Power Systems

Hardware Management Console REST APIs

Power Systems

Hardware Management Console REST APIs

Note
Before using this information and the product it supports, read the information in "Notices" on page 205.

This edition applies to IBM Hardware Management Console Version 8 Release 8.4.0 Maintenance Level 0 and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

HMC REST APIs

Learn about IBM® Power Systems™ Hardware Management Console (HMC) Representational State Transfer (REST) application program interfaces (APIs). REST is a style of web service, hosted by a web server. In REST, the web server focus is on managing nouns. Nouns represent known things like logical partitions, managed systems. Each noun has a unique identity. GET, PUT, POST, and DELETE HTTP methods are used to act on the nouns. Internet Media Type (MIME Type) identifies the types of data. HTTP request headers, response headers, and status codes are a vital part of the application. The addition of hyper links into the data content of the nouns expands the REST web services into a richer web application.

The HMC provides systems administrators a tool for planning, deploying, and managing IBM Power Systems servers.

HMC provides functions such as IBM Power Systems hardware management and virtualization (partition) management.

The HMC REST web services provide the following services:

1. Power Systems server virtualization
2. Performance Capacity and Monitoring
3. Power® Enterprise Pools (CoD)

Programmatic consumers of HMC REST web services are:

- A web-based client capable of using Atom Publishing Protocol and XML payloads.
- A client that can communicate over HTTP protocol and parse XML content.
  - Examples include the following programming languages: Python, Java™, Ruby, PHP, C++, C#

URL Model

- **Design goals:**
  - Simple URL patterns
  - Polymorphic URL patterns
  - Use Atom Publishing Protocol
  - Enable rich web application
  - Use polymorphic job pattern when asynchronous or RPC style interactions required
- **Concepts:**
  - Anchor URL patterns provide services for a type of root/child element
  - Instance URL patterns provide services for a uniquely identified root/child element
- **URL Pattern Grammar building blocks:**
  - {R} root element type
  - {C} child element type
  - {D} detail element type. {D} can be accessed as part of {R} or {C}
  - {UUID} a unique UUID value
  - {OP} the name of a type of job (an operation)
  - {QUERY} a search query
  - {JOBID} the ID of a submitted job
  - {QP} a quick path variable name
  - {EXTENDED} extended group list

HTTP Port

HMC REST web services operate on 12443 HTTP Port. The following URL is an example of the full URL that must be used to get the management console information:

`https://{HMC}:12443/rest/api/uom/ManagementConsole`. Here, *HMC* can be the IP address of the HMC or the host name of the HMC.

URL Model Use Cases for Root Elements

- ROOT ANCHOR URL patterns:
  - /rest/api/uom/{R}/operations ----------> get the defined job operations for {R}
  - /rest/api/uom/{R}/jobs ----------> get all known jobs for {R}
  - /rest/api/uom/{R}/quick ----------> get the defined quick properties for {R}
  - /rest/api/uom/{R}/quick/all ----------> get a list of all defined quick properties for type {R}
  - /rest/api/uom/{R}/search ----------> get the defined search parameters for {R}
  - /rest/api/uom/{R}/search/{QUERY} ----> get the feed of instances of {R} matching the query
  - /rest/api/uom/{R} ----------> get the feed of all known instances of {R}
- ROOT INSTANCE URL patterns:
  - /rest/api/uom/{R}/{UUID}/do/{OP} -------> get a template for job of type {OP}
  - /rest/api/uom/{R}/{UUID}/jobs ----------> get all known jobs for this element of {R}
  - /rest/api/uom/{R}/{UUID}/jobs/{JOBID} -> get the details for the one job
  - /rest/api/uom/{R}/{UUID}/quick/{QP} ----> get the quick property value of {QP}
  - /rest/api/uom/{R}/{UUID} ----------> get the XML details of this instance of {R}

URL Model for Child Elements Uses the Same Patterns. (Must be anchored to an instance of a parent Root Element)

- CHILD ANCHOR URL patterns:
  - /rest/api/uom/{R}/{UUID}/{C}/operations ----------> get the defined job operations for {C}
  - /rest/api/uom/{R}/{UUID}/{C}/jobs ----------> get all known jobs for {C}
  - /rest/api/uom/{R}/{UUID}/{C}/quick ----------> get the defined quick properties for {C}
  - /rest/api/uom/{R}/{UUID}/{C}/search ----------> get the defined search parameters for {C}
  - /rest/api/uom/{R}/{UUID}/{C}/search/{QUERY} -----> get the feed of instances of {C} matching the query
  - /rest/api/uom/{R}/{UUID}/{C}/ ----------> get the feed of all known instances of {C}
- CHILD INSTANCE URL patterns:
  - /rest/api/uom/{R}/{UUID}/{C}/{UUID}/do/{OP} ------> get a template for job of type {OP}
  - /rest/api/uom/{R}/{UUID}/{C}/{UUID}/jobs ----------> get all known jobs for this element of {C}
  - /rest/api/uom/{R}/{UUID}/{C}/{UUID}/jobs/{JOBID} -> get the details for the one job
  - /rest/api/uom/{R}/{UUID}/{C}/{UUID}/quick/{QP} ---> get the quick property value of {QP}
  - /rest/api/uom/{R}/{UUID}/{C}/{UUID}/ ----------> get the XML details of this instance of {C}

Schema Definition File

Every {R} or {C} or detailed object is bound by rules defined in a corresponding XML schema file. Using the URL pattern described, any object schema definition can be fetched.

- /rest/api/web/schema/{R}.xsd ----> Substitute the {R} for Root Object
- /rest/api/web/schema/{C}.xsd ----> Substitute the {C} for Child Object
- /rest/api/web/schema/{D}.xsd ----> Substitute the {D} for Detailed Object

Extended Group

The complete set of attributes that are defined under different HMC objects and supported by REST APIs are classified under different groups. List of all supported groups is provided below:

- None - Get none of the supported extended group attributes for an object type. Get only the default attributes.
- Advanced - Group of attributes that are classified as Advanced.
- SystemNetwork - Attributes and objects that are related to Managed System's Virtual Network.
- ViosStorage - Objects that are related to Storage managed by VIOS.
- Hypervisor - Attributes and objects that are retrieved from Hypervisor.
- ViosNetwork - Objects that are related to Virtual Network managed by VIOS.
- ViosFCMapping - List of configured VirtualFibreChannelMapping.
- ViosSCSIMapping - List of configured VirtualSCSIMapping.

An attribute that is not classified under any of the defined groups is implicitly assumed to be classified under group Default. The users of the APIs can use the groups by passing one or more extended groups as query parameter. The extended group is supported by HTTP methods - GET, PUT, POST, and DELETE.

Usage

If an object does not support any known extended group that is defined, calling the URL with group=None might result in HTTP Status code 400 Bad Request response

- /rest/api/uom/{R}?group={EXTENDED} ----------> Supports one or more of query parameters gets {R} with mentioned extended group
- /rest/api/uom/{R}/{UUID}?group={EXTENDED} ----------> Supports one or more of query parameters gets {R}/(UUID) with mentioned extended group
- /rest/api/uom/{R}/{UUID}/{C}?group={EXTENDED} ----------> Supports one or more of query parameters gets {C} with mentioned extended group
- /rest/api/uom/{R}/{UUID}/{C}/{UUID}?group={EXTENDED} ----------> Supports one or more of query parameters gets {C}/(UUID) with mentioned extended group

Related concepts:

"Virtual Network Management" on page 25
This section provides the description and APIs for virtual network management operations that can be performed within a managed system.

"Virtual Storage Management" on page 47
Virtual Storage Management provides end-to-end view of the storage assigned to client logical partition.

"Cluster" on page 52
A Cluster is a set of one or more networked Virtual I/O Server (VIOS) partitions, where each VIOS within the cluster has access to a common set of physical volumes.

Related reference:

"Managed System" on page 10
Managed System API provides a list of all system information managed by Hardware Management Console (HMC), such as system name, system machine type, model and serial number, system state, system capabilities, IP address, system migration information, system processor, system memory, and system I/O adapters information for all of the managed system. In addition, each managed system has a list of all child objects, such as:

"Logical Partition" on page 13
Logical Partition provides information about AIX®, Linux or IBM i partitions.

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical

partitions within the server.

HTTP Protocol

The Web Services API provides an extensive set of operations that client applications can invoke to obtain information about the manageable resources of the system, to change those resource characteristics, and to operate on them. Because the API is designed using a web services orientation, these operations are accessed by Hypertext Transport Protocol (HTTP) messages across TCP/IP network connections.

The Web Services API has been designed in accordance with the HTTP version 1.1 protocol, as defined in the W3C internet standards document **RFC 2616, Hypertext Transfer Protocol – HTTP version 1.1, June 1999**. This RFC can be found in HTML format at Hypertext Transfer Protocol website (http://www.w3.org/Protocols/rfc2616/rfc2616.htm).

The API requires that all clients interact using the HTTP version 1.1 protocol. The API does not support clients that use HTTP version 1.0 protocol.

HTTP Header Field Usage

HTTP request and response messages include elements known as headers fields (often referred to as headers) that provide request metadata. Certain headers are required or provided in all HTTP messages, while others are present in selected messages depending on the content.

Related reference:

"Request Headers"
The following HTTP request headers are relevant to all request methods (GET, PUT, POST and DELETE) and are required on all API requests.

"Response Headers" on page 6
The following section describes the required and optional response headers. The response headers are applicable to HTTP methods (PUT, GET, POST and DELETE).

"HTTP Status Codes" on page 7
The MC API uses standard HTTP status codes in response to indicate the success or failure of the request. Unless stated otherwise in the description of an operation, the following general interpretations of the status code values apply.

Request Headers

The following HTTP request headers are relevant to all request methods (GET, PUT, POST and DELETE) and are required on all API requests.

Required Request Header Fields

Table 1. HTTP Header

HTTP Header	Description
Host	Specifies the internet host and port number of the HMC to which the request is being directed, as obtained from the original URI specified by the client application. The Web Services API enforces that the header is provided as required by the HTTP protocol, but does not check or use the value of the header.

Table 1. HTTP Header (continued)

HTTP Header	Description
X-API-Session	An opaque string that provides a cryptographically strong identifier of the API session (known as a session id) under which this request is executed. This header is required on all requests that require authentication. The Login operation begins a new HMC session and includes credentials identifying the HMC user for the session. Upon successful authentication, the Login operation returns the value to be used in the X-API-Session header for all subsequent requests of the same session. Failure to include this header on a request requiring authentication results in status code 403 (Forbidden) with reason code 4. Specifying an invalid session id results in status code 403 (Forbidden) with reason code 5.
Content-Length	When used in a request, specifies the length of the request body. If omitted, the request is presumed to not contain a body.
Content-Type	When used in a request, specifies the MIME media type of the request body contained in the request. This header is required if the Content-Length header is supplied and specifies a non-zero request body length, otherwise status code 400 (Bad Request) will result.
Accept	Specifies the list of response MIME media types that the client application is prepared to accept for the response to the request. This header is provided for content negotiation between the client and the server in cases where the Web Services API supports multiple possible response media types for a given operation. If the header is included, it must allow the media types that the operation supports, otherwise the request will fail with HTTP status code 406 (Not Acceptable).

Optional Request Header Fields

Table 2. HTTP Header.

HTTP Header	Description
X-Audit-Memento	The HMC REST Web Services logs each consumer request into its Audit log. The Audit log is formatted using RFC-5424 extended syslog formatting. For each GET, PUT, POST, or DELETE request that is written to the Audit log, if the X-Audit-Memento header has been specified, that value will be placed into the Audit log. The header name will not be placed in the Audit log.

Table 2. HTTP Header (continued).

HTTP Header	Description
X-Client-Correlator	An opaque string that provides a cryptographically strong identifier of the API session (known as a session id) under which this request is executed. This header is required on all requests that require authentication. The Login operation begins a new HMC session and includes credentials identifying the HMC user for the session. Upon successful authentication, the Login operation returns the value to be used in the X-API-Session header for all subsequent requests of the same session. Failure to include this header on a request requiring authentication results in status code 403 (Forbidden) with reason code 4. Specifying an invalid session id results in status code 403 (Forbidden) with reason code 5.
If-Match	Supplies a previously received ETag value or wildcard, and enables conditional PUT or POST operations based on a match of the supplied value with the actual value currently known to the server.
If-None-Match	Supplies a previous received ETag value or wildcard, and enables conditional PUT or POST operations based on not matching the supplied value.
X-HMC-Schema-Version	Specifies the schema version. When specified with an HMC version 1.1.0 against an HMC version 1.2.0, response contains attributes introduced till version 1.1.0. If schema version is not specified, the latest version supported by HMC is returned as a response.
X-Transaction-ID	When set by the client code with an alphanumeric value, the transaction ID is logged in the log files for debugging. When the client is not specifying this value, REST server auto generates this value for the incoming request. The main purpose is to filter various logs at once for a single transaction ID, to ease the debugging

Related reference:

"Response Headers"
The following section describes the required and optional response headers. The response headers are applicable to HTTP methods (PUT, GET, POST and DELETE).

Response Headers

The following section describes the required and optional response headers. The response headers are applicable to HTTP methods (PUT, GET, POST and DELETE).

Response Headers

The following HTTP response headers are always provided in the response to all requests.

Table 3. Required Response Parameters

HTTP Header	Description
Date	The date and time, from the perspective of the management console clock, at which the response message was generated. As required by the HTTP protocol specification, this date is an HTTP full date sent in the RFC 1123-defined fixed length format. Example: Sun, 08 Oct 1961 10:08:00 GMT
X-Transaction-ID	Returns the same X-Transaction-ID set by the client. If it is set by the client, server auto-generated alpha numeric value is returned.

Optional Response Headers

The following HTTP response headers are provided in response to all requests except those that result in a 204 (No Content) HTTP status code.

Table 4. Optional Response Parameters

HTTP Header	Description
Content-Length	When used in a response, specifies the length of the response body. If omitted, the response does not contain a body.
X-HMC-Schema-Version	Specifies the schema version. When specified with an HMC version 1.1.0 against an HMC version 1.2.0, response contains attributes introduced till version 1.1.0. If the schema version is not specified, the latest version supported by HMC is returned as response.
ETag	Used to assist in caching of the response by the client. Holds an opaque identifier or checksum of the returned element.
Last-Modified	Used to assist in caching of the response by the client. Holds an opaque timestamp of the returned element. Granularity of the time stamp is one second.
Location	Returned on PUT. Holds the URL of the newly created element.

Related reference:

"Response Headers" on page 6
The following section describes the required and optional response headers. The response headers are applicable to HTTP methods (PUT, GET, POST and DELETE).

HTTP Status Codes

The MC API uses standard HTTP status codes in response to indicate the success or failure of the request. Unless stated otherwise in the description of an operation, the following general interpretations of the status code values apply.

Table 5. HTTP Status Codes

HTTP Status Code	HTTP Status Message	Description
200	OK	The request has succeeded completely. A response body is provided that contains the results of the request.
204	No Content	The request succeeded completely, and no additional response information is provided.
400	Bad Request	The request was missing required input, had errors in the provided input, or included extraneous input. Additional information regarding the error is provided in an error response body that includes a reason code with additional information.
403	Forbidden	Multiple error conditions result in this status code: • The request requires authentication but no X-API-Session header was provided, or one was provided but the session ID was invalid. • The user under which the API request was authenticated is not authorized to perform the requested operation.
404	Not Found	Multiple error conditions result in this status code: • The URI does not designate an extant resource, or designates a resource for which the API user does not have object-access permission. • The URI designates a resource or operation that is not supported by the MC because it is currently the alternate MC.
405	Method Not Allowed	The request specifies an HTTP method that is not valid for the designated URI.
409	Conflict	The managed resource is in an incorrect state (status) for performing the requested operation. Additional information regarding the error is provided in an error response body that includes a reason code with additional information.
412	Resource Modified	Precondition Failed. If client supplied ETag and server ETag match then proceed with POST. Otherwise results in 412 resource modified error.
413	Request Body Too Large	The request includes a request body that is too large.

Table 5. HTTP Status Codes (continued)

HTTP Status Code	HTTP Status Message	Description
415	Unsupported Media Type	The Content-Type header for the request specifies a representation that is not supported by the Web Services API.
500	Internal Server Error	A server error occurred during processing of the request.
501	Not Implemented	The request specifies an HTTP method that is not recognized by the server (for any resource).
503	Service Unavailable	The request could not be carried out by the MC due to some temporary condition.
505	HTTP Version Not Supported	The request specifies an HTTP protocol version that is not supported by the Web Services API.

Related reference:

"Response Headers" on page 6
The following section describes the required and optional response headers. The response headers are applicable to HTTP methods (PUT, GET, POST and DELETE).

Logon

Logon request establishes a trusted session with the Web Services APIs.

The API receives UserID and Password as Logon Request and responds with X-API-Session. This establishes a valid user session. The X-API-Session must be used for all subsequent REST API calls.

Resource

`/rest/api/web/Logon`

Since: Version 1_1_0

Table 6. Supported methods

Method	Description	Internet media type or content type
PUT	Valid credentials establishes web services session with the server and returns X-API-Session	`application/vnd.ibm.powervm.web+xml;type=LogonRequest`
DELETE	This method is used to request the web server to delete the session.	`application/vnd.ibm.powervm.web+xml;type=LogonRequest`

Example request

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<LogonRequest xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 schemaVersion="V1_1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <UserID kb="CUR" kxe="false">{hmc_user_id}</UserID>
    <Password kb="CUR" kxe="false">{hmc_password}</Password>
</LogonRequest>
```

Example response

In case of successful execution the server returns response with X-API-Session.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<LogonResponse xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 schemaVersion="V1_1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <X-API-Session kxe="false" kb="ROR">
    gDHtF9lOmKjiHPkdaR0Bv4Ep1fXn8Qh4H8okf88pJgz9mVRT3hG_XD8Iu3d4fZ1KB95ytiNusdu
    hpKSdMFXCBQgY4YcABiPyztI8ZZSTqk33bqDrDnmdrmJdyCXLWDYgFCIGs6Cba_fs7_83JvaVAb
    V7zaTwmpiwfVD2r_8faiawZ2SiPNN9fDbvQHx0Z1S6KT92osIo4d09U2J1C4aJKtJWs65JcPkQD-wtunLa_j4=
    </X-API-Session>
</LogonResponse>
```

In your all subsequent REST calls add the X-API-Session as a request header parameter to maintain the session.

Management Console

Management Console the Hardware Management Console (HMC), used to administer IBM Power Systems servers and related resources.

Resource

`/rest/api/uom/ManagementConsole/{uuid}`

`/rest/api/uom/ManagementConsole`

Management Console lists the following information:

1. Links to the systems managed by this management console.
2. Network information regarding the management Console.
3. Version of the management console.

