity in Error", "Completed"} If the processor is Configured, the first element is 2 (for "OK"). If the processor is Marked for configuration, then the first element is set to 2 (for "OK") and the second element is set to 8 (for "Starting"). If the processor is Marked for Deconfiguration and Active, then the first element is set to 2 (for "OK") and the second element is set to 9 (for "Stopping"). If the Deconfiguration status of the processor is unknown, or not retrievable, this array will contain a single element 0, indicating an "Unknown" status.

string StatusDescriptions[]	Inherited from CIM_ManagedSystemElement	This contains string descriptions for the status values returned in the Operational Status array described above. Each value in the StatusDescriptions array corresponds to the (localized) verbose status description for the value at the same index in the OperationalStatus array. The Strings describing the different values in the OperationalStatus Array are as follows: The Description string corresponding to a value of "2" in the OperationalStatus array is: "Processor-Module is OK". The Description string corresponding to a value of "8" in the OperationalStatus array is: "The Processor-Module is currently deconfigured, and will be configured in the next reboot." The Description string corresponding to a value of "9" in the OperationalStatus array is: "The Processor-Module is currently configured, and will be deconfigured in the next reboot." The Description string corresponding to a value of "6" in the OperationalStatus array is: "Processor-Module has Error". The Description string corresponding to a value of "0" in the OperationalStatus array is: "Processor-Module is in an UNKNOWN state". For example, consider that the processor-chip is Marked for Deconfiguration. The OperationalStatus array will be set such that: OperationalStatus[0] = 2 OperationalStatus[1] = 9 Then, the StatusDescriptions array will correspondingly be set such that: StatusDescriptions[0] = "Processor-Module is OK" StatusDescriptions[1] = "The Processor-Module is currently configured, and will be deconfigured in the next reboot."
String Tag [Key]	Inherited from CIM_PhysicalElement	This string will be set to a unique value, indicating the Physical Location of the processor-chip.
String CreationClassName [Key]	Inherited from CIM_PhysicalElement	The name of the subclass being instantiated, i.e. "HP_ProcessorChip".
String Model	Inherited from CIM_PhysicalElement	This contains the model string, identifying the model of the processor-chip. For HP Integrity Servers, this contains a string of the format: "Intel (R) Itanium (R) 2 Processor (Family F Model M Stepping S)" where F: identifies the family of the processor (E.g. 31 for Itanium 2) M: indicates the model of the processor-chip S: indicates the stepping string of the processor-chip. E.g. For an rx7620 Server, the string could be: "Intel Itanium 2 (R) Processor (Family 31 Model 1 Stepping B1)"

String SerialNumber	Inherited from CIM_PhysicalElement	This contains the serial-number of the processor-chip.
Uint16 ArchitectureRevision	Inherited from HP_ProcessorChip	The Value-Map of this property is as follows: ValueMap {"0", "1", "2", "3", "4", "5"}, Values {"Unknown", "Other", "PARISC 1.0", "PARISC 1.1", "PARISC 2.0", "Itanium Architecture"}] The Value of this property is 5 for Intel ® Itanium ® 2
String FirmwareRevision	Inherited from HP_ProcessorChip	Firmware Recipe Number, identifying the Processor's Firmware Revision.

table 3: HP_ProcessorLocation and HPVMS_ProcessorLocation properties

Table 3 describes the properties of the HP_ProcessorLocation and HPVMS_ProcessorLocation classes. It has three columns. The first is the property name (including type and units), the second is the property inheritance (indicating which class or superclass defines the property), and the third is the property's value and data source. Each row describes a property.

Property name	***Property inheritance***	***Property value (and data source)***
string Caption	Inherited from CIM_ManagedElement.	This value is always returned as "Processor Slot".
string Description	Inherited from CIM_ManagedElement.	For HP Integrity Servers, this string is set to "This is a processor-slot location, with the following details" followed by the location-details for the processor. Location details include (where available): • Cabinet Number • Card Cage Number • Backplane Number • Cell Slot number • Slot number (CPU Slot)
string ElementName	Inherited from CIM_ManagedElement	For HP Integrity Servers, this string is set to "Processor Slot".

String Name [Key]	Inherited from CIM_Location	The location is returned as a string of the form which ever is identified on the system hardware
		"CabinetNumber = <Cabinet#>:
		CardCageNumber = <CardCage#> :
		BackPlaneNumber = <BackPlane#> :
		CellSlotNumber = <CellSlot#> :
		SlotNumber = <Slot#> : "
		Note that the SPU# is the number assigned to the processor by that instance of the OS"

String PhysicalPosition [Key]	Inherited from CIM_Location	A string indicating (uniquely) the position of the processor.
String CellNumber	Inherited from HP_ProcessorLocation	The number of the Cell to which the processor belongs.
String SlotNumber	Inherirted from HP_ProcessorLocation	The number of the slot in which the processor rests.

table 4: HP_RealizesProcessor properties

Table 4 describes the properties of the HP_RealizesProcessor association class (associating HP_ProcessorChip and HP_Processor). It has three columns. The first is the property name (including type and units), the second is the property inheritance (indicating which class or superclass defines the property), and the third is the property's value and data source. Each row describes a property.