Related reference:

"Managed System"
Managed System API provides a list of all system information managed by Hardware Management Console (HMC), such as system name, system machine type, model and serial number, system state, system capabilities, IP address, system migration information, system processor, system memory, and system I/O adapters information for all of the managed system. In addition, each managed system has a list of all child objects, such as:

Managed System

Managed System API provides a list of all system information managed by Hardware Management Console (HMC), such as system name, system machine type, model and serial number, system state, system capabilities, IP address, system migration information, system processor, system memory, and system I/O adapters information for all of the managed system. In addition, each managed system has a list of all child objects, such as:

- "Logical Partition" on page 13
- "Virtual I/O Server" on page 18
- "Shared memory pool" on page 21
- "Shared Processor Pool" on page 23
- "Host Ethernet Adapter" on page 45
- "Reserved storage device pool" on page 22
- "Virtual Network" on page 26

- "Virtual Switch" on page 25
- "Network Bridge" on page 27

Resource

```
/rest/api/uom/ManagedSystem
```

This API provides all system information in the console.

```
/rest/api/uom/ManagedSystem/{uuid}
```

This API provides system information of the given instance.

Quick properties

User can fetch only the specified property.

Resource

```
/rest/api/uom/ManagedSystem/{uuid}/quick/{Property Name}
```

Supported property names available at /rest/api/uom/ManagedSystem/quick URI.

Table 7. List of quick properties

Property Name	Description
State	Fetch current state of the system.
SystemName	Fetch name of the system.
IPAddress	Fetch IP address of the system.
MTMS	Fetch system machine type, model, and Serial number
PhysicalSystemAttentionLEDState	Fetch LED state of the system
MemoryDefragmentationState	Fetch memory defragmentation state of the system

Extended groups

Certain attribute always make calls to the underlying layer to get the data. These attributes will not be cached and populated in REST API unless user requested it.

Resource

```
/rest/api/uom/ManagedSystem?group={Group Name}
```

This API provides system information of the specified extended group attributes.

```
/rest/api/uom/ManagedSystem/{uuid}?group={Group Name}
```

This API provides the specified extended group attributes of the given instance.

Table 8. List of extended groups

Group name	Description
EnergyManagement	This is an extended group, returns Energy Scale settings and tuanble parameters details.
Hypervisor	This is an extended group which makes a call to hypervisor to get the data.
SystemNetwork	This extended group returns network related information like network bridge, virtual network and virtual switch.
None	This extended group returns only default attributes.

Table 9. List of Methods

Method Name	Description	Internet Media Type
GET	It retrieves information identified by the Request-URI. Produced data is returned as the entity in the response.	application/ vnd.ibm.powervm.uom+xml; Type=ManagedSystem
POST	It is used to request the web server to accept the entity enclosed in the request and allow them to change state of the entity.	application/ vnd.ibm.powervm.uom+xml; Type=ManagedSystem

Since: Version 1.1.0

Search

User can search a resource by specifying a property name and value in the managed system URI to get the matching instance.

Resource

```
/rest/api/uom/ManagedSystem/search/({Property name}=={Property value})
```

Supported property name available at /rest/api/uom/ManagedSystem/search URI.

List of jobs supported for managed system

- "PowerOff_ManagedSystem Job" on page 64
- "PowerOn_ManagedSystem Job" on page 64
- "ResetConnection_ManagedSystem Job" on page 65
- "RemoveConnection_ManagedSystem Job" on page 66
- "QueryReservedMemoryRequiredForPartition_ManagedSystem Job" on page 66
- "ModifySRIOVAdapterMode_ManagedSystem Job" on page 67
- "ClearSRIOVPhysicalPortStatistics_ManagedSystem Job" on page 69

Related reference:

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

"Shared memory pool" on page 21
Active Memory Sharing (AMS) allows selected logical partitions to share memory from a single pool of physical memory. This new function is supported by a new level of abstraction managed by the Hypervisor.

"Shared Processor Pool" on page 23
Shared Processor Pool is a pool of shared processors whose processing capacity is shared among multiple logical partitions. Logical partition is allocated with the shared processor pool.

"Host Ethernet Adapter" on page 45
This API provides the details of the port configuration of the physical Host Ethernet Adapters on the managed system, and also allows to change the configuration of any of the ports on an HEA. Host Ethernet Adapter is child object of ManagedSystem.

"Reserved storage device pool" on page 22
Reserved storage device pool is a pool of physical memory made up of various discs. Logical partition is

allocated with the reserved storage device pool.

"SR-IOV Adapter" on page 42
Single Root I/O Virtualization Adapter provides I/O virtualization capabilities for the managed system. The APIs provide the configuration of SR-IOV adapters.

"Virtual Network" on page 26
Virtual Network is a managed system level object representing the Virtual LAN connectivity across the logical partitions.

"Virtual Switch" on page 25
PowerVM Virtual Switch APIs

"Network Bridge" on page 27
Network Bridge is the REST representation of the Shared Ethernet Adapters. NetworkBridge is a wrapper around the Shared Ethernet Adapter (SEA) and provides a unique way of managing the network bridge functionality for all configurations such as Failover, LoadBalancing, and Non-Failover.

Logical Partition

Logical Partition provides information about AIX, Linux or IBM i partitions.

Resource

This API provides information about all the logical partitions managed by a particular managed system given its UUID:

```
/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/LogicalPartition
```

This API provides information about a particular logical partition which is being managed by a particular managed system given their respective UUIDs:

```
/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/LogicalPartition/{LogicalPartition_uuid}
```

This API provides information about all the logical partitions managed by the HMC:

```
/rest/api/uom/LogicalPartition
```

This API provides information about a particular logical partition given the UUID:

```
/rest/api/uom/LogicalPartition/{LogicalPartition_uuid}
```

Resource

```
/rest/api/uom/LogicalPartition/{LogicalPartition_uuid}/quick/{Property name}
```

Supported property names available at /rest/api/uom/LogicalPartition/quick URI.

Table 10. Quick properties

Quick property	Description
IsVirtualServiceAttentionLEDOn	The virtual service attention LED state.
MigrationState	The state of the partition's migration operation.
ProgressState	The progress state of the partition's hibernation operation.
PartitionType	The partition environment, as an enumeration key (for example, 'AIX_LINUX', 'OS_400', etc.).
PartitionName	The name of the partition.
PartitionID	The integer ID of the partition.
PartitionState	The state of the partition.
RemoteRestartState	The state of the partition's Remote Restart operation.

Table 10. Quick properties (continued)

Quick property	Description
AssociatedManagedSystem	The REST URI of the partition's parent managed system.
RMCState	The state of the partition's Resource Monitoring Control (RMC) connection.
PowerManagementMode	The power management mode.

Extended groups

There are certain attribute which always makes call to its underlying layer to get the data. Those attributes are not cached and populated in REST API unless the user requested for them.

Resource

```
/rest/api/uom/LogicalPartition?group={Group Name}
```

This API provides all the Logical Partition information of the specified extended group attributes.

```
/rest/api/uom/LogicalPartition/{LogicalPartition_uuid}?group={Group Name}
```

This API provides the specified extended group attributes of the given instance.

```
/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/
LogicalPartition?group={Group Name}
```

This API provides the specified extended group attributes of the given instance.

```
/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/LogicalPartition/
{LogicalPartition_uuid}?group={Group Name}
```

This API provides the specified extended group attributes of the given instance.

Table 11. List of extended groups

Group name	Description
Advanced	This is an extended group that makes a call to hypervisor to get the data.
None	This is an extended group that provides the default attributes.

Table 12. Supported methods

Method	Description	Internet media type
GET	It retrieves information identified by the Request-URI. Produced data is returned as the entity in the response.	`application/ vnd.ibm.powervm.uom+xml; Type=LogicalPartition`
PUT	It is used to request the web server to accept the entity enclosed in the request and allow them to create a new instance.	`application/ vnd.ibm.powervm.uom+xml; Type=LogicalPartition`
POST	It is used to request the web server to accept the entity enclosed in the request and allow them to change state of the entity.	`application/ vnd.ibm.powervm.uom+xml; Type=LogicalPartition`
DELETE	It is used to request the web server to delete the requested entity.	`application/ vnd.ibm.powervm.uom+xml; Type=LogicalPartition`

Since: Version 1_1_0

Search

User can search a resource by giving a property name and value in the LogicalPartition URI to get the matching instance.

Resource

```
/rest/api/uom/LogicalPartition/search/({Property name}=={Property value})
```

Supported property name available at /rest/api/uom/LogicalPartition/search URI.

List of jobs supported for logical partition

- "GetNetworkBootDevices_LogicalPartition Job" on page 70
- "LogicalPartition_RemoteRestart Job" on page 71
- "Migrate_LogicalPartition Job" on page 74
- "MigrateAbort_LogicalPartition Job" on page 76
- "MigrateRecover_LogicalPartition Job" on page 77
- "MigrateValidate_LogicalPartition Job" on page 78
- "PowerOff_LogicalPartition Job" on page 81
- "PowerOn_LogicalPartition Job" on page 82

Related reference:

"Managed System" on page 10
Managed System API provides a list of all system information managed by Hardware Management Console (HMC), such as system name, system machine type, model and serial number, system state, system capabilities, IP address, system migration information, system processor, system memory, and system I/O adapters information for all of the managed system. In addition, each managed system has a list of all child objects, such as:

"SR-IOV Ethernet Logical Port" on page 43
REST APIs for configuring SR-IOV Ethernet Logical Ports on LogicalParition and VirtualIOServer.

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

"Logical Partition Profile" on page 20
Logical partition profile is a record on the Hardware Management Console (HMC) that specifies a possible set of resource allocations and activation settings for a logical partition. Logical Partition Profile API is a child of Logical Partition API and Virtual IO Server API.

Virtual SCSI Client Adapter

Virtual Small Computer Systems Interface (SCSI) is based on a client/server relationship. The Virtual I/O Server (VIOS) owns the physical resources and acts as server or, in SCSI terms, target device. The client logical partitions access the virtual SCSI backing storage devices provided by the VIOS as clients. The client partition accesses its assigned disks through a virtual SCSI client adapter. The virtual SCSI client adapter sees the disks, logical volumes or file-backed storage through this virtual adapter as virtual SCSI disk devices.

Resource

```
/rest/api/uom/LogicalPartition/{LogicalPartition_uuid}/VirtualSCSIClientAdapter/
{VirtualSCSIClientAdapter_uuid}
```

Table 13. Supported methods

Method	Description	Internet media type
GET	It retrieves information identified by the Request-URI. Produced data is returned as the entity in the response.	`application/ vnd.ibm.powervm.uom+xml; Type=VirtualSCSIClientAdapter`
PUT	It is used to request the web server to accept the entity enclosed in the request and allow them to create a new instance.	`application/ vnd.ibm.powervm.uom+xml; Type=VirtualSCSIClientAdapter`
DELETE	It is used to request the web server to delete the requested entity.	`application/ vnd.ibm.powervm.uom+xml; Type=VirtualSCSIClientAdapter`

Since: Version 1.1.0

Related reference:

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

Virtual Fiber Channel Client Adapter

With N-Port ID Virtualization (NPIV), you can configure the managed system so that multiple logical partitions can access independent physical storage through the same physical Fiber Channel adapter. The Virtual Fiber Channel Client Adapter API is a child of Logical Partition API.

Resource

`/rest/api/uom/LogicalPartition/{LogicalPartition_uuid}/VirtualFibreChannelClientAdapter/ {VirtualFibreChannelClientAdapter_uuid}`

Table 14. Supported methods

Method	Description	Internet media type
GET	It retrieves information identified by the Request URI. Produced data is returned as the entity in the response.	`application/vnd.ibm.powervm.uom+xml; Type=VirtualFibreChannelClientAdapter`
PUT	It is used to request the web server to accept the entity enclosed in the request and allow them to create a new instance.	`application/vnd.ibm.powervm.uom+xml; Type=VirtualFibreChannelClientAdapter`
POST	It is used to request the web server to accept the entity enclosed in the request and allow them to change state of the entity.	`application/vnd.ibm.powervm.uom+xml; Type=VirtualFibreChannelClientAdapter`
DELETE	It is used to request the web server to delete the requested entity.	`application/vnd.ibm.powervm.uom+xml; Type=VirtualFibreChannelClientAdapter`

Since: Version 1.1.0

Related reference:

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

Client Network Adapter

Client Network Adapter API allows logical partitions to communicate with each other without having to assign physical hardware to the logical partitions. Client Network Adapter API is a child of Logical Partition API.

Resource

`/rest/api/uom/LogicalPartition/{LogicalPartition_uuid}/ClientNetworkAdapter/{ClientNetworkAdapter_uuid}`

Table 15. Supported methods

Method	Description	Internet media type
GET	It retrieves information identified by the Request-URI. Produced data is returned as the entity in the response.	`application/ vnd.ibm.powervm.uom+xml; Type=ClientNetworkAdapter`
PUT	It is used to request the web server to accept the entity enclosed in the request and allow them to create a new instance.	`application/ vnd.ibm.powervm.uom+xml; Type=ClientNetworkAdapter`
POST	It is used to request the web server to accept the entity enclosed in the request and allow them to change state of the entity.	`application/ vnd.ibm.powervm.uom+xml; Type=ClientNetworkAdapter`
DELETE	It is used to request the web server to delete the requested entity.	`application/ vnd.ibm.powervm.uom+xml; Type=ClientNetworkAdapter`

Since: Version 1_1_0

Related reference:

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

"Virtual Switch" on page 25
PowerVM Virtual Switch APIs

"Network Bridge" on page 27
Network Bridge is the REST representation of the Shared Ethernet Adapters. NetworkBridge is a wrapper around the Shared Ethernet Adapter (SEA) and provides a unique way of managing the network bridge functionality for all configurations such as Failover, LoadBalancing, and Non-Failover.

"Virtual Network" on page 26
Virtual Network is a managed system level object representing the Virtual LAN connectivity across the logical partitions.

Virtual NIC Dedicated

A VirtualNICDedicated element represents all the properties and features of a dedicated, virtualized network interface controller associated with an Ethernet port in a logical partition.

Resource

/rest/api/uom/LogicalPartition/{LogicalPartition_uuid}/VirtualNICDedicated/{VirtualNICDedicated_uuid}

Table 16. Supported methods

Method	Description	Internet media type
GET	Fetch data from the server.	application/vnd.ibm.powervm.uom+xml; Type=VirtualNICDedicated
PUT	Create a new instance.	application/vnd.ibm.powervm.uom+xml; Type=VirtualNICDedicated
DELETE	Delete an instance.	application/vnd.ibm.powervm.uom+xml; Type=VirtualNICDedicated

Since: Version 1_4_0

Related reference:

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

Virtual I/O Server

The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

Resource

The Virtual IO Server API provides information about all the VIOS managed by a particular managed system given the UUID:

/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/VirtualIOServer

This API provides information about the specific VIOS that is being managed by the specific managed system given their respective UUIDs:

/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/VirtualIOServer/{VirtualIOServer_uuid}

This API provides information about all the VIOS managed by the HMC:

/rest/api/uom/VirtualIOServer

This API provides information about a particular VIOS given its UUID:

/rest/api/uom/VirtualIOServer/{VirtualIOServer_uuid}

Quick properties

User can fetch the only specified property.

Resource

/rest/api/uom/VirtualIOServer/{VirtualIOServer_uuid}/quick/{Property name}

Supported property names available at /rest/api/uom/VirtualIOServer/quick URI.

Table 17. Quick properties

Quick property	Description
IsVirtualServiceAttentionLEDOn	The virtual service attention LED state.
APICapable	API capability.
PartitionType	The partition environment, as an enumeration key (for example,'AIX_LINUX', 'OS_400', 'VIRTUAL_IO_SERVER').

Table 17. Quick properties (continued)

Quick property	Description
PartitionName	The name of the partition.
PartitionID	The integer ID of the partition.
PartitionState	The state of the partition.
AssociatedManagedSystem	The REST URI of the partition's parent managed system.
RMCState	The state of the partition's Resource Monitoring Control (RMC) connection.

Extended groups

There are certain attribute which always makes call to its underlying layer to get the data. Those attributes are not cached and populated in REST API unless user requested for them.

Resource

This API provides VIOS information of specified extended group attributes:
`/rest/api/uom/VirtualIOServer?group={Group Name}`

This API provides information of specified extended group attributes of the given instance:
`/rest/api/uom/VirtualIOServer/{VirtualIOServer_uuid}?group={Group Name}`

This API provides information of specified extended group attributes of the given instance:
`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/VirtualIOServer?group={Group Name}`

This API provides information of specified extended group attributes of the given instance:
`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/VirtualIOServer/{VirtualIOServer_uuid}?group={Group Name}`

Table 18. List of extended groups

Group name	Description
ViosStorage	This is an extended group that gets data related to storage by making a call to VIOS.
None	This is an extended group that provides the default attributes.
ViosNetwork	This is an extended group that gets data related to network by making a call to VIOS.
ViosFCMapping	This is an extended group that will provide the client and server Fiber Channel mapping details.
ViosSCSIMapping	This is an extended group that will provide the client and server SCSI client adapter mapping details.

Table 19. Supported methods

Method	Description	Internet media type
GET	It retrieves information identified by the Request-URI. Produced data is returned as the entity in the response.	`application/ vnd.ibm.powervm.uom+xml; Type=VirtualIOServer`
PUT	It is used to request the web server to accept the entity enclosed in the request and allow them to create a new instance.	`application/ vnd.ibm.powervm.uom+xml; Type=VirtualIOServer`

Table 19. Supported methods (continued)

Method	Description	Internet media type
POST	It is used to request the web server to accept the entity enclosed in the request and allow them to change state of the entity.	`application/ vnd.ibm.powervm.uom+xml; Type=VirtualIOServer`
DELETE	It is used to request the web server to delete the requested entity.	`application/ vnd.ibm.powervm.uom+xml; Type=VirtualIOServer`

Since: Version 1_1_0

Search

User can search a resource by giving a property name and value in the VIOS URI to get the matching instance.

Resource

```
/rest/api/uom/VirtualIOserver/search/({Property name}=={Property value})
```

Supported property name available at /rest/api/uom/VirtualIOServer/search URI.

Related reference:

"Managed System" on page 10
Managed System API provides a list of all system information managed by Hardware Management Console (HMC), such as system name, system machine type, model and serial number, system state, system capabilities, IP address, system migration information, system processor, system memory, and system I/O adapters information for all of the managed system. In addition, each managed system has a list of all child objects, such as:

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

Logical Partition Profile

Logical partition profile is a record on the Hardware Management Console (HMC) that specifies a possible set of resource allocations and activation settings for a logical partition. Logical Partition Profile API is a child of Logical Partition API and Virtual IO Server API.

Resource

```
/rest/api/uom/LogicalPartition/{LogicalPartition_uuid}/LogicalPartitionProfile
```

This API provides all the partition profiles information for the given logical partition.

```
/rest/api/uom/LogicalPartition/{LogicalPartition_uuid}/LogicalPartitionProfile/
{LogicalPartitionProfile_uuid}
```

This API provides the profile information of the given instance for a logical partition.

```
/rest/api/uom/VirtualIOServer/{VirtualIOServer_uuid}/LogicalPartitionProfile
```

This API provides all the partition profiles information for the given Virtual I/O Server (VIOS).

```
/rest/api/uom/VirtualIOServer/{VirtualIOServer_uuid}/LogicalPartitionProfile/
{LogicalPartitionProfile_uuid}
```

This API provides the profile information of the given instance for a Virtual IO Server API.

Since: Version 1_1_0

Resource

```
/rest/api/uom/LogicalPartition/{LogicalPartition_uuid}/LogicalPartitionProfile/quick/{Property name}
/rest/api/uom/LogicalPartition/{LogicalPartition_uuid}/LogicalPartitionProfile/
{LogicalPartitionProfile_uuid}/quick/{Property name}
/rest/api/uom/VirtualIOSerever/{VirtualIOSerever_uuid}/LogicalPartitionProfile/quick/{Property name}
/rest/api/uom/VirtualIOSerever/{VirtualIOSerever_uuid}/LogicalPartitionProfile/
{LogicalPartitionProfile_uuid}/quick/{Property name}
```

Since: Version 1_1_0

Table 20. Quick property

Property name	Description
ProfileName	Fetch the name of the logical partition profile

Table 21. Supported methods

Operation name	Description	Internet media type
GET	It retrieves information identified by the Request-URI. Produced data is returned as the entity in the response. The list of attributes produced as part of feed is found at LogicalPartitionProfile_xsd.htm	`application/vnd.ibm.powervm.uom+xml; Type=LogicalPartitionProfile`
PUT	It is used to request the web server to accept the entity enclosed in the request and allow them to create a new instance. The list of attributes that are required or allowed is found at LogicalPartitionProfile_xsd.html	`application/vnd.ibm.powervm.uom+xml; Type=LogicalPartitionProfile`
POST	It is used to request the web server to accept the entity enclosed in the request and allow them to change state of the entity. The list of attributes which allow to change is found at LogicalPartitionProfile_xsd.html	`application/vnd.ibm.powervm.uom+xml; Type=LogicalPartitionProfile`
DELETE	It is used to request the web server to delete the requested entity	`application/vnd.ibm.powervm.uom+xml; Type=LogicalPartitionProfile`

Related reference:

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

Shared memory pool

Active Memory Sharing (AMS) allows selected logical partitions to share memory from a single pool of physical memory. This new function is supported by a new level of abstraction managed by the Hypervisor.

Resource

```
rest/api/uom/ManagedSystem/[ManagedSystem_uuid]/SharedMemoryPool/[SharedMemoryPool_uuid]
```

The real physical memory is part of the shared memory pool that is virtualized by the Hypervisor to allocate physical memory to the shared memory partitions and the Hypervisor.

Table 22. Supported methods

Method	Description	Internet media type
GET	It retrieves information identified by the Request-URI. Produced data is returned as the entity in the response.	`application/ vnd.ibm.powervm.uom+xml; Type=SharedMemoryPool`
PUT	It is used to request the web server to accept the entity enclosed in the request and allow them to create a new instance.	`application/ vnd.ibm.powervm.uom+xml; Type=SharedMemoryPool`
POST	It is used to request the web server to accept the entity enclosed in the request and allow them to change state of the entity.	`application/ vnd.ibm.powervm.uom+xml; Type=SharedMemoryPool`
DELETE	It is used to request the web server to delete the requested entity.	`application/ vnd.ibm.powervm.uom+xml; Type=SharedMemoryPool`

Since: Version 1.1.0

Reserved storage device pool

Reserved storage device pool is a pool of physical memory made up of various discs. Logical partition is allocated with the reserved storage device pool.

Resource

`rest/api/uom/ManagedSystem/[ManagedSystem_uuid]/ReservedStorageDevicePool/[ReservedStorageDevicePool_uuid]`

Reserved storage device pool is collection of reserved storage devices. A reserved storage device can exist in many reserved storage device pools. When you create a reserved storage device, ensure that you add the reserved storage device to only one reserved storage device pool on the host. Otherwise, multiple LPARs might use the same reserved storage device. However, you cannot use a reserved storage device that exists in multiple reserved storage device pools.

Table 23. Supported methods

Method	Description	Internet media type
GET	It retrieves information identified by the Request-URI. Produced data is returned as the entity in the response.	`application/ vnd.ibm.powervm.uom+xml; Type=ReservedStorageDevicePool`
PUT	It is used to request the web server to accept the entity enclosed in the request and allow them to create a new instance.	`application/ vnd.ibm.powervm.uom+xml; Type=ReservedStorageDevicePool`
POST	It is used to request the web server to accept the entity enclosed in the request and allow them to change state of the entity.	`application/ vnd.ibm.powervm.uom+xml; Type=ReservedStorageDevicePool`
DELETE	It is used to request the web server to delete the requested entity.	`application/ vnd.ibm.powervm.uom+xml; Type=ReservedStorageDevicePool`

Since: Version 1.1.0

Shared Processor Pool

Shared Processor Pool is a pool of shared processors whose processing capacity is shared among multiple logical partitions. Logical partition is allocated with the shared processor pool.

Resource

`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/SharedProcessorPool`

This API provides all the shared processor pool information of the given managed system.

`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/SharedProcessorPool/{SharedProcessorPool_uuid}`

This API provides the shared processor pool information of the given instance.

Since: Version 1.1.0

Resource

`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/SharedProcessorPool/quick/{Property name}`

`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/SharedProcessorPool/{SharedProcessorPool_uuid}/quick/{Property name}`

Since: Version 1.1.0

Table 24. Quick property

Property name	Description
PoolName	Fetch the name of the Shared processor pool

Table 25. Supported methods

Method	Description	Internet media type
GET	It retrieves information identified by the Request-URI. Produced data is returned as the entity in the response. The list of attributes produced as part of feed is found at SharedProcessorPool_xsd.htm	`application/vnd.ibm.powervm.uom+xml;Type=SharedProcessorPool`
PUT	It is used to request the web server to accept the entity enclosed in the request and allow them to create a new instance. The list of attributes that are required or allowed is found at SharedProcessorPool_xsd.html	`application/vnd.ibm.powervm.uom+xml;Type=SharedProcessorPool`
POST	It is used to request the web server to accept the entity enclosed in the request and allow them to change the state of the entity. The list of attributes that allow to change is found at SharedProcessorPool_xsd.html	`application/vnd.ibm.powervm.uom+xml;Type=SharedProcessorPool`
DELETE	It is used to request the web server to delete the requested entity	`application/vnd.ibm.powervm.uom+xml;Type=SharedProcessorPool`

Related reference:

"Managed System" on page 10
Managed System API provides a list of all system information managed by Hardware Management Console (HMC), such as system name, system machine type, model and serial number, system state, system capabilities, IP address, system migration information, system processor, system memory, and system I/O adapters information for all of the managed system. In addition, each managed system has a list of all child objects, such as:

Power Enterprise Pool

Power Enterprise Pool is a group of managed systems that can share mobile Capacity on Demand (CoD) processor resources and memory resources. In addition, each pool can have a list of pool member child objects:

- "Power Enterprise Pool Member"

Resource

`/rest/api/uom/PowerEnterprisePool`

This API provides all Power Enterprise Pools managed by the management console.

`/rest/api/uom/PowerEnterprisePool/{uuid}`

This API provides Power Enterprise Pool information for the specified instance.

Table 26. Supported methods

Method	Description	Internet media type
GET	Fetch data from the server.	`application/ vnd.ibm.powervm.uom+xml; Type=PowerEnterprisePool`
POST	Update the existing instance. You can perform the following actions: • Rename Enterprise Pool name. • Add Management System to Pool. • Remove Management System from Pool.	`application/ vnd.ibm.powervm.uom+xml; Type=PowerEnterprisePool`

Since: Version 1.1.0

List of jobs supported for power enterprise pool

- "CreatePowerEnterprisePool_PowerEnterprisePool Job" on page 114
- "SetMasterConsole_PowerEnterprisePool Job" on page 118
- "SyncSystemPool_PowerEnterprisePool Job" on page 120
- "UpdatePowerEnterprisePool_PowerEnterprisePool Job" on page 120

Related reference:

"Power Enterprise Pool Member"
Power Enterprise Pool Member is a managed system that is a member of a Power Enterprise Pool.

Power Enterprise Pool Member

Power Enterprise Pool Member is a managed system that is a member of a Power Enterprise Pool.

Resource

`/rest/api/uom/PowerEnterprisePool/{uuid}/PowerEnterprisePoolMember`

This API provides the list of the Power Enterprise Pool members that are managed by the specified Power Enterprise Pool.

`/rest/api/uom/PowerEnterprisePool/{uuid}/PowerEnterprisePoolMember/{uuid}`

This API provides the Power Enterprise Pool Member information for the specified Power Enterprise Pool.

Table 27. Supported methods

Method	Description	Internet media type
GET	Fetch data from the server.	`application/ vnd.ibm.powervm.uom+xml; Type=PowerEnterprisePoolMember`
POST	Update the Pool member. • Add processor to pool member. • Remove processor from pool member. • Add memory to pool member. • Remove memory from pool member.	`application/ vnd.ibm.powervm.uom+xml; Type=PowerEnterprisePoolMember`

Since: Version 1.1.0

Virtual Network Management

This section provides the description and APIs for virtual network management operations that can be performed within a managed system.

Virtual Network Management APIs allows the users to configure and setup all of the network infrastructure required for the client partitions and VIOSs within the managed system.

Related concepts:

"Virtual Storage Management" on page 47
Virtual Storage Management provides end-to-end view of the storage assigned to client logical partition.

Related reference:

"Managed System" on page 10
Managed System API provides a list of all system information managed by Hardware Management Console (HMC), such as system name, system machine type, model and serial number, system state, system capabilities, IP address, system migration information, system processor, system memory, and system I/O adapters information for all of the managed system. In addition, each managed system has a list of all child objects, such as:

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

Virtual Switch

PowerVM Virtual Switch APIs

Resource

`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/VirtualSwitch/{VirtualSwitch_uuid}`

`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/VirtualSwitch`

Since Version 1_1_0

By default, the managed system has one virtual switch configured on a VSwitch capable managed system.

Virtual Switch has the following properties:

1. SwitchID
2. SwitchName
3. SwitchMode

Valid SwitchModes are VEB and VEPA.

Switch Mode is applicable only for the VirtualServerNetworkingPhase2Capable managed systems.

Table 28. Supported methods

Method	Description	Internet media type
GET	Fetch the specified Virtual Switch information.	`application/ vnd.ibm.powervm.uom+xml, type=VirtualSwitch`
PUT	Create a Virtual Switch on the Managed System.	`application/ vnd.ibm.powervm.uom+xml, type=VirtualSwitch`
POST	Modify Virtual Switch properties. SwitchName and SwitchMode are modifiable.	`application/ vnd.ibm.powervm.uom+xml, type=VirtualSwitch`
DELETE	Delete the Virtual Switch.	`application/ vnd.ibm.powervm.uom+xml, type=VirtualSwitch`

Related reference:

"Virtual Network"
Virtual Network is a managed system level object representing the Virtual LAN connectivity across the logical partitions.

"Network Bridge" on page 27
Network Bridge is the REST representation of the Shared Ethernet Adapters. NetworkBridge is a wrapper around the Shared Ethernet Adapter (SEA) and provides a unique way of managing the network bridge functionality for all configurations such as Failover, LoadBalancing, and Non-Failover.

"Client Network Adapter" on page 17
Client Network Adapter API allows logical partitions to communicate with each other without having to assign physical hardware to the logical partitions. Client Network Adapter API is a child of Logical Partition API.

Virtual Network

Virtual Network is a managed system level object representing the Virtual LAN connectivity across the logical partitions.

Resource

`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/VirtualNetwork/{VirtualNetwork_uuid}`

`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/VirtualNetwork`

Since: Version 1.1.0

The Virtual Network has the following properties:

- NetworkName - User defined name
- NetworkVLANID - VLAN ID of the Network
- VSwitch - The associated VirtualSwitch
- TaggedNetwork - specifies whether the network must be tagged or untagged.

The Tagged/Untagged property impacts only when the network is bridged. If the virtual network is untagged and bridged, it indicates that the traffic from all the partitions that are connected to this network are routed to the external physical network without the VLAN tagging. If the virtual network is tagged and bridged, it indicates that the traffic from all the partitions that are connected to this network are routed to the external physical network with the VLAN tag.

Note the following points while deleting a virtual network:

- Deletion of a Virtual Network that is equivalent to the Trunk Adapter PVID will fail
- Users need to delete the NetworkBridge containing this Virtual Network first and then attempt to delete the Virtual Network

Table 29. Supported methods

Method	Description	Internet media type
GET	Get the Virtual Network details	`application/ vnd.ibm.powervm.uom+xml, type=VirtualNetwork`
PUT	Create a new Virtual Network on the ManagedSystem.	`application/ vnd.ibm.powervm.uom+xml, type=VirtualNetwork`
POST	Modify Virtual Network properties. (Only NetworkName can be modified)	`application/ vnd.ibm.powervm.uom+xml, type=VirtualNetwork`
DELETE	Delete the Virtual Network.	`application/ vnd.ibm.powervm.uom+xml, type=VirtualNetwork`

Related reference:

"Virtual Switch" on page 25
PowerVM Virtual Switch APIs

"Network Bridge"
Network Bridge is the REST representation of the Shared Ethernet Adapters. NetworkBridge is a wrapper around the Shared Ethernet Adapter (SEA) and provides a unique way of managing the network bridge functionality for all configurations such as Failover, LoadBalancing, and Non-Failover.

"Client Network Adapter" on page 17
Client Network Adapter API allows logical partitions to communicate with each other without having to assign physical hardware to the logical partitions. Client Network Adapter API is a child of Logical Partition API.

Network Bridge

Network Bridge is the REST representation of the Shared Ethernet Adapters. NetworkBridge is a wrapper around the Shared Ethernet Adapter (SEA) and provides a unique way of managing the network bridge functionality for all configurations such as Failover, LoadBalancing, and Non-Failover.

Resource

`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/NetworkBridge/{NetworkBridge_uuid}`

`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/NetworkBridge`

Since: Version 1.1.0

The NetworkBridge APIs provide the interface for configuring the SEA(s) without requiring the user to log on to VIOS console. NetworkBridge is a wrapper functionality for SEAs provided as API from Management Console. The API provides configuration of NetworkBridge in the following modes:

1. NonFailover
2. Failover
3. LoadBalancing

To configure Network Bridge, the XML should have the following basic attributes:

1. FailoverEnabled - Boolean (true/false)
2. LoadBalancingEnabled - Boolean (true/false)
3. ControlChannelID - Required when failover is enabled (For 'ManagementVLANForControlChannelCapable' managed systems, this field is optional even if failover is enabled)
4. PortVLANID - The Untagged VLAN ID for the SEA(s)
5. SharedEthernetAdapters - A collection of SEAs (For Failover Enabled 2 SEAs, For Failover Disabled 1 SEA)
6. LoadGroups - A collection of LoadGroup objects
   - Minimum of One LoadGroup for NonFailover and Failover NetworkBridge.
   - For LoadBalancingEnabled, a minimum of two load groups must be specified.
   - One of the LoadGroup's PVID must be same as that of NetworkBridge PortVLANID.

When a VIOS does not have active RMC connection, the Network Bridge information is not returned. For a failover enabled NetworkBridge, if one of the VIOSs does not have active RMC connection or VIOS is shutdown, the MC will provide only one SEA in the NetworkBridge.

Table 30. Supported methods

Method	Description	Internet media type
GET	Fetch the configured/specified NetworkBridges from the Managed System	`application/ vnd.ibm.powervm.uom+xml, type=NetworkBridge`
PUT	Create a NetworkBridge	`application/ vnd.ibm.powervm.uom+xml, type=NetworkBridge`
POST	Modify the NetworkBridge configuration viz., Enable Failover, Enable Load Balancing, Add LoadGroup(s), Modify SEA properties, Add/Remove VirtualNetworks to/from NetworkBridge	`application/ vnd.ibm.powervm.uom+xml, type=NetworkBridge`
DELETE	Delete the NetworkBridge	`application/ vnd.ibm.powervm.uom+xml, type=NetworkBridge`

Related reference:

"Managed System" on page 10
Managed System API provides a list of all system information managed by Hardware Management Console (HMC), such as system name, system machine type, model and serial number, system state, system capabilities, IP address, system migration information, system processor, system memory, and system I/O adapters information for all of the managed system. In addition, each managed system has a

list of all child objects, such as:

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

"Virtual Network" on page 26
Virtual Network is a managed system level object representing the Virtual LAN connectivity across the logical partitions.

"Virtual Switch" on page 25
PowerVM Virtual Switch APIs

"Link Aggregation"
Link Aggregation APIs for the configuring EtherChannel devices on the VIOS

Load Group

Detailed object within NetworkBridge API provides configuration for load sharing of virtual network traffic across the physical adapters in failover setup.

Resource

`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/NetworkBridge/{NetworkBridge_uuid}`

Since: Version 1.1.0

LoadGroup is a detailed element within the NetworkBridge. LoadGroup represents the pair of TrunkAdapters in a Failover and LoadBalancing enabled NetworkBridge. With a POST operation on NetworkBridge, user can add or remove the LoadGroups in a NetworkBridge. For a NonFailover NetworkBridge each LoadGroup represents one TrunkAdapter within the SEA. On a particular LoadGroup, user can add or remove virtual networks.

LoadGroup details can be viewed as a GET operation on the NetworkBridge.

PUT operation on NetworkBridge creates the corresponding LoadGroups.

POST operation on NetworkBridge allows to add, remove the load groups within the NetworkBridge. User can add or remove the associated VirtualNetworks from the LoadGroup.

Related reference:

"Network Bridge" on page 27
Network Bridge is the REST representation of the Shared Ethernet Adapters. NetworkBridge is a wrapper around the Shared Ethernet Adapter (SEA) and provides a unique way of managing the network bridge functionality for all configurations such as Failover, LoadBalancing, and Non-Failover.

"Virtual Network" on page 26
Virtual Network is a managed system level object representing the Virtual LAN connectivity across the logical partitions.