Property name	***Property inheritance***	***Property value (and data source)***
HP_ProcessorChip ref Antecedent	Property of HP_RealizesProcessor	Object path of the HP_ProcessorChip Instance.
HP_Processor ref Dependent	Property of HP_RealizesProcessor	Object path of the HP_Processor Instance.

table 5: HP_ProcessorChipInLocation properties

Table 5 describes the properties of the HP_ProcessorChipInLocation association class (associating HP_ProcessorChip and HP_ProcessorLocation). It has three columns. The first is the property name (including type and units), the second is the property inheritance (indicating which class or superclass defines the property), and the third is the property's value and data source. Each row describes a property.

Property name	***Property inheritance***	***Property value (and data source)***
HP_ProcessorChip ref Element	Property of HP_ProcessorChipInLocation	Object path of the HP_ProcessorChip Instance.
HP_ProcessorLocation ref PhysicalLocation	Property of HP_ProcessorChipInLocation	Object path of the HP_ProcessorLocation Instance.

table 6: HP_ProcessorCollection supported properties (Properties that are not supported are not mentioned.)

This class represents the processor-subsystem, i.e. the collection of processors (CPUs) in the computer system. The getInstance() method is not supported for this association class.

Property name	***Property inheritance***	***Property value (and data source)***
String InstanceID (Key)	"Inherited from CIM_SystemSpecificCollection"	"Hewlett-Packard:diags.sfm:<CreationClassName>:<LocalID> CreationClassName reflects the collection class name. LocalID is always 1, as we are creating only 1 instance of collection class."
String Caption	"Inherited from HP_GroupSystemSpecificCollection"	"HP_ProcessorCollection"
Uint16[] GroupOperationalStatus	"Inherited from HPV_GroupSystemSpecificCollection"	ValueMap {"0", "1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15", "16", "17"}, Values {"Unknown", "Other", "OK", "Degraded", "Stressed", "Predictive Failure", "Error", "Non-Recoverable Error", "Starting", "Stopping", "Stopped", "In Service", "No Contact", "Lost Communication", "Aborted", "Dormant", "Supporting Entity in Error", "Completed"}
String[] GroupStatusDescriptions	"Inherited from HP_GroupSystemSpecificCollection"	"Depending on the corresponding GroupOperationalStatus value, this will be one of the

		following "All member devices are OK." "At least one member device is degraded." "At least one member device is Stressed." "At least one member device has Unknown Status."

table 7: HP_ProcessorGroupHostedCollection supported properties. (Properties that are not supported are not mentioned.)

Instances of this class associate the Processor Collection with Computer System that contains it. The getInstance() method is not supported for this association class.

Property name	***Property inheritance***	***Property value (and data source)***
CIM_ComputerSystem ref Antecedent	Property of HP_GroupHostedCollection	The reference to the CIM_ComputerSystem.
CIM_SystemSpecificCollection ref Dependent	Property of HP_GroupHostedCollection	The reference to the processor-collection of the system.

table 8: HP_MemberOfProcessorCollection supported properties. (Properties that are not supported are not mentioned.)

This class associates the Processor/ProcessorChip instances to the ProcessorCollection to which they belong. These instances contribute to the GroupOperationalStatus of the processor-subsystem.

Property name	***Property inheritance***	***Property value (and data source)***
HP_ProcessorCollection REF Collection	Property of HP_GroupHostedCollection	Object path of HP_ProcessorCollection.
HP_Processor REF Member	Overridden by HP_GroupHostedCollection	Object path of HP_Processor

table 9: intrinsic methods for all the CIM classes supported by CPU Instance Provider

Table 9 describes the intrinsic methods supported by this provider. It has three columns. The first is the method name, the second is a description of the provider's actions based on invoking that method, and the third is a list of any exceptions that could result from invoking the method. Each row describes a method.

Method name	***Description***	***Exceptions thrown***
enumerateInstances	Returns all instances of class with values of supported properties. (See tables above.)	
enumerateInstanceNames	Returns object path of all instances of class.	
getInstance	Returns an instance that matches the keys with values of supported properties. (See tables above.)	
modifyInstance	This operation is not supported by the CPU Instance Provider. This is indicated to the client, via exceptions.	CIMNotSupportedException

deleteInstance	This operation is not supported by the CPU Instance Provider. This is indicated to the client, via exceptions.	CIMNotSupportedException
createInstance	This operation is not supported by the CPU Instance Provider. This is indicated to the client, via exceptions.	CIMNotSupportedException

indications generated by this provider

This Provider does not currently generate any indications.

Related Documentation

- **WBEM information**
  - For a CIM tutorial, go to http://www.dmtf.org/education/tutorials
  - For information about HP WBEM Services go to http://h71000.www7.hp.com/openvms/products/wbem/wbem_index.html.
  - HP WBEM Providers Release Notes bundled with the WBEM Providers kit.
  - HP WBEM Providers Installation and Administrator's Guide bundled with the WBEM Providers kit.

For additional information on HP products and services, visit us at http://www.hp.com.

Technical information contained in this document is subject to change without notice.

9/2007