Link Aggregation

Link Aggregation APIs for the configuring EtherChannel devices on the VIOS

Resource

`rest/api/uom/VirtualIOServer/{VirtualIOServer_uuid}/LinkAggregation/{LinkAggregation_uuid}`

`rest/api/uom/VirtualIOServer/{VirtualIOServer_uuid}/LinkAggregation`

Since: Version 1.1.0

Table 31. Supported methods

Method	Description	Internet media type
GET	GET configured/specified Link Aggregation from VIOS	`application/ vnd.ibm.powervm.uom+xml, type=LinkAggregation`
PUT	Create a LinkAggregation on VIOS.	`application/ vnd.ibm.powervm.uom+xml, type=LinkAggregation`
POST	Modify LinkAggregation properties	`application/ vnd.ibm.powervm.uom+xml, type=LinkAggregation`
DELETE	Delete the LinkAggregation	`application/ vnd.ibm.powervm.uom+xml, type=LinkAggregation`

Related reference:

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

"Network Bridge" on page 27
Network Bridge is the REST representation of the Shared Ethernet Adapters. NetworkBridge is a wrapper around the Shared Ethernet Adapter (SEA) and provides a unique way of managing the network bridge functionality for all configurations such as Failover, LoadBalancing, and Non-Failover.

Shared Ethernet Adapter

Detailed object within NetworkBridge provides configuration of Shared Ethernet Adapter (SEA) properties on a VIOS.

Resource

`rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/NetworkBridge/{NetworkBridge_uuid}`

Since: Version 1.1.0

SharedEthernetAdapter is a detailed element within the NetworkBridge. The SEA attributes can be modified with a POST operation on NetworkBridge. While creating the NetworkBridge the SEA needs to be specified within the NetworkBridge with associated VirtualIOServer atom link within it. The following SEA attributes are mandatory for configuring a NetworkBridge:

1. IsPrimary - Indicates whether the SEA is primary or secondary in a failover configuration
2. AssignedVirtualIOServer - The VirtualIOServer AtomLink
3. BackingDeviceChoice.EthernetBackingDevice - The Ethernet Device used for physical network connectivity

The following properties of the SharedEthernetAdapter can be modified with a POST operation on NetworkBridge:

1. JumboFramesEnabled
2. QualityOfServiceMode
3. QueueSize
4. ThreadModeEnabled

Table 32. Supported methods

Method	Description	Internet media type
GET	SharedEthernetAdapter details can be viewed as a GET operation on the NetworkBridge.	`application/atom+xml, application/ vnd.ibm.powervm.uom+xml; type=NetworkBridge`
PUT	PUT operation on NetworkBridge creates the corresponding SharedEthernetAdapters.	`application/atom+xml, application/ vnd.ibm.powervm.uom+xml; type=NetworkBridge`
POST	POST operation on NetworkBridge allows to add (with failover enable), modify the SharedEthernetAdapters within the NetworkBridge.	`application/atom+xml, application/ vnd.ibm.powervm.uom+xml; type=NetworkBridge`

Related reference:

"Network Bridge" on page 27
Network Bridge is the REST representation of the Shared Ethernet Adapters. NetworkBridge is a wrapper around the Shared Ethernet Adapter (SEA) and provides a unique way of managing the network bridge functionality for all configurations such as Failover, LoadBalancing, and Non-Failover.

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

Template Library

Learn about the Template Library APIs that can be used to query templates that contain system or logical partition configuration information.

System Templates

REST API for querying all system templates:

`/rest/api/templates/SystemTemplate`

Table 33. Universal record locator (URL) Parameters

Param Name	Description
draft	Checks for draft templates or original templates request. **[values: true|false]**
detail	Checks whether an request for complete data or list of templates has been made. **[values: full|table]**

REST API for querying a particular system template:

`/rest/api/templates/SystemTemplate/{UUID}`

REST API for deleting a particular system template:

`/rest/api/templates/SystemTemplate/{UUID}`

REST API for adding a new template to template library:

`/rest/api/templates/SystemTemplate {template xml as payload}`

Table 34. Supported methods

Method	Description	Internet media type
GET	To retrieve the template or templates.	`application/atom+xml`
PUT	To add a new template to template library.	`application/ vnd.ibm.powervm.templates+xml;type=SystemTemplate`
DELETE	To delete a partition template.	`application/vnd.ibm.powervm.web+xml`

Partition Templates

REST API for querying all partition templates:
`/rest/api/templates/PartitionTemplate`

Table 35. URL Parameters

Param Name	Description
draft	Checks for draft templates or original templates request. **[values: true|false]**
detail	Checks whether an request for complete data or list of templates has been made. **[values: full|table]**

REST API for querying a particular partition template:
`/rest/api/templates/PartitionTemplate/{UUID}`

REST API for deleting a particular partition template:
`/rest/api/templates/PartitionTemplate/{UUID}`

REST API for adding a new template to template library:
`/rest/api/templates/PartitionTemplate {template xml as payload}`

Table 36. Supported methods

Method	Description	Internet media type
GET	To retrieve the template or templates.	`application/atom+xml`
PUT	To add a new template to template library.	`application/ vnd.ibm.powervm.templates+xml;type=PartitionTemplate`
DELETE	To delete a partition template.	`application/vnd.ibm.powervm.web+xml`

Template REST JOB API

Learn about capturing a particular system or logical partition configuration on a managed system using REST APIs.

Capture Partition configuration job

Capture a partition job that is used to capture a particular partition configuration as template.

Resource:
`/rest/api/templates/PartitionTemplate/do/capture`

Content Type:

```
application/vnd.ibm.powervm.web+xml; type=JobRequest
```

Request:

Invoke this job to capture partition configuration as template.

Table 37. Request parameters

Request Parameter Name	Description
TargetUuid	The corresponding logical partition UUID.
NewTemplateName	The name to be given for the new template.
ManagedSystemUuid	The managed system UUID.

Response:

This job returns the job id that is used to poll the job status.

Sample Job request:

```
 http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/
 xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
     <Metadata>
         <Atom/>
     </Metadata>
     <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
         <Metadata>
             <Atom/>
         </Metadata>
         <OperationName kxe="false" kb="ROR">ClearStatistics</OperationName>
         <GroupName kxe="false" kb="ROR">SRIOVFibreChannelOverEthernetLogicalPort</GroupName>
     </RequestedOperation>
     <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
         <Metadata>
             <Atom/>
         </Metadata>
     </JobParameter ns2:id="xpaimemento" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR"
 kxe="false">K_X_API_SESSION_MEMENTO</ParameterName>
     <ParameterValue kb="CUR" kxe="false">jBzF36NqAvgEqS5dH3_w5UA3x6RS2UhE-nx6NfsBAYpoo_
p9dl3ExmNNblWM6_eABoZ8r89Zy2u9VTTn0lsiPa2StdEIVzF_bgQYcRuZLX4lvEm3-8RkHSC51o9fcSMKvHBx7DO_
qKfiIbMKhYC_6c5SaoCFGmSJO3roNkn0G6tUROXgdBV2gg8cotnM1VRmnFsQmAym2E79IoE5kDGQ8w==&lt;/
ParameterValue>
   </JobParameter>
     </JobParameter ns2:id="jp11" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR" kxe="false">TargetUuid</ParameterName>
     <ParameterValue kb="CUR" kxe="false">afc0d74a-8157-3e55-803e-d6884a0f6c11</ParameterValue>
   </JobParameter>
     </JobParameter ns2:id="jp12" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR" kxe="false">NewTemplateName</ParameterName>
     <ParameterValue kb="CUR" kxe="false">syser567</ParameterValue>
   </JobParameter>
     </JobParameter ns2:id="jp14" schemaVersion="V1_2_0">
```

```
        <Metadata>
            <Atom/>
        </Metadata>
    <ParameterName kb="ROR" kxe="false">WithPhysicalIO</ParameterName>
    <ParameterValue kb="CUR" kxe="false">true</ParameterValue>
  </JobParameter>
 </JobParameters>
</JobRequest>
```

Capture system configuration job

Capture system configuration job is used to capture a particular system configuration as template.

Resource:

`/rest/api/templates/SystemTemplate/do/capture`

Content type:

`application/vnd.ibm.powervm.web+xml; type=JobRequest`

Request:

Invoke this job to capture a system configuration as a template.

Table 38. Request parameters

Request Parameter Name	Description
TargetUuid	The corresponding logical partition UUID.
NewTemplateName	The name to be given for the new template.
WithPhysicalIO	Sets to true to capture the physical I/O details.
else	Sets to false.

Response:

This job returns the job id that is used to poll the job status.

Sample Job request:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_2">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kxe="CUR" kb="fasle" ns2:base="B" ns2:id="rop" schemaVersion="V1_2">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kb="ROR" kxe="false">Capture</OperationName>
        <GroupName kb="ROR" kxe="false">SystemTemplate</GroupName>
    <ProgressType kb="ROR" kxe="false">DISCRETE</ProgressType>
    </RequestedOperation>
    <JobParameters kb="CUR" kxe="false" schemaVersion="V1_2">
        <Metadata>
            <Atom/>
        </Metadata>
    </JobParameter ns2:id="xpaimemento" schemaVersion="V1_2_0">
        <Metadata>
            <Atom/>
        </Metadata>
    <ParameterName kb="ROR"
```

```
 kxe="false">K_X_API_SESSION_MEMENTO</ParameterName>
     <ParameterValue kb="CUR" kxe="false">jBzF36NqAvgEqS5dH3_w5UA3x6RS2UhE-nx6NfsBAYpoo_
p9dl3ExmNNblWM6_eABoZ8r89Zy2u9VTTn0lsiPa2StdEIVzF_bgQYcRuZLX4lvEm3-8RkHSC51o9fcSMKvHBx7DO_
qKfiIbMKhYC_6c5SaoCFGmSJO3roNkn0G6tUROXgdBV2gg8cotnM1VRmnFsQmAym2E79IoE5kDGQ8w==&lt;/
ParameterValue>
   </JobParameter>
     </JobParameter ns2:id="jp11" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR" kxe="false">TargetUuid</ParameterName>
     <ParameterValue kb="CUR" kxe="false">afc0d74a-8157-3e55-803e-d6884a0f6c11</ParameterValue>
   </JobParameter>
     </JobParameter ns2:id="jp12" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR" kxe="false">NewTemplateName</ParameterName>
     <ParameterValue kb="CUR" kxe="false">syser567</ParameterValue>
   </JobParameter>
     </JobParameter ns2:id="jp14" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR" kxe="false">WithPhysicalIO</ParameterName>
     <ParameterValue kb="CUR" kxe="false">true</ParameterValue>
   </JobParameter>
  </JobParameters>
 </JobRequest>
```

SystemTemplateCheck job

This is the first step in the process for deploying a system. This job validates all the capabilities and configuration in a template against the system on which it is deployed. If the validation is successful, then a draft template is created and the UUID for the same is set in the response.

Resource:

`/rest/api/templates/SystemTemplate/{Template_UUID}/do/check`

Table 39. Request parameters

Request Parameter Name	Description
TargetUuid	The corresponding managed system UUID against which the template configuration is validated.
PERFORM_SYSTEM_RESET	This parameter tells the user whether to reset all the configuration on the system that the configuration is being deployed on.
K_X_API_SESSION_MEMENTO	X-API session ID for the logged in user.

Response:

This job returns the below parameters and the valid status such as Completed_OK and Failed_Before_Complete.

Table 40. Request parameters

Request Parameter Name	Description
ReadyForDeployWizard	If all the configuration in the template is correct, then the value is set to **yes**, indicating that the template is ready for deployment.
ExceptionText	This parameter contains the error messages that is thrown by Check API on wrong or invalid configuration of template.

Table 40. Request parameters (continued)

Request Parameter Name	Description
TEMPLATE_UUID	This is the daft template UUID (temporary template) which is a replica of the original template that is selected for deployment. All the further operations happens on this template.

Sample Job request:

```
  <JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
  xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_2">
     <Metadata>
         <Atom/>
     </Metadata>
     <RequestedOperation kxe="CUR" kb="fasle" ns2:base="B" ns2:id="rop" schemaVersion="V1_2">
         <Metadata>
             <Atom/>
         </Metadata>
         <OperationName kb="ROR" kxe="false">Capture</OperationName>
         <GroupName kb="ROR" kxe="false">SystemTemplate</GroupName>
     <ProgressType kb="ROR" kxe="false">DISCRETE</ProgressType>
     </RequestedOperation>
     <JobParameters kb="CUR" kxe="false" schemaVersion="V1_2">
         <Metadata>
             <Atom/>
         </Metadata>
     </JobParameter ns2:id="xpaimemento" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR"
 kxe="false">K_X_API_SESSION_MEMENTO</ParameterName>
     <ParameterValue kb="CUR" kxe="false">Y5V4xELtPfyGj1SZi3KgOl1jfMyVRKf1dozL_CF-cZ1resHhP3VsUZA
emHVT5x7eoID-nNiClIpRtl4ncJoSYoslGkrcA9u4H4poEIJyR-By91tprh8XJbxG-e9b-WLWCKGmp2FJsfwu_LKXFSb5ZIiNVyMoN
d3uqQipLRmhG6kF7zF36KcjizJL0JGoAbwymUYNqNzlrSw0PsBLf46Byw==</ParameterValue>
   </JobParameter>
     </JobParameter ns2:id="jp11" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR" kxe="false">TargetUuid</ParameterName>
     <ParameterValue kb="CUR" kxe="false">4cce8ab8-2241-384c-861e-876e1be24561</ParameterValue>
   </JobParameter>
     </JobParameter ns2:id="jp12" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR" kxe="false">PERFORM_SYSTEM_RESET</ParameterName>
     <ParameterValue kb="CUR" kxe="false">false</ParameterValue>
   </JobParameter>
  </JobParameters>
 </JobRequest>
```

SystemTemplateTransform job

Transform is the second step in the process of the system template deployment. This job fills up the required default or findable values into the template.

Resource:

```
/rest/api/templates/SystemTemplate/{DraftTemplate_UUID}/do/transform
```

Table 41. Request parameters

Request Parameter Name	Description
K_X_API_SESSION_MEMENTO	X-API session ID for the logged in user.

Response:

This job returns the valid status such as Completed_OK and Failed_Before_Complete.

Sample Job request:

```
  <JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
  xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_2_0">
     <Metadata>
         <Atom/>
     </Metadata>
     <RequestedOperation kxe="CUR" kb="fasle" ns2:base="B" ns2:id="rop" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
         <OperationName kb="ROR" kxe="false">Transform</OperationName>
         <GroupName kb="ROR" kxe="false">SystemTemplate</GroupName>
     <ProgressType kb="ROR" kxe="false">DISCRETE</ProgressType>
     </RequestedOperation>
     <JobParameters kb="CUR" kxe="false" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     </JobParameter ns2:id="xpaimemento" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR"
 kxe="false">K_X_API_SESSION_MEMENTO</ParameterName>
     <ParameterValue kb="CUR" kxe="false">Y5V4xELtPfyGj1SZi3KgOl1jfMyVRKf1dozL_CF-cZ1resHhP3VsUZA
emHVT5x7eoID-nNiClIpRtl4ncJoSYoslGkrcA9u4H4poEIJyR-By91tprh8XJbxG-e9b-WLWCKGmp2FJsfwu_LKXFSb5ZIiNVyMoN
d3uqQipLRmhG6kF7zF36KcjizJL0JGoAbwymUYNqNzlrSw0PsBLf46Byw==</ParameterValue>
   </JobParameter>
  </JobParameters>
 </JobRequest>
```

SystemTemplateDeploy job

This job initiates the deployment and the system is configured with the templates configuration.

Resource:

`/rest/api/templates/SystemTemplate/{DraftTemplate_UUID}/do/deploy`

Table 42. Request parameters

Request Parameter Name	Description
TemplateUuid	The draft UUID of the configuration template that is created after the success of the Check and Transform Job.
Stage	This Parameter defines the type of configuration that is applied on the system. Table 43 on page 38 describes the possible values.
extViosCreation	This parameter is set to **true** if the job is invoked from the Manage PowerVM. By default, it is set to **false**.
K_X_API_SESSION_MEMENTO	X-API session ID for the logged in user.

Table 43. Deployment configuration

Deployment configuration	Stage (request parameter value)
Basic system configuration	1
VIOS installation	2
VIOS license acceptance	3
Shared memory pool creation	4
Network and storage configuration Reserved storage pool creation	5

Response:

This job returns the valid status such as Completed_OK and Failed_Before_Complete.

Sample Job request:

```
  <JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
  xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_2">
     <Metadata>
         <Atom/>
     </Metadata>
     <RequestedOperation kxe="CUR" kb="fasle" ns2:base="B" ns2:id="rop" schemaVersion="V1_2">
         <Metadata>
             <Atom/>
         </Metadata>
         <OperationName kb="ROR" kxe="false">Deploy</OperationName>
         <GroupName kb="ROR" kxe="false">SystemTemplate</GroupName>
     <ProgressType kb="ROR" kxe="false">DISCRETE</ProgressType>
     </RequestedOperation>
     <JobParameters kb="CUR" kxe="false" schemaVersion="V1_2">
         <Metadata>
             <Atom/>
         </Metadata>
     </JobParameter ns2:id="xpaimemento" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR"
 kxe="false">K_X_API_SESSION_MEMENTO</ParameterName>
     <ParameterValue kb="CUR" kxe="false">Y5V4xELtPfyGj1SZi3KgOl1jfMyVRKf1dozL_CF-cZ1resHhP3VsUZA
emHVT5x7eoID-nNiClIpRtl4ncJoSYoslGkrcA9u4H4poEIJyR-By91tprh8XJbxG-e9b-WLWCKGmp2FJsfwu_LKXFSb5ZIiNVyMoN
d3uqQipLRmhG6kF7zF36KcjizJL0JGoAbwymUYNqNzlrSw0PsBLf46Byw==</ParameterValue>
   </JobParameter>
     </JobParameter ns2:id="jp11" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR" kxe="false">TemplateUuid</ParameterName>
     <ParameterValue kb="CUR" kxe="false">58288f99-1c8a-45f7-a882-2a47635045b9</ParameterValue>
   </JobParameter>
     </JobParameter ns2:id="jp12" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR" kxe="false">Stage</ParameterName>
     <ParameterValue kb="CUR" kxe="false">1</ParameterValue>
   </JobParameter>
     </JobParameter ns2:id="jp16" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR" kxe="false">extViosCreation</ParameterName>
```

```
    <ParameterValue kb="CUR" kxe="false">false</ParameterValue>
  </JobParameter>
 </JobParameters>
 </JobRequest>
```

PartitionTemplateCheck job

This job validates the configuration in the template against the system that the partition is being created on.

Resource:

`/rest/api/templates/PartitionTemplate/{Template_UUID}/do/check`

Table 44. Request parameters

Request Parameter Name	Description
TargetUuid	The UUID of managed system on which partition is created.
K_X_API_SESSION_MEMENTO	X-API session ID for the logged in user.

Table 45. Request parameters

Request Parameter Name	Description
ReadyForDeployWizard	If all the configuration in the template is correct, then this value is 'yes'. This means the template is ready for deployment.
ExceptionText	This parameter contains the error messages that is thrown by Check API on wrong or invalid Configuration of template.
TEMPLATE_UUID	This is the daft template UUID (temporary Template) which is a replica of the original template that is selected for deployment. All the further operations happens on this template.

Sample Job request:

```
  <JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
  xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_2_0">
     <Metadata>
         <Atom/>
     </Metadata>
     <RequestedOperation kxe="CUR" kb="fasle" ns2:base="B" ns2:id="rop" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
         <OperationName kb="ROR" kxe="false">Check</OperationName>
         <GroupName kb="ROR" kxe="false">PartitionTemplate</GroupName>
     <ProgressType kb="ROR" kxe="false">DISCRETE</ProgressType>
     </RequestedOperation>
     <JobParameters kb="CUR" kxe="false" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     </JobParameter ns2:id="xpaimemento" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR"
 kxe="false">K_X_API_SESSION_MEMENTO</ParameterName>
     <ParameterValue kb="CUR" kxe="false">jBzF36NqAvgEqS5dH3_w5UA3x6RS2UhE-nx6NfsBAYoXDxFZaSh3wAc
8k8Zmzh3fbQhl_AOxwK3dzxa45wUtLLuPLOcXyXNbqjEUrctES8WIGgbzQuj6JzOxNIH7KW1BsllvKuqNNTBXj4jCabO0eAqXvCClh
2yLj2YCMakkaEnyty90RI10oSpUP0TTfTjE3lKCWMgRkfZSgOy9n21h0Q==</ParameterValue>
   </JobParameter>
     </JobParameter ns2:id="jp1" schemaVersion="V1_2_0">
         <Metadata>
```

```
            <Atom/>
        </Metadata>
    <ParameterName kb="ROR" kxe="false">TemplateUuid</ParameterName>
    <ParameterValue kb="CUR" kxe="false">4cce8ab8-2241-384c-861e-876e1be24561</ParameterValue>
  </JobParameter>
  </JobParameters>
</JobRequest>
```

PartitionTemplateTransform job

Transform is the second step in the process of the partition template deployment. This job fills up the required default or findable values into the template.

Resource:

`/rest/api/templates/PartitionTemplate/{DraftTemplate_UUID}/do/transform`

Table 46. Request parameters

Request Parameter Name	Description
TargetUuid	The UUID of managed system on which partition is created.
K_X_API_SESSION_MEMENTO	X-API session ID for the logged in user.

Response:

This job returns the valid status such as Completed_OK and Failed_Before_Complete.

Sample Job request:

```
  <JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
  xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_2_0">
     <Metadata>
         <Atom/>
     </Metadata>
     <RequestedOperation kxe="CUR" kb="fasle" ns2:base="B" ns2:id="rop" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
         <OperationName kb="ROR" kxe="false">Transform</OperationName>
         <GroupName kb="ROR" kxe="false">PartitionTemplate</GroupName>
     <ProgressType kb="ROR" kxe="false">DISCRETE</ProgressType>
     </RequestedOperation>
     <JobParameters kb="CUR" kxe="false" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     </JobParameter ns2:id="xpaimemento" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR"
 kxe="false">K_X_API_SESSION_MEMENTO</ParameterName>
     <ParameterValue kb="CUR" kxe="false">jBzF36NqAvgEqS5dH3_w5UA3x6RS2UhE-nx6NfsBAYoXDxFZaSh3wAc
8k8Zmzh3fbQhl_AOxwK3dzxa45wUtLLuPLOcXyXNbqjEUrctES8WIGgbzQuj6JzOxNIH7KW1BsllvKuqNNTBXj4jCabO0eAqXvCClh
2yLj2YCMakkaEnyty90RI10oSpUP0TTfTjE3lKCWMgRkfZSgOy9n21h0Q==</ParameterValue>
   </JobParameter>
     </JobParameter ns2:id="jp1" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR" kxe="false">TemplateUuid</ParameterName>
```

```
    <ParameterValue kb="CUR" kxe="false">4cce8ab8-2241-384c-861e-876e1be24561</ParameterValue>
  </JobParameter>
  </JobParameters>
</JobRequest>
```

PartitionTemplateDeploy job

This job starts the creation of the partition and configures the properties on the system.

Resource:

`/rest/api/templates/PartitionTemplate/{DraftTemplate_UUID}/do/deploy`

Table 47. Request parameters

Request Parameter Name	Description
TargetUuid	The UUID of managed system on which partition is created.
TemplateUuid	This is the draft Ttemplate UUID (temporary template) that is a replica of the original template that is selected for deployment. All further operations happens on this template.
K_X_API_SESSION_MEMENTO	X-API session ID for the logged in user.

Response:

This job returns the valid status such as Completed_OK and Failed_Before_Complete.

Sample Job request:

```
  <JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
  xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_2_0">
     <Metadata>
         <Atom/>
     </Metadata>
     <RequestedOperation kxe="CUR" kb="fasle" ns2:base="B" ns2:id="rop" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
         <OperationName kb="ROR" kxe="false">Deploy</OperationName>
         <GroupName kb="ROR" kxe="false">PartitionTemplate</GroupName>
     <ProgressType kb="ROR" kxe="false">DISCRETE</ProgressType>
     </RequestedOperation>
     <JobParameters kb="CUR" kxe="false" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     </JobParameter ns2:id="xpaimemento" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR"
 kxe="false">K_X_API_SESSION_MEMENTO</ParameterName>
     <ParameterValue kb="CUR" kxe="false">jBzF36NqAvgEqS5dH3_w5UA3x6RS2UhE-nx6NfsBAYoXDxFZaSh3wAc
8k8Zmzh3fbQhl_AOxwK3dzxa45wUtLLuPLOcXyXNbqjEUrctES8WIGgbzQuj6JzOxNIH7KW1BsllvKuqNNTBXj4jCabO0eAqXvCClh
2yLj2YCMakkaEnyty90RI10oSpUP0TTfTjE3lKCWMgRkfZSgOy9n21h0Q==</ParameterValue>
   </JobParameter>
     </JobParameter ns2:id="jp11" schemaVersion="V1_2_0">
         <Metadata>
             <Atom/>
         </Metadata>
     <ParameterName kb="ROR" kxe="false">TemplateUuid</ParameterName>
```

```
    <ParameterValue kb="CUR" kxe="false">398548a9-4fc4-484c-a50c-b0e6cd9702e6</ParameterValue>
  </JobParameter>
  </JobParameters>
</JobRequest>
```

SR-IOV

This section describes the Single Root I/O Virtualization (SR-IOV) configuration APIs.

SR-IOV provides the I/O virtualization within the managed system. This section provides the required APIs for configuring the SR-IOV adapters, physical ports and logical ports.

Related concepts:

"Virtual Storage Management" on page 47
Virtual Storage Management provides end-to-end view of the storage assigned to client logical partition.

Related reference:

"Managed System" on page 10
Managed System API provides a list of all system information managed by Hardware Management Console (HMC), such as system name, system machine type, model and serial number, system state, system capabilities, IP address, system migration information, system processor, system memory, and system I/O adapters information for all of the managed system. In addition, each managed system has a list of all child objects, such as:

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

SR-IOV Adapter

Single Root I/O Virtualization Adapter provides I/O virtualization capabilities for the managed system. The APIs provide the configuration of SR-IOV adapters.

Resource

`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}`

Since: Version 1.1.0

SRIOVAdapter is a detailed element within the ManagedSystem. The adapter details are retrieved by executing GET operation on the ManagedSystem. The adapter mode can be transformed from Dedicated to SR-IOV and from SR-IOV to Dedicated with a Job operation.

Supported methods

This is a detailed object. GET, PUT, POST, DELETE operations are not supported directly. Users can perform the operation by using ManagedSystem REST URI Get Operations.

Related reference:

"ModifySRIOVAdapterMode_ManagedSystem Job" on page 67
Operation to modify the state of SRIOV Adapter of a Managed System.

"SR-IOV Ethernet Physical Port" on page 43
Ethernet ports present on the SR-IOV adapter.

"SR-IOV Converged Ethernet Physical Port" on page 44
Converged Ethernet ports present on the SR-IOV adapter.

"SR-IOV Ethernet Logical Port" on page 43
REST APIs for configuring SR-IOV Ethernet Logical Ports on LogicalParition and VirtualIOServer.

SR-IOV Ethernet Logical Port

REST APIs for configuring SR-IOV Ethernet Logical Ports on LogicalParition and VirtualIOServer.

Resource

```
/rest/api/uom/LogicalPartition/{LogicalPartition_uuid}/SRIOVEthernetLogicalPort/
{SRIOVEthernetLogicalPort_uuid}
```

```
/rest/api/uom/VirtualIOServer/{VirtualIOServer_uuid}/SRIOVEthernetLogicalPort/
{SRIOVEthernetLogicalPort_uuid}
```

Since: Version 1_1_0

Table 48. Supported methods

Method	Description	Internet media type
GET	Get the configured SR-IOV Ethernet Logical Ports on the LogicalPartition and VIOS.	`application/ vnd.ibm.powervm.uom+xml; type=SRIOVEthernetLogicalPort`
PUT	Create an Ethernet Logical Port on VIOS and LogicalPartition.	`application/ vnd.ibm.powervm.uom+xml; type=SRIOVEthernetLogicalPort`
POST	Modify the Ethernet Logical Port properties.	`application/ vnd.ibm.powervm.uom+xml; type=SRIOVEthernetLogicalPort`
DELETE	Delete the Ethernet Logical Port.	`application/ vnd.ibm.powervm.uom+xml; type=SRIOVEthernetLogicalPort`

Related reference:

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

"SR-IOV Adapter" on page 42
Single Root I/O Virtualization Adapter provides I/O virtualization capabilities for the managed system. The APIs provide the configuration of SR-IOV adapters.

"SR-IOV Ethernet Physical Port"
Ethernet ports present on the SR-IOV adapter.

"SR-IOV Converged Ethernet Physical Port" on page 44
Converged Ethernet ports present on the SR-IOV adapter.

SR-IOV Ethernet Physical Port

Ethernet ports present on the SR-IOV adapter.

Resource

```
/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}
```

Since: Version 1.1.0

SRIOVEthernetPhysicaPort is a detailed element within the SRIOVAdapter. The physical ports are retrieved by executing GET operation on the ManagedSystem (associated within the SRIOVAdapter).

SRIOVEthernetPhysicalPort properties are configured with a POST operation on the ManagedSystem.

Supported methods Physical port details are retrieved with GET operation on the ManagedSystem contained within the SRIOVAdapter.

Modify the physical port properties with a POST operation on the containing ManagedSystem.

Related reference:

"Managed System" on page 10
Managed System API provides a list of all system information managed by Hardware Management Console (HMC), such as system name, system machine type, model and serial number, system state, system capabilities, IP address, system migration information, system processor, system memory, and system I/O adapters information for all of the managed system. In addition, each managed system has a list of all child objects, such as:

"SR-IOV Adapter" on page 42
Single Root I/O Virtualization Adapter provides I/O virtualization capabilities for the managed system. The APIs provide the configuration of SR-IOV adapters.

"SR-IOV Converged Ethernet Physical Port"
Converged Ethernet ports present on the SR-IOV adapter.

"SR-IOV Ethernet Logical Port" on page 43
REST APIs for configuring SR-IOV Ethernet Logical Ports on LogicalParition and VirtualIOServer.

SR-IOV Converged Ethernet Physical Port

Converged Ethernet ports present on the SR-IOV adapter.

Resource

```
/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}
```

Since: Version 1.1.0

SRIOVConvergedNetworkAdapterPhysicalPort is a detailed element within the SRIOVAdapter. The Physical ports are retrieved by executing GET operation on the ManagedSystem (associated within the SRIOVAdapter).

SRIOVConvergedNetworkAdapterPhysicalPort properties are configured with a POST operation on the ManagedSystem.

Supported methods Physical port details are retrieved with GET operation on the ManagedSystem contained within the SRIOVAdapter.

Modify the physical port properties with a POST operation on the containing ManagedSystem.

Related reference:

"Managed System" on page 10
Managed System API provides a list of all system information managed by Hardware Management Console (HMC), such as system name, system machine type, model and serial number, system state, system capabilities, IP address, system migration information, system processor, system memory, and system I/O adapters information for all of the managed system. In addition, each managed system has a list of all child objects, such as:

"SR-IOV Adapter" on page 42
Single Root I/O Virtualization Adapter provides I/O virtualization capabilities for the managed system. The APIs provide the configuration of SR-IOV adapters.

"SR-IOV Ethernet Physical Port" on page 43
Ethernet ports present on the SR-IOV adapter.

"SR-IOV Ethernet Logical Port" on page 43
REST APIs for configuring SR-IOV Ethernet Logical Ports on LogicalParition and VirtualIOServer.

Host Ethernet Adapter

Host Ethernet Adapter (HEA) which is a physical Ethernet adapter that is integrated directly into the system (GX+ bus) on a managed system. HEAs are also known as Integrated Virtual Ethernet adapters (IVE adapters).

HEAs offer high throughput, low latency, and virtualization support for Ethernet connections. HEAs have the same uses as other types of physical Ethernet adapters. For example, you can use an HEA to establish a console connection to a logical partition.

Unlike most other types of I/O devices, you can not assign the HEA to a logical partition. Instead, multiple logical partitions can connect directly to the HEA and use the HEA resources. This allows the logical partitions to access external networks through the HEA without having to go through an Ethernet bridge on another logical partition.

Related reference:

"Managed System" on page 10
Managed System API provides a list of all system information managed by Hardware Management Console (HMC), such as system name, system machine type, model and serial number, system state, system capabilities, IP address, system migration information, system processor, system memory, and system I/O adapters information for all of the managed system. In addition, each managed system has a list of all child objects, such as:

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

Host Ethernet Adapter

This API provides the details of the port configuration of the physical Host Ethernet Adapters on the managed system, and also allows to change the configuration of any of the ports on an HEA. Host Ethernet Adapter is child object of ManagedSystem.

Resource

`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/HostEthernetAdapter`

This API provides all the HostEthernetAdapters information for the given ManagedSystem

`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/HostEthernetAdapter/{HostEthernetAdapter_uuid}`

This API provides the HostEthernetAdapters information of the given instance for a ManagedSystem

Since: Version 1.1.0

Resource

`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/HostEthernetAdapter/quick/{Property name}`

`/rest/api/uom/ManagedSystem/{ManagedSystem_uuid}/HostEthernetAdapter/{HostEthernetAdapter_uuid}/quick/{Property name}`

Since: Version 1.1.0

Table 49. Quick property

Property name	Description
AdapterState	Fetch the state of the HostEthernetAdapter

Table 50. Supported methods

Method	Description	Internet media type
GET	It retrieves information identified by the Request-URI. Produced data shall be returned as the entity in the response. The list of attributes produced as part of feed is found at HostEthernetAdapter_xsd.html	`application/ vnd.ibm.powervm.uom+xml; Type=HostEthernetAdapter`
POST	It is used to request the web server to accept the entity enclosed in the request and allow them to change state of the entity. The list of attributes which allow to change is found at HostEthernetAdapter_xsd.html	`application/ vnd.ibm.powervm.uom+xml; Type=HostEthernetAdapter`

Host Ethernet Adapter Logical Port

An Host Ethernet Adapter Logical Port is a representation of a physical HostEthernetAdapter on a logical partition. Virtual Local Area Networks (VLANs) in which the selected logical port can participate. This allows you to connect VLANs on your managed system to external networks through a Host Ethernet Adapter.

The LogicalPartition/VirtualIOServer will be able to connect to the physical port on the Host Ethernet Adapter using the logical port that you select.

Host Ethernet Adapter Logical port is detail object of Logical partition and VirtualIOServer.

Host Ethernet Adapter Logical Port information can be obtained as part of GET LogicalPartition and VirtualIOServer. Operations on Host Ethernet Adapter Logical Port are done as part of PUT/POST LogicalPartition and VirtualIOServer. The different operations which can be done on Host Ethernet Adapter Logical Port using LogicalPartition and VirtualIOServer API are as follows:

- GET Host Ethernet Adapter Logical Port
- PUT Host Ethernet Adapter Logical Port
- POST Host Ethernet Adapter Logical Port
- DELETE Host Ethernet Adapter Logical Port

Since: Version 1.1.0

Host Ethernet Adapter Physical Port

Host Ethernet Adpater Physical Port is detail object of HostEthernetAdater. Configure the required settings for Host Ethernet Adapter Physical port using HostEthernetAdapter API.

Host Ethernet Adapter Physical Port information can be obtained as part of GET HostEthernetAdapter. Operations on Host Ethernet Adapter Physical Port are done as part of POST HostEthernetAdapter. The different operations that can be done on Host Ethernet Adapter Physical Port using HostEthernetAdapter API are as follows:

- GET Host Ethernet Adapter physical port
- POST Host Ethernet Adapter physical port

Since: Version 1.1.0

Virtual Storage Management

Virtual Storage Management provides end-to-end view of the storage assigned to client logical partition.

The detailed objects used for this purpose are Virtual SCSI Mapping and Virtual FC Mapping. Virtual SCSI Mapping provides end-to-end mapping between client and server adapters. This contains the following attributes: associatedLogicalPartition, clientAdapter, serverAdapter, storage, targetDevice. Virtual FC Mapping provides end-to-end mapping between client and server adapters. This contains the following attributes: associatedLogicalPartition, clientAdapter, serverAdapter, fibre channel port.

Related concepts:

"Virtual Network Management" on page 25
This section provides the description and APIs for virtual network management operations that can be performed within a managed system.

Related reference:

"Managed System" on page 10
Managed System API provides a list of all system information managed by Hardware Management Console (HMC), such as system name, system machine type, model and serial number, system state, system capabilities, IP address, system migration information, system processor, system memory, and system I/O adapters information for all of the managed system. In addition, each managed system has a list of all child objects, such as:

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

Physical Volume

A physical volume is a collection of regions on one or more disks. Data is stored in physical volumes and administrative operations are performed on physical volumes.

Physical Volume information can be obtained as part of GET VirtualIOServer.

Operations on Physical Volume are done as part of POST VirtualIOServer. This is a detailed object of VirtualIOServer Object

The different operations which can be done on Physical Volume are as follows:

- Add / Change Reserve Policy to Physical Volume.

Since: Version 1.1.0

Related reference:

"Volume Group" on page 50
Volume Group is a storage pool created from one or more physical volumes.

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

Virtual Disk

A Virtual Disk is a portion of a physical volume. It is also known as Logical Volume.

Within each volume group, one or more logical volumes are defined. Logical volumes are groups of information located on physical volumes. Data on logical volumes appears to the user to be contiguous but can be discontiguous on the physical volume.

Virtual Disk is a detailed object and can be obtained as part of GET Volume Group. Virtual Disks inside Volume Group rootvg will not be listed. Operations on Virtual Disk are done as part of POST Volume Group. The different operations which can be done on Virtual Disk are as follows:

- Create Virtual Disk
- Delete Virtual Disk
- Extend Virtual Disk

Since: Version 1.1.0

Related reference:

"Volume Group" on page 50
Volume Group is a storage pool created from one or more physical volumes.

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

Virtual Media Repository

The Virtual Media Repository provides a single container to store and manage file-backed Virtual Optical Media files.

Media stored in the repository can be loaded into file-backed virtual optical devices for exporting to client partitions. Only one repository can be created within a Virtual I/O Server.

Virtual Media Repository gives the following information:

- Name of virtual media repository, which is always VMLibrary
- Virtual media repository size
- Virtual optical media disks in the virtual media repository

Virtual Media Repository is a detailed object and can be obtained as part of GET VirtualIOServer as well as GET VolumeGroup. Operations on Virtual Media Repository are done as part of POST Volume Group. The different operations that can be done on Virtual Media Repository are as follows:

- Create Media Repository
- Delete Media Repository
- Change Media Repository size
- Create Virtual Optical Media : Only Blank Optical Media can be created as part of POST Volume Group
- Delete Virtual Optical Media

Since: Version 1.1.0

Related reference:

"Virtual Optical Media" on page 49
Any optical device equipped on the Virtual I/O Server partition (either CD-ROM, DVD-ROM, or DVD-RAM) can be virtualized and assigned at any logical partition.

"Volume Group" on page 50
Volume Group is a storage pool created from one or more physical volumes.

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

Virtual Optical Media

Any optical device equipped on the Virtual I/O Server partition (either CD-ROM, DVD-ROM, or DVD-RAM) can be virtualized and assigned at any logical partition.

Virtual Optical Media is a detailed object. Virtual Optical Media is information can be obtained in GET VirtualIOServer or GET Volume Group inside Virtual Media Repository.

Operations on Virtual Optical Media are done as part of POST Volume Group. The different operations which can be done on Virtual Optical Media are as follows:

- Create Virtual Optical Media
- Delete Virtual Optical Media

Since: Version 1.1.0

Related reference:

"Virtual Media Repository" on page 48
The Virtual Media Repository provides a single container to store and manage file-backed Virtual Optical Media files.

"Volume Group" on page 50
Volume Group is a storage pool created from one or more physical volumes.

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

Virtual Fibre Channel Mapping

Virtual FC Mapping provides end-to-end mapping between client and server adapter and its association with fibre channel port.

The clientAdapter and serverAdapter are optional. If not provided, then the client and server adapters are created by using the next available free virtual slot.

Port needs to be provided that indicates the Fibrechannel Port to be used to communicate with the storage.

While creating the newer fibre channel objects the client and server adapters are optional.

This is a detailed object of VirtualIOServer Object

Resource

`/rest/api/uom/VirtualIOServer/{vios_uuid}`

Supported methods

This is the detailed object. Hence GET, PUT, POST, DELETE operations are not supported directly. Getting FC mapping is done through GET of VIOS API and POST is done through POST of VIOS API.

Since: Version 1.1.0

Related reference:

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical

partitions within the server.

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

Virtual SCSI Mapping

Virtual SCSI Mapping provides end-to-end mapping between client and server adapter and its association with storage.

The clientAdapter and serverAdapter are optional. If not provided, the client and server adapters are created by using the next available free slot IDs.

The storage provided by the user could be one of the following:

- LogicalUnit
- PhysicalVolume
- VirtualDisk
- VirtualOpticalMedia

Physical optical device is not supported through this URI.

The targetDevice provided by the user could be one of the following:

- LogicalVolumeVirtualTargetDevice
- PhysicalVolumeVirtualTargetDevice
- SharedStoragePoolLogicalUnitVirtualTargetDevice
- VirtualOpticalTargetDevice

The targetDevice is an optional attribute. If this attribute is not provided, the Virtual IO Server creates corresponding target device for the storage provided.

This is a detailed object of VirtualIOServer Object.

Resource

```
/rest/api/uom/VirtualIOServer/{VirtualIOServer_uuid}
```

Supported methods

This is the detailed object. Hence GET, PUT, POST, DELETE operations are not supported directly. Getting SCSI mapping is done through GET of VirtualIOServer API and POST is done through POST of VirtualIOServer API.

Since: Version 1.1.0

Related reference:

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

Volume Group

Volume Group is a storage pool created from one or more physical volumes.

Resource

```
/rest/api/uom/VirtualIOServer/{VirtualIOServer_uuid}/VolumeGroup
```

This API provides information about all the VolumeGroups in the VirtualIOServer.
`/rest/api/uom/VirtualIOServer/{VirtualIOServer_uuid}/VolumeGroup/{VolumeGroup_uuid}`

This API provides information about a particular VolumeGroup in the VirtualIOServer given its UUID.

When you install a VirtualIOServer, one volume group (the root volume group, called rootvg) is automatically created that contains the base set of logical volumes required to start the system, and any other logical volumes you specify to the installation script. The rootvg includes paging space, the journal log, boot data, and dump storage, each in its own separate logical volume. No operations can be performed on rootvg. Virtual Disks inside rootvg are not be listed.

The REST API for volume Group provides the following information:
- Free Space in the Volume Group in MBytes
- Media Repository in Volume Group, if any
- Physical Volumes from which Volume Group is created
- Virtual Disks created in the Volume Group

Table 51. Supported methods

Method	Description	Internet media type
GET	Gets details of the volume group like physical volume, virtual disks and virtual media repository.	`application/ vnd.ibm.powervm.uom+xml; type=VolumeGroup`
PUT	Creates a Volume Group	`application/ vnd.ibm.powervm.uom+xml; type=VolumeGroup`
POST	The following operations on Volume Group can be done in POST: • Extend Volume Group - Add one or more physical volumes to the Volume Group • Reduce Volume Group - Remove one or more physical volumes from the Volume Group • Create Media Repository • Delete Media Repository • Change Media Repository size • Create Virtual Optical Media • Delete Virtual Optical Media • Extend Virtual Disk • Create Virtual Disk • Delete Virtual Disk	`application/ vnd.ibm.powervm.uom+xml; type=VolumeGroup`
DELETE	Deletes the Volume Group.	

Since: Version 1.1.0

Related reference:

"Physical Volume" on page 47
A physical volume is a collection of regions on one or more disks. Data is stored in physical volumes and administrative operations are performed on physical volumes.

"Virtual Media Repository" on page 48
The Virtual Media Repository provides a single container to store and manage file-backed Virtual Optical Media files.

"Virtual Disk" on page 47
A Virtual Disk is a portion of a physical volume. It is also known as Logical Volume.
"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.
"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

Cluster

A Cluster is a set of one or more networked Virtual I/O Server (VIOS) partitions, where each VIOS within the cluster has access to a common set of physical volumes.

Cluster provides a distributed storage repository for Power systems. The Cluster is formed by shared storage pool which is a collection of physical volumes that can be accessed by every VIOS that is a part of the Cluster. A VIOS that is a part of the Cluster is called as a Node in the Cluster. All Nodes in the Cluster can read, write, and access data on physical volumes that are part of the shared storage pool.

The cluster repository disk is used as the central repository for the cluster configuration data. The minimal size of the repository is dependent on the cluster configuration.

The cluster capabilities attribute specifies whether the cluster is Tier capable and Tier Mirror capable

The major attributes in the Cluster are :
- ClusterName : User defined name for Cluster
- ClusterID : VIOS generated ID for Cluster
- RepositoryDisk : Physical volume used for book keeping
- ClusterSharedStoragePool : Atom-link generated for associated Shared Storage Pool created for Cluster
- Node : Collection of VIOS that are part of Cluster
- ClusterCapabilities : Capabilities to show if Cluster is Tier & Tier Mirror Capable

Resource

`/rest/api/uom/Cluster/{Cluster_uuid}`

Table 52. Supported methods

Method	Description	Internet media type
GET	Fetch the cluster information from all managed VIOS.	`application/atom+xml, application/ vnd.ibm.powervm.uom+xml; type=Cluster`
POST	Update the existing Cluster such as : • Add Nodes to Cluster. • Remove Nodes from Cluster. • Replace Repository disk in the Cluster.	`application/atom+xml, application/ vnd.ibm.powervm.uom+xml; type=Cluster`

Quick Properties

User can fetch the only specified property.

Resource

`/rest/api/uom/Cluster/quick/All`

Supported property names available at /rest/api/uom/Cluster/quick/All URI.

Table 53. List of Quick properties

Property Name	Description
ClusterName	User defined name for Cluster.
ClusterID	VIOS generated ID for Cluster.
UUID	UUID of Cluster.

Resource

```
/rest/api/uom/Cluster/{Cluster_uuid}/quick
```

Supported property names available at /rest/api/uom/Cluster/{Cluster_uuid}/quick URI.

Table 54. List of Quick properties

Property Name	Description
ClusterName	User defined name for Cluster.
ClusterID	VIOS generated ID for Cluster.

Resource

```
/rest/api/uom/Cluster/{Cluster_uuid}/quick/{Property Name}
```

Supported property names available at /rest/api/uom/Cluster/{Cluster_uuid}/quick/{Property Name} URI.

Table 55. List of Quick properties

Property Name	Description
ClusterName	User defined name for Cluster.
ClusterID	VIOS generated ID for Cluster.

Note :

The Cluster does not support PUT and DELETE operation, kindly use below Jobs to create and delete Cluster.

- "Create_Cluster Job" on page 92
- "Delete_Cluster Job" on page 99

Replacing repository disk of the Cluster is also supported by Job

- "Cluster_Replace_Repository_Disk Job" on page 106

Since: Version 1_1_0

Related tasks:

"Create_Cluster Job" on page 92
The cluster create job is used to create a cluster with shared storage pool. The cluster and shared storage pool details are available in an XML format.

Related reference:

"Delete_Cluster Job" on page 99
Delete_Cluster Job is used to delete a cluster with Shared Storage Pool (SSP).

Shared Storage Pool

Shared storage pool contains a logical organization of one or more physical volumes that will be used to provide block storage.

The shared storage pool is composed of collection of physical volumes that can be accessed by every Virtual I/O Server (VIOS) that is part of Cluster to which shared storage pool belongs. Shared storage pools extend storage virtualization to multiple VIOS on multiple Power servers. This storage is used to create logical units (LUs) that can be mapped to logical partitions as storage.

The attributes in shared storage pool are :

- StoragePoolName : User defined name for the shared storage pool.
- UniqueDeviceID : VIOS generated ID for the shared storage pool.
- Capacity : Total capacity of the shared storage pool in GB.
- FreeSpace : Total free space left out in the shared storage pool in GB.
- OverCommitSpace : Amount of space overcommited by the shared storage pool to provide thin provisioning.
- AlertThreshold : Specifies alert set on total percentage of capacity being used by the shared storage pool.
- PhysicalVolumes : Collection of physical volumes that are part of the shared storage pool.
- LogicalUnits : Collection of logical volumes that are created from the shared storage pool.
- MultiDataTierConfigured : Specifies if multi-data tier is configured for the shared storage pool.
- MultiFailureGroupConfigured : Specifies if multi-failure group configured for the shared storage pool.

Resource

`/rest/api/uom/SharedStoragePool/{SharedStoragePool_uuid}`

Table 56. Supported methods

Method	Description	Internet media type
GET	Fetch the shared storage pool information from all managed VIOS.	`application/atom+xml, application/vnd.ibm.powervm.uom+xml; type=SharedStoragePool`
POST	Update the existing shared storage pool such as : • Add physical volumes to the shared storage pool. • Replace physical volumes from shared stoarge pool. • Create logical units. • Delete logical units. • Set alert for threshold at shared storage pool level.	`application/atom+xml, application/vnd.ibm.powervm.uom+xml; type=SharedStoragePool`

Resource

`/rest/api/uom/SharedStoragePool/quickAll`

All the property names are listed at /rest/api/uom/SharedStoragePool/quickAll URI.

Resource

`/rest/api/uom/SharedStoragePool/{SharedStoragePool_uuid}/quick/{Property Name}`

Supported property names available at /rest/api/uom/SharedStoragePool/{SharedStoragePool_uuid}/quick/{Property Name} URI.

Table 57. List of Quick properties

Property Name	Description
StoragePoolName	User defined name for the shared storage pool.
UniqueDeviceID	UDID of shared storage pool.
Capacity	Capacity of shared storage pool.
FreeSpace	Free space available in shared storage pool.
UUID	UUID of shared storage pool.
AlertThreshold	Alert threshold of shared storage pool.

Note :

The shared storage pool is created by default when a cluster is created and deleted when a cluster is deleted.

Since: Version 1_1_0

Related tasks:

"Logical Unit" on page 58
Logical unit is a file in the pool that provides file-backed storage for Power systems.

Related reference:

"CreateLogicalUnit_Cluster Job" on page 99
CreateLogicalUnit_Cluster Job is used to create a logical unit (LU) in a cluster or Shared Storage Pool (SSP).

"LULinkedClone_Cluster Job" on page 104
Logical Unit Clone job is used to create a clone of link clone of logical units.

Tier

Tier is collection of physical disk that have similar performance characteristics.

Data isolation is an important requirement for segregating storage based on the workloads. Grouping of physical volumes having similar characteristics is critical to provide data isolation. Virtual I/O Server (VIOS) allows data isolation for shared storage pool by grouping the physical disks of a tier. There can be multiple tiers in the same shared storage pool thus providing data isolation based on the requirement of workload.

The other requirement is data mirroring. If applications use logical units from the shared storage pool in a data critical system need data mirroring. This can be achieved by enabling the failure group at the tier level. The VIOS allows physical disks to be associated with a failure group and assign it to a tier. So, you can mirror only the data that is critical.

The major attributes in Tier are :

- Name : User defined name for tier.
- UniqueDeviceID : VIOS generated ID for tier.
- Type : Specifies it tier is System or User tier.
- IsDefault : Specifies if tier is a default tier.
- Capacity : Total capacity of Tier in GB.
- FreeSpace : Total free space left out in Tier in GB.
- OverCommitSpace : Amount of space over-committed by the tier to provide thin provisioning.

- FreeSpaceThreshold : Specifies alert set on total percentage of frees pace available in tier.
- OverCommitSpaceThreshold : Specifies alert set on total percentage of space over-commited by tier.
- TotalLogicalUnitSize : Total size of logical units created under tier
- MirrorState : Specifies if mirrioring is enabled on tier
- FailureGroups : Collection of failgure configured for tier
- FailureGroup Name: Specifies the user defined name of failure group
- FailureGroup UniqueDeviceID: VIOS generated ID for tier.
- FailureGroup Capacity : Total capacity of FailureGroup in GB.
- FailureGroup State : Specifies the state of failure group
- PhysicalVolumes : Collection of Physical Volume that are part of Shared storage pool.
- AssociatedLogicalUnits : Collection of Logical units that are carved out of tier.
- AssociatedSharedStoragePool : Specifies the Shared storage pool to which tier belong

Resource

```
/rest/api/uom/Tier
```

This API provides information about the tiers that are a part of the shared storage pool managed by the hardware management console (HMC).

```
/rest/api/uom/Tier/{Tier_uuid}
```

This API provides information about a particular tier given its UUID.

```
/rest/api/uom/SharedStoragePool/SharedStoragePool_uuid/Tier
```

This API provides information about the tiers that are a part of the specified shared storage pool.

```
/rest/api/uom/SharedStoragePool/SharedStoragePool_uuid/Tier/{Tier_uuid}
```

This API provides information about a particular tier given its UUID, which is present in the specified shared storage pool.

Ignore error or Partial response

This option ignores the error generated from bad clusters and populates the tier information from the healthy clusters by populating **partialresponseexception** attribute at the feed level.

Resource

```
/rest/api/uom/Tier?ignoreError=true
```

This option ignores the error from bad clusters and populates the tier information from the other healthy clusters by populating the **partialresponseexception** attribute at feed level.

```
/rest/api/uom/SharedStoragePool/SharedStoragePool_uuid/Tier?ignoreError=true
```

This option ignores the error from the bad clusters and populates the tier information only from the specified healthy cluster by populating the **partialresponseexception** attribute at feed level.

Quick properties

You can fetch the only the specified property.

Resource

```
/rest/api/uom/Tier/{Tier_uuid}/quick/{Property name}
```

Supported property names are available at /rest/api/uom/Tier/quick URI.

Table 58. Quick properties

Quick property	Description
Type	The tier type as an enumeration key. For example, RestrictedSystemTier, UnrestrictedSystemTier, UserTier, or Unknown).
Capacity	The capacity of the tier.
UniqueDeviceId	The unique device ID of the Tier.
MirrorState	The tier MirrorState as an enumeration key. For example, AIX_LINUX or OS_400.
Name	The name of the tier.

Extended groups

Some attribute make calls to the underlying layer to get the data. These attributes are not cached and populated in REST API unless you specifically requested for them.

Resource

```
/rest/api/uom/Tier?group={Group Name}
```

This API provides the tier information about the specified extended group attributes.

```
/rest/api/uom/Tier/{Tier_uuid}?group={Group Name}
```

This API provides information about the specified extended group attributes of the given tier instance.

```
/rest/api/uom/SharedStoragePool/{SharedStoragePool_uuid}/Tier?group={Group Name}
```

This API provides information about the specified extended group attributes of the given instance.

```
/rest/api/uom/SharedStoragePool/{SharedStoragePool_uuid}/Tier/{Tier_uuid}?group={Group Name}
```

This API provides information about the specified extended group attributes of the given instance.

Table 59. List of extended groups

Group name	Description
Advanced	This extended group call populates FreeSpace, OverCommitSpace, and TotalLogicalUnitSize value of a tier.
TierThreshold	This extended group call populates FreeSpaceThreshold and OverCommitSpaceThreshold value of a tier.
None	This is an extended group that provides the default attributes.

Table 60. Supported methods

Method	Description	Internet media type
GET	Fetch the tier information from all managed VIOS.	`application/atom+xml`, `application/vnd.ibm.powervm.uom+xml; type=Tier`
PUT	Create a new tier in a specified Shared storage pool.	`application/atom+xml`, `application/vnd.ibm.powervm.uom+xml; type=Tier`

Table 60. Supported methods (continued)

Method	Description	Internet media type
POST	Update the existing tier. You can do any of the following actions: • Modify tier name. • Modify tier type. • Set tier as default tier. • Modify FreeSpaceThreshold of Tier. • Modify OverCommitSpaceThreshold of Tier. • Add new failure group to Tier. • Modify failure group name in a Tier. • Add Physcial Volumes to failure group. • Remove Physical Volumes from failure group. • Replace Physical Volumes in failure group. • Remove failure group from a Tier.	`application/atom+xml, application/vnd.ibm.powervm.uom+xml; type=Tier`
DELETE	Delete the specified tier.	`application/atom+xml, application/vnd.ibm.powervm.uom+xml; type=Tier`

Note :

There exists a default tier of SYSTEM type which gets created when cluster is created without specifying multiple tier.

Since: Version 1_3_0

Related tasks:

"Logical Unit"
Logical unit is a file in the pool that provides file-backed storage for Power systems.

Related reference:

"CreateLogicalUnit_Cluster Job" on page 99
CreateLogicalUnit_Cluster Job is used to create a logical unit (LU) in a cluster or Shared Storage Pool (SSP).

"LULinkedClone_Cluster Job" on page 104
Logical Unit Clone job is used to create a clone of link clone of logical units.

Logical Unit

Logical unit is a file in the pool that provides file-backed storage for Power systems.

Logical volumes are created from the shared storage pool to which they belongs. Logical units can be of different types, but all are files in the pool. The hardware management console (HMC) supports the following types of logical units :

- VirtualIO_Disk : A disk logical unit simulates a disk and is mapped to a client partition.
- VirtualIO_Image : An Image logical unit is a part of image management and can be used to rapidly deploy client logical partitions. An image logical unit contains the operating system image. Image logical units can not directly be mapped to a client logical partition.

A logical unit can be created as either with a THICK or a THIN provisioning capability.

- THIN : THIN logical units might present a much larger image than the current or actual physical use. These logical units are not fully backed by physical storage if the blocks are not in actual use.
- THICK : THICK logical units do not allow to commit beyond the available physical storage. These logical units are fully backed by physical storage, irrespective of the actual usage.

Logical units also support cloning. The HMC provides option to perform linked clone of logical units.

If the **ClonedFrom** attribute is mentioned when the logical unit is created, then the newly created logical unit will also be cloned from the specified source logical unit.

List of jobs supported for Logical unit.

- "CreateLogicalUnit_Cluster Job" on page 99
- "DeleteLogicalUnit_Cluster Job" on page 101
- "LULinkedClone_Cluster Job" on page 104
- "MigrateLogicalUnit_Cluster Job" on page 103
- "GrowLogicalUnit_Cluster Job"

Since: Version 1_1_0

Related tasks:

"Shared Storage Pool" on page 54
Shared storage pool contains a logical organization of one or more physical volumes that will be used to provide block storage.

"DeleteLogicalUnit_Cluster Job" on page 101
The DeleteLogicalUnit_Cluster job is used to remove or delete a logical unit from the cluster or a shared storage pool.

"MigrateLogicalUnit_Cluster Job" on page 103
The MigrateLogicalUnit_Cluster job is used to move a logical unit from one tier to another tier within the same cluster.

"GrowLogicalUnit_Cluster Job"
The GrowLogicalUnit job is used to increase the size of the logical unit in a cluster or shared storage pool.

Related reference:

"CreateLogicalUnit_Cluster Job" on page 99
CreateLogicalUnit_Cluster Job is used to create a logical unit (LU) in a cluster or Shared Storage Pool (SSP).

"LULinkedClone_Cluster Job" on page 104
Logical Unit Clone job is used to create a clone of link clone of logical units.

GrowLogicalUnit_Cluster Job

The GrowLogicalUnit job is used to increase the size of the logical unit in a cluster or shared storage pool.

Resource

```
/rest/api/uom/Cluster/{Cluster_UUID}/do/GrowLogicalUnit
```

Request

The details of the logical unit such as the unique device identifier (UDID) and the new capacity are required.

Table 61. Request parameters

Request parameter	Description
LogicalUnitUDID	The value of UDID of the logical unit. This value is mandatory.
Capacity	The capacity to which the size of the logical unit must be increased.

Response

This job returns the UDID and capacity of the logical unit, and the valid job status values such as `Complete_Ok` and `Failed_Before_Complete`.

Table 62. Response parameters

Response parameter	Description
LogicalUnitUDID	UDID of the logical unit
Capacity	Updated logical unit size.

Sample Job Request(s)

```
 <<JobRequest xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"

xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">

    <Metadata>

        <Atom/>

    </Metadata>

    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">

        <Metadata>

            <Atom/>

        </Metadata>

        <OperationName kxe="false" kb="ROR">GrowLogicalUnit</OperationName>

        <GroupName kxe="false" kb="ROR">Cluster</GroupName>

        <ProgressType kxe="false" kb="ROR">LINEAR</ProgressType>

    </RequestedOperation>

    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">

        <Metadata>

            <Atom/>

        </Metadata>

        <JobParameter schemaVersion="V1_0">

            <Metadata>

                <Atom/>

            </Metadata>

 <ParameterName kb="ROR" kxe="false">LogicalUnitUDID</ParameterName>
```

```
 <ParameterValue kxe="false" kb="CUR">25cd6928608fe811e48fa740f2e91329

50a087e7d483d711d4</ParameterValue>

        </JobParameter>

         <JobParameter schemaVersion="V1_0">

            <Metadata>

                <Atom/>

            </Metadata>

             <ParameterName kb="ROR" kxe="false">Capacity</ParameterName>

             <ParameterValue kxe="false" kb="CUR">20480</ParameterValue>

        </JobParameter>

    </JobParameters>

</JobRequest>
```

Related concepts:

"Cluster" on page 52
A Cluster is a set of one or more networked Virtual I/O Server (VIOS) partitions, where each VIOS within the cluster has access to a common set of physical volumes.

Related tasks:

"Shared Storage Pool" on page 54
Shared storage pool contains a logical organization of one or more physical volumes that will be used to provide block storage.

"Logical Unit" on page 58
Logical unit is a file in the pool that provides file-backed storage for Power systems.

Jobs

HMC REST API Framework supports invocation of long running operations through the use of Jobs.

For every Job there are two main semantics:

1. Job Object
2. Job Operation

On invoking any Job, a Job ID is returned to the user. This Job ID is to be used to poll the Status of the Job.

Resource to obtain Job Status using Job ID

`rest/api/uom/jobs/{job_id}`

Methods

Table 63. List of methods.

Method Name	Description	Internet Media Type
PUT	Request web server to perform an operation on the specified object based on the specified parameters.	application/ vnd.ibm.powervm.web+xml; type=JobRequest

EventLogger_ManagementConsole Job

Operation can be invoked to Log events to Developers Logs till the Job is Cancelled.

Resource

```
/rest/api/uom/ManagementConsole/{ManagementConsole_UUID}/do/EventLogger
```

Overview of EventLogger_ManagementConsole Job

We can invoke this job to log all the Generated Core Events to Developer Logs.

User needs to **Cancel** the job to terminate it. If not cancelled, the job continue indefinitely.

Sample Job Request

```
<JobRequest
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
   <Metadata>
       <Atom/>
   </Metadata>
   <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
       <Metadata>
           <Atom/>
       </Metadata>
       <OperationName kxe="false" kb="ROR">EventLogger</OperationName>
       <GroupName kxe="false" kb="ROR">ManagementConsole</GroupName>
       <ProgressType kxe="false" kb="ROR">DISCRETE</ProgressType>
   </RequestedOperation>
   <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
       <Metadata>
           <Atom/>
       </Metadata>
   </JobParameters>
</JobRequest>
```

CLIRunner_ManagementConsole Job

Operation to run CLI Commands using REST Framework on Hardware Management Console.

Resource

```
/rest/api/uom/ManagementConsole/{ManagementConsole_UUID}/do/CLIRunner
```

Overview of CLIRunner_ManagementConsole Job

On a Hardware Management Console, we can invoke this job, as a substitute for invoking CLI Commands on the command line interface. CLI commands that can be run through the CLIRunner job are as follows:

```
bkprofdata, chcod, chhmc, chhmcencr, chhmcfs.
chhmcldap, chhmcusr, chipsec, chled, chnportlogin.
chproxy, chpsm, chpwdpolicy, chsvc, chsvcevent.
chtskey, chusrtca, cpdump, cpfile, getdump.
getfile, hmcshutdown, hwdbg, lpcfgop, lscod.
lsdump, lsfru, lshmc, lshmcencr, lshmcfs.
lshmcusr, lsipsec, lslic, lslock, lslogon.
lsmediadev, lsproxy, lspsm, lspwdpolicy, lssacfg.
lssvcevents, lssysconn, lstskey, lsusrtca, lsvet.
migrlpar, mkauthkeys, mkprofdata, refdev, rmfile.
rmlock, rmprofdata, rmpwdpolicy, rmsyscfg, rmsysconn.
rmvterm, rsthwres, rstprofdata, rstupgdata, startdump.
termtask, updlic, viosvrcmd.
```

Sample Job Request

```
<JobRequest:JobRequest
xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_2">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_2">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kxe="false" kb="ROR">CLIRunner</OperationName>
        <GroupName kxe="false" kb="ROR">ManagementConsole</GroupName>
    </RequestedOperation>
    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_2">
        <Metadata>
            <Atom/>
        </Metadata>
      <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">cmd</ParameterName>
            <ParameterValue kxe="false" kb="CUR">lssyscfg -r sys</ParameterValue>
        </JobParameter>
      <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">acknowledgeThisAPIMayGoAwayInTheFuture</ParameterName>
            <ParameterValue kxe="false" kb="CUR">true</ParameterValue>
        </JobParameter>
    </JobParameters>
</JobRequest:JobRequest>
```

AddManagedSystem_ManagementConsole Job

Operation to add a particular Managed System to Management Console.

Resource

```
/rest/api/uom/ManagementConsole/{ManagementConsole_UUID}/do/AddManagedSystem
```

Overview of AddManagedSystem_ManagementConsole Job

On a Management Console, we can invoke this job, to add a Managed System to it.

Sample Job Request(s)

```
 <JobRequest:JobRequest
xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kb="ROR" kxe="false">AddManagedSystem</OperationName>
        <GroupName kb="ROR" kxe="false">ManagementConsole</GroupName>
    </RequestedOperation>
    <JobParameters kb="CUR" kxe="false" schemaVersion="V1_0">
        <Metadata>
```

```
            <Atom/>
          </Metadata>
   <JobParameter schemaVersion="V1_0"><Metadata><Atom/></Metadata>
          <ParameterName kb="ROR" kxe="false">host</ParameterName>
          <ParameterValue kb="CUR" kxe="false">9.3.180.18</ParameterValue>
       </JobParameter>
    <JobParameter schemaVersion="V1_0"><Metadata><Atom/></Metadata>
          <ParameterName kb="ROR" kxe="false">password</ParameterName>
          <ParameterValue kb="CUR" kxe="false">Passw0rd</ParameterValue>
       </JobParameter>
    <JobParameter schemaVersion="V1_0"><Metadata><Atom/></Metadata>
          <ParameterName kb="ROR" kxe="false">autoDiscover</ParameterName>
          <ParameterValue kb="CUR" kxe="false">auto</ParameterValue>
       </JobParameter>
    <JobParameter schemaVersion="V1_0"><Metadata><Atom/></Metadata>
          <ParameterName kb="ROR" kxe="false">force</ParameterName>
          <ParameterValue kb="CUR" kxe="false">true</ParameterValue>
       </JobParameter>
     </JobParameters>
</JobRequest:JobRequest>
```

PowerOn_ManagedSystem Job

Operation to PowerOn a particular ManagedSystem.

Resource

```
/rest/api/uom/ManagedSystem/{ManagedSystem_UUID}/do/PowerOn
```

Overview of PowerOn_ManagedSystem Job

For any ManagedSystem, we can invoke this job, to power it on.

Sample Job Request(s)

```
  <JobRequest
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kxe="false" kb="ROR">PowerOn</OperationName>
        <GroupName kxe="false" kb="ROR">ManagedSystem</GroupName>
        <ProgressType kxe="false" kb="ROR">LINEAR</ProgressType>
    </RequestedOperation>
    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">operation</ParameterName>
            <ParameterValue kxe="false" kb="CUR">on</ParameterValue>
        </JobParameter>
    </JobParameters>
</JobRequest>
```

PowerOff_ManagedSystem Job

Operation to power off a particular managed system.

Resource

```
/rest/api/uom/ManagedSystem/{ManagedSystem_UUID}/do/PowerOff
```

Overview of PowerOff_ManagedSystem Job

For any ManagedSystem, we can invoke this job, to power off the managed system.

Sample Job Request(s)

```
<JobRequest
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kxe="false" kb="ROR">PowerOff</OperationName>
        <GroupName kxe="false" kb="ROR">ManagedSystem</GroupName>
        <ProgressType kxe="false" kb="ROR">LINEAR</ProgressType>
    </RequestedOperation>
    <JobParameters kb="CUR" kxe="false" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kb="ROR" kxe="false">immediate</ParameterName>
            <ParameterValue kxe="false" kb="CUR">true</ParameterValue>
        </JobParameter>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kb="ROR" kxe="false">restart</ParameterName>
            <ParameterValue kxe="false" kb="CUR">true</ParameterValue>
        </JobParameter>
    </JobParameters>
</JobRequest>
```

ResetConnection_ManagedSystem Job

Operation to remove a particular Managed System connection from a Management Console.

Resource

```
/rest/api/uom/ManagementConsole/{ManagementConsole_UUID}/ManagedSystem/{ManagedSystem_UUID}/
do/ResetConnection
```

Overview of ResetConnection_ManagedSystem Job

On any Managed System, we can invoke this job to reset its connection with the Management Console.

Sample Job Request

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<JobRequest:JobRequest
xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
```

```
        <Atom/>
    </Metadata>
    <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kb="ROR" kxe="false">ResetConnection</OperationName>
        <GroupName kb="ROR" kxe="false">ManagedSystem</GroupName>
    </RequestedOperation>
    <JobParameters kb="CUR" kxe="false" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
    </JobParameters>
</JobRequest:JobRequest>
```

RemoveConnection_ManagedSystem Job

Operation to remove a particular Managed System connection from a Management Console.

Resource

```
/rest/api/uom/ManagementConsole/{ManagementConsole_UUID}/ManagedSystem/{ManagedSystem_UUID}/
do/RemoveConnection
```

Overview of RemoveConnection_ManagedSystem Job

On any Managed System, we can invoke this job to remove it from the Management Console.

Sample Job Request

```
 <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<JobRequest:JobRequest
xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kb="ROR" kxe="false">RemoveConnection</OperationName>
        <GroupName kb="ROR" kxe="false">ManagedSystem</GroupName>
    </RequestedOperation>
    <JobParameters kb="CUR" kxe="false" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
    </JobParameters>
</JobRequest:JobRequest>
```

QueryReservedMemoryRequiredForPartition_ManagedSystem Job

Operation to QueryReservedMemoryRequiredForPartition for a Managed System.

Resource

```
/rest/api/uom/ManagedSystem/{ManagedSystem_UUID}/do/QueryReservedMemoryRequiredForPartition
```

Overview of QueryReservedMemoryRequiredForPartition_ManagedSystem Job

On any Managed System, we can invoke this job to query for values that should be specified for memory attributes such asRequiredMinimumMemory, CurrentAvailableSystemMemory, CurrentAvailableMirroredMemory, RequiredMemory, or RequiredMirroredMemory.

For the job arameter **LogicalPartitonEnvironment**, the value must be one of the following: AIX/Linux, OS400, Virtual I/O Server.

Sample Job Request(s)

```
  <JobRequest
 xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/" schemaVersion="V1_0">
  <Metadata>
  <Atom/>
  </Metadata>
  <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_0">
   <Metadata>
   <Atom/>
   </Metadata>
   <OperationName kxe="false" kb="ROR">QueryReservedMemoryRequiredForPartition</OperationName>
   <GroupName kxe="false" kb="ROR">ManagedSystem</GroupName>
   <ProgressType kxe="false" kb="ROR">NONE</ProgressType>
  </RequestedOperation>
  <JobParameters kb="CUR" kxe="false" schemaVersion="V1_0">
   <Metadata>
   <Atom/>
   </Metadata>
   <JobParameter schemaVersion="V1_0">
    <Metadata>
    <Atom/>
    </Metadata>
    <ParameterName kxe="false" kb="ROR">LogicalPartitionEnvironment</ParameterName>
    <ParameterValue kxe="false" kb="CUR">OS400</ParameterValue>
   </JobParameter>
   <JobParameter schemaVersion="V1_0">
    <Metadata>
    <Atom/>
    </Metadata>
    <ParameterName kxe="false" kb="ROR">MaximumMemory</ParameterName>
    <ParameterValue kxe="false" kb="CUR">102400</ParameterValue>
   </JobParameter>
   <JobParameter schemaVersion="V1_0">
    <Metadata>
    <Atom/>
    </Metadata>
    <ParameterName kxe="false" kb="ROR">DesiredMemory</ParameterName>
    <ParameterValue kxe="false" kb="CUR">1024</ParameterValue>
   </JobParameter>
  </JobParameters>
 </JobRequest>
```

ModifySRIOVAdapterMode_ManagedSystem Job

Operation to modify the state of SRIOV Adapter of a Managed System.

Resource

`/rest/api/uom/ManagedSystem/{ManagedSystem_UUID}/do/ModifySRIOVAdapterMode`

Overview of ModifySRIOVAdapterMode_ManagedSystem Job

For a Managed System that has SRIOV Adapter Card installed on it, we can switch the configuration state of the SRIOV Adapter.

```
The Possible configurations for the adapter are:
 1. Configured (SRIOV) mode.
 2. Un-Configured (Dedicated) mode.
```

By invoking this job on SRIOV Adapter, we can switch the configuration from Configured (SRIOV) mode to Un-Configured (Dedicated) mode and vice-versa.

To switch the configuration from Configured (SRIOV) mode to Un-Configured (Dedicated) mode, set the value of the **operation** job parameter to `remove` (see Example 1).

To switch the configuration from Un-Configured (Dedicated) mode to Configured (SRIOV) mode, set the value of the **operation** job parameter to `add` (see Example 2).

Sample Job Request(s)

Example 1:

```
 <JobRequest
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kxe="false" kb="ROR">ModifySRIOVAdapterMode</OperationName>
        <GroupName kxe="false" kb="ROR">ManagedSystem</GroupName>
        <ProgressType kxe="false" kb="ROR">LINEAR</ProgressType>
    </RequestedOperation>
    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">adapterId</ParameterName>
            <ParameterValue kxe="false" kb="CUR">553844769</ParameterValue>
        </JobParameter>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">operation</ParameterName>
            <ParameterValue kxe="false" kb="CUR">remove</ParameterValue>
        </JobParameter>
    </JobParameters>
</JobRequest>
```

Example 2:

```
 <JobRequest
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kxe="false" kb="ROR">ModifySRIOVAdapterMode</OperationName>
```

```
        <GroupName kxe="false" kb="ROR">ManagedSystem</GroupName>
        <ProgressType kxe="false" kb="ROR">LINEAR</ProgressType>
    </RequestedOperation>
    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">adapterId</ParameterName>
            <ParameterValue kxe="false" kb="CUR">553844769</ParameterValue>
        </JobParameter>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">operation</ParameterName>
            <ParameterValue kxe="false" kb="CUR">add</ParameterValue>
        </JobParameter>
    </JobParameters>
</JobRequest>
```

ClearSRIOVPhysicalPortStatistics_ManagedSystem Job

Operation to clear the statistical data available on a particular SRIOV physical port of the Managed System.

Resource

```
/rest/api/uom/ManagedSystem/{ManagedSystem_UUID}/do/ClearSRIOVPhysicalPortStatistics
```

Overview of ClearSRIOVPhysicalPortStatistics_ManagedSystem Job

For a Managed System that has SRIOV Adapter Card installed on it, there are one or more physical ports available. For each of the physical ports, you can invoke this job to reset the statistics maintained regarding its Usage/Traffic-flow.

Sample Job Request(s)

```
 <JobRequest:JobRequest
xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kxe="false" kb="ROR">ClearSRIOVPhysicalPortStatistics</OperationName>
        <GroupName kxe="false" kb="ROR">ManagedSystem</GroupName>
    </RequestedOperation>
    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
      <JobParameter schemaVersion="V1_0"><Metadata><Atom/></Metadata>
         <ParameterName kb="ROR" kxe="false">physPortId</ParameterName>
         <ParameterValue kb="CUR" kxe="false">1</ParameterValue>
      </JobParameter>
      <JobParameter schemaVersion="V1_0"><Metadata><Atom/></Metadata>
         <ParameterName kb="ROR" kxe="false">sriovAdapterId</ParameterName>
```

```
        <ParameterValue kb="CUR" kxe="false">1</ParameterValue>
      </JobParameter>
    </JobParameters>
 </JobRequest:JobRequest>
```

GetNetworkBootDevices_LogicalPartition Job

Get Network Boot devices job is used to get network devices from the profile of logical partition.

Resource

```
/rest/api/uom/LogicalPartition/{LogicalPartitionUUID}/do/GetNetworkBootDevices
```

Request

User gets the list of network boot devices.

Table 64. Request Parameters

Request param	Description
LogicalPartitionProfileUUID	Logical Partition Profile UUID of partition to get network boot devices.

Response

This job would return the list of network boot devices belonging to partition.

Sample Job Request(s)

```
<JobRequest:JobRequest
 xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_1">
 <Metadata>
  <Atom />
 </Metadata>
 <RequestedOperation kb="CUR" kxe="false"
  schemaVersion="V1_1">
  <Metadata>
   <Atom />
  </Metadata>
  <OperationName kb="ROR" kxe="false">GetNetworkBootDevices
  </OperationName>
  <GroupName kb="ROR" kxe="false">LogicalPartition</GroupName>
 </RequestedOperation>
 <JobParameters kb="CUR" kxe="false" schemaVersion="V1_1">
  <Metadata>
   <Atom />
  </Metadata>
  <JobParameter schemaVersion="V1_1">
   <Metadata>
    <Atom />
   </Metadata>
   <ParameterName kxe="false" kb="ROR">LogicalPartitionProfileUUID
   </ParameterName>
   <ParameterValue kxe="false" kb="CUR">7778f188-bdb0-3fed-8ac6-2b255b8a11a7
   </ParameterValue>
  </JobParameter>
 </JobParameters>
</JobRequest:JobRequest>
```

LogicalPartition_RemoteRestart Job

Logical Partition remote restart Job is used to perform the remote restart operations on the logical partition. You can perform validate, recover, restart, cleanup, and cancel operations using this job.

Resource

`/rest/api/uom/ManagedSystem/{ManagedSystem_UUID}/LogicalPartition/{LogicalPartition_UUID}/do/RemoteRestart`
`/rest/api/uom/LogicalPartition/{LogicalPartition_UUID}/do/RemoteRestart`

Overview of LogicalPartition_RemoteRestart Job

On a PowerVM-partition-restart-capable Managed System, we can invoke this job on a PowerVM-partition-restart-capable LPAR to restart the partition on a destination Managed System.

Prior to invoking the restart operation, user can invoke the validate operation, to determine whether the restart operation is valid for the Logical Partition from source Managed System to the specified target Managed System.

Possible Job Parameters:

```
Operation
targetManagedSystemUUID
targetManagedSystem
Redundancy
Verbose
vlanbridge
force
usecurrdata
retaindev
targetRemoteHMCIPAddress
targetRemoteHMCUserId
sharedProcessorPoolName
sharedProcessorPoolID
virtualFCMappings
noconnection
```

Request

The source managed system and target managed system UUID must be given. Other optional parameters can be specified based on the operation type.

Table 65. Request parameters

Request param	Description
Operation	This is a mandatory parameter. Allowed values are `validate`, `recover`, `restart`, `cleanup`, and `cancel`.
targetManagedSystemUUID	This attribute is optional if the **`targetManagementSystem`** parameter is mentioned or if the **`operation value`** parameter is set to `cleanup`. When both MTMS and UUID are specified together, they must represent the same target system.
targetManagedSystem	This attribute is optional if the **`targetManagedSystemUUID`** parameter is mentioned or if the **`operation value`** parameter is set to `cleanup`. When both MTMS and UUID are specified together, they must represent the same target system.

Table 65. Request parameters (continued)

Request param	Description
Redundancy	Allowed values are `ALL` or `NONE`. The **`Redundancy`** parameter is applied for RedundantVios and IO (mpio). ALL sets a value of 1 to both **`redundantvios`** and **`mpio`** and NONE sets a value of 0 to **`redundantvios`** and 2 to **`mpio`**. **`RedundantVios`** specifies whether the partition is to be configured to use redundant VIOS. The default value is 2, if the option is not specified. You can specify 0 to indicate **No**, 1 to indicate **Yes**, 2 to indicate **If Possible**. IO (mpio) - specifies whether the management console is required to maintain an equivalent multipath I/O configuration of the partition virtual SCSI and virtual fiber channel adapters. The default value is 1. The values are 1 indicating **Yes**, 2 indicating **If Possible**.
Verbose	Allowed values are `TRUE` and `FALSE`. The value `True` sets the verbose output true and also the Detail level is set to 5. The value `False` sets the verbose output false and the Detail level is set 0. **Note:** Detail level is the level of detail requested from operating system commands issued by the HMC to all partitions participating in the remote restart values range from 0 (none) to 5 (highest). Default value is 1, if the option is not specified.
vlanbridge	Allowed values are 1 and 2. This parameter specifies whether each of the partition virtual Ethernet adapters is required to be configured so that it is bridged on the same VLAN to an external network . The default value is 1. The values are 1 indicating **Yes**, 2 indicating **If Possible**.
force	This option allows you to force a clean up or recover to proceed when errors are encountered. or remote restart of suspended simplified remote restart partition operation.
usecurrdata	This attribute is optional on the restart operation. Instructs the restart to proceed when the reserved storage device has stale data.
retaindev	This attribute is optional on the cleanup operation. Instructs the cleanup operation to not remove the reserved storage device from the source servers reserved storage device pool.
targetRemoteHMCIPAddress	This is a mandatory parameter, only for Cross-HMC-capable and Simplified-Remote-Restart capable partitions. Valid for the **`Validate`**, **`Restart`**, and **`Recover`** operations.
targetRemoteHMCUserId	This parameter is applicable nly for Cross-HMC-capable and Simplified-Remote-Restart capable partitions. Valid for the **`Validate`**, **`Restart`**, and **`Recover`** operations.
sharedProcessorPoolName	Allows the user to specify a shared processor pool on the target system by unis the shared processor pool name. Valid for the **`Validate`** and **`Restart`** operations. **Note:** You must specify the name of the shared processor pool only if you are not specifying the ID of the shared processor pool.

Table 65. Request parameters (continued)

Request param	Description
sharedProcessorPoolID	Allows the user to specify a specific shared processor pool on the target system by using shared processor pool ID. Valid for the **Validate** and **Restart** operations. **Note:** You must specify the ID of the shared processor pool only if you are not specifying the name of the shared processor pool.
virtualFCMappings	Allows the users to specify mappings for Virtual FC adapters on target system, such as target VIOS and target FC port. The mappings can be specified in the same format as in CLI. Valid for the **Validate** and **Restart** operations.
noconnection	Allows the user to proceed with the remote restart operations when the source Managed system is in no connection state. Valid for the **Validate** and **Restart** operations.

Response

This job would return the Job status like Complete_Ok, Failed_Before_Complete and so on.

Table 66. Response parameters

Response param	Description
returnCode	This attribute contains the value of the return code from underlying CLI invoked
result	The STDOUT out of underlying CLI invoked
ErrorData	The STDERR of underlying CLI invoked

Sample Job Request(s)

```
<JobRequest:JobRequest
xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_1_0">
   <Metadata>
       <Atom/>
   </Metadata>
   <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_1_0">
       <Metadata>
           <Atom/>
       </Metadata>
       <OperationName kb="ROR" kxe="false">RemoteRestart</OperationName>
       <GroupName kb="ROR" kxe="false">LogicalPartition</GroupName>
   </RequestedOperation>
   <JobParameters kb="CUR" kxe="false" schemaVersion="V1_1_0">
       <Metadata>
           <Atom/>
       </Metadata>
      <JobParameter schemaVersion="V1_0">
     <Metadata>
         <Atom/>
     </Metadata>
     <ParameterName kb="ROR" kxe="false">targetManagedSystem</ParameterName>
     <ParameterValue kxe="false" kb="CUR">HV4-221</ParameterValue>
 </JobParameter>
      <JobParameter schemaVersion="V1_0">
     <Metadata>
```

```
        <Atom/>
      </Metadata>
      <ParameterName kb="ROR" kxe="false">targetManagedSystemUUID</ParameterName>
      <ParameterValue kxe="false" kb="CUR">b73f1565-0ae4-3070-8eac-58f35a81e898</ParameterValue>
  </JobParameter>
      <JobParameter schemaVersion="V1_0">
      <Metadata>
          <Atom/>
      </Metadata>
      <ParameterName kb="ROR" kxe="false">operation</ParameterName>
      <ParameterValue kxe="false" kb="CUR">validate</ParameterValue>
  </JobParameter>
    </JobParameters>
</JobRequest:JobRequest>
```

Related reference:

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

Migrate_LogicalPartition Job

Operation to Migrate a particular Logical Partition from one Managed System to another.

Resource

```
/rest/api/uom/LogicalPartition/{LogicalPartition_UUID}/do/Migrate
```

Overview of Migrate_LogicalPartition Job

On a Migration Capable ManagedSystem, we can invoke this job on a Migration Capable LPAR to migrate it from one Managed System to another.

Prior to invoking Migrate Job, user can invoke MigrateValidate Job, to determine whether the Migrate Job is valid for this LogicalPartition from Source Managed System to the specified Target ManagedSystem.

```
Possible Job Parameters:

TargetManagedSystemName
TargetRemoteHMCIPAddress
TargetRemoteHMCUserID
DestinationLparID
TargetProfileName
RedundantVIOS
MultipathIOOverride
VLANBridgeOverride
VSIOverride
ProtectStorage
VirtualFCMappings
VirtualSCSIMappings
SourceMSPID
SourceMSPIPaddr
DestMSPID
DestMSPIPaddr
SharedProcPoolID
PrimaryRSViosID
DetailedLevel
WaitTime
```

Sample Job Request(s)

Example 1:

```
 <JobRequest
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
```

```
        <Atom/>
    </Metadata>
    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kxe="false" kb="ROR">Migrate</OperationName>
        <GroupName kxe="false" kb="ROR">LogicalPartition</GroupName>
        <ProgressType kxe="false" kb="ROR">DISCRETE</ProgressType>
    </RequestedOperation>
    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">TargetManagedSystemName</ParameterName>
            <ParameterValue kxe="false" kb="CUR">vrml12-fsp</ParameterValue>
        </JobParameter>
    </JobParameters>
</JobRequest>
```

Example 2:

```
 <JobRequest
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
   <Metadata>
       <Atom/>
   </Metadata>
   <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
       <Metadata>
           <Atom/>
       </Metadata>
       <OperationName kxe="false" kb="ROR">Migrate</OperationName>
       <GroupName kxe="false" kb="ROR">LogicalPartition</GroupName>
       <ProgressType kxe="false" kb="ROR">DISCRETE</ProgressType>
   </RequestedOperation>
   <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
       <Metadata>
           <Atom/>
       </Metadata>
       <JobParameter schemaVersion="V1_0">
           <Metadata>
               <Atom/>
           </Metadata>
           <ParameterName kxe="false" kb="ROR">TargetManagedSystemName</ParameterName>
           <ParameterValue kxe="false" kb="CUR">firebird110</ParameterValue>
       </JobParameter>
       <JobParameter schemaVersion="V1_0">
           <Metadata>
               <Atom/>
           </Metadata>
           <ParameterName kxe="false" kb="ROR">DestinationLparID</ParameterName>
           <ParameterValue kxe="false" kb="CUR">20</ParameterValue>
       </JobParameter>
    </JobParameters>
</JobRequest>
```

Example 3:

```
 <JobRequest xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
   <Metadata>
       <Atom/>
   </Metadata>
```

```
<RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <OperationName kxe="false" kb="ROR">Migrate</OperationName>
    <GroupName kxe="false" kb="ROR">LogicalPartition</GroupName>
    <ProgressType kxe="false" kb="ROR">DISCRETE</ProgressType>
</RequestedOperation>
<JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <JobParameter schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <ParameterName kxe="false" kb="ROR">TargetManagedSystemName</ParameterName>
        <ParameterValue kxe="false" kb="CUR">vrml12-fsp</ParameterValue>
    </JobParameter>
    <JobParameter schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <ParameterName kxe="false" kb="ROR">TargetRemoteHMCIPAddress</ParameterName>
        <ParameterValue kxe="false" kb="CUR">9.124.63.149</ParameterValue>
    </JobParameter>
    <JobParameter schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <ParameterName kxe="false" kb="ROR">TargetRemoteHMCUserID</ParameterName>
        <ParameterValue kxe="false" kb="CUR">hscroot</ParameterValue>
    </JobParameter>
    <JobParameter schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <ParameterName kxe="false" kb="ROR">DestinationLparID</ParameterName>
        <ParameterValue kxe="false" kb="CUR">20</ParameterValue>
    </JobParameter>
    <JobParameter schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <ParameterName kxe="false" kb="ROR">VirtualSCSIMappings</ParameterName>
        <ParameterValue kxe="false" kb="CUR">1/vrml12-vios1//3</ParameterValue>
    </JobParameter>
    <JobParameter schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <ParameterName kxe="false" kb="ROR">SourceMSPIPaddr</ParameterName>
        <ParameterValue kxe="false" kb="CUR">9.3.46.29</ParameterValue>
    </JobParameter>
    <JobParameter schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <ParameterName kxe="false" kb="ROR">DestMSPIPaddr</ParameterName>
        <ParameterValue kxe="false" kb="CUR">9.3.46.39</ParameterValue>
    </JobParameter>
  </JobParameters>
</JobRequest>
```

MigrateAbort_LogicalPartition Job

Operation to Abort the previously invoked Migration of a Logical Partition.

Resource

```
/rest/api/uom/LogicalPartition/{LogicalPartition_UUID}/do/MigrateAbort
```

Overview of MigrateAbort_LogicalPartition Job

On a Migration Capable ManagedSystem, we can invoke this job on a LPAR, which is in one of the intermediate states of Migration.

We can specify the "timeout" value in the Job request in units of Seconds.

This example shows we specified that the Job has to complete within 60 Seconds of invocation.

Sample Job Request(s)

```
 <JobRequest
 xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kxe="false" kb="ROR">MigrateAbort</OperationName>
        <GroupName kxe="false" kb="ROR">LogicalPartition</GroupName>
        <ProgressType kxe="false" kb="ROR">DISCRETE</ProgressType>
    </RequestedOperation>
    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">timeout</ParameterName>
            <ParameterValue kxe="false" kb="CUR">60</ParameterValue>
        </JobParameter>
    </JobParameters>
</JobRequest>
```

MigrateRecover_LogicalPartition Job

Operation to recover a logical partition that has been stuck in an intermediate state after failure of the preceding migrate operation.

Resource

```
/rest/api/uom/LogicalPartition/{LogicalPartition_UUID}/do/MigrateRecover
```

Overview of MigrateRecover_LogicalPartition Job

On a Migration Capable ManagedSystem, we can invoke this job on a LPAR that is stuck in one of the intermediate states of Migration.

Sample Job Request(s)

```
  <JobRequest
 xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
```

```
    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kxe="false" kb="ROR">MigrateRecover</OperationName>
        <GroupName kxe="false" kb="ROR">LogicalPartition</GroupName>
        <ProgressType kxe="false" kb="ROR">DISCRETE</ProgressType>
    </RequestedOperation>
    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">Force</ParameterName>
            <ParameterValue kxe="false" kb="CUR">true</ParameterValue>
        </JobParameter>
    </JobParameters>
</JobRequest>
```

MigrateValidate_LogicalPartition Job

Operation to Validate the Migrate operation of a particular Logical Partition from one Managed System to another.

Resource

```
/rest/api/uom/LogicalPartition/{LogicalPartition_UUID}/do/MigrateValidate
```

Overview of MigrateValidate_LogicalPartition Job

On a Migration Capable ManagedSystem, we can invoke this job on an LPAR to validate if it can be migrated to the Target ManagedSystem in the manner scpecified.

Prior to invoking Migrate Job, user can invoke MigrateValidate Job, to determine whether the Migrate Job is valid for this LogicalPartition from Source Managed System to the specified Target ManagedSystem.

Possible Job Parameters:

```
TargetManagedSystemName
TargetRemoteHMCIPAddress
TargetRemoteHMCUserID
DestinationLparID
TargetProfileName
RedundantVIOS
MultipathIOOverride
VLANBridgeOverride
VSIOverride
ProtectStorage
VirtualFCMappings
VirtualSCSIMappings
SourceMSPID
SourceMSPIPaddr
DestMSPID
DestMSPIPaddr
SharedProcPoolID
PrimaryRSViosID
DetailedLevel
WaitTime
```

Sample Job Request(s)

Example 1:

```
<JobRequest
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
   <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
       <Metadata>
           <Atom/>
       </Metadata>
       <OperationName kxe="false" kb="ROR">MigrateValidate</OperationName>
       <GroupName kxe="false" kb="ROR">LogicalPartition</GroupName>
       <ProgressType kxe="false" kb="ROR">DISCRETE</ProgressType>
   </RequestedOperation>
   <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
       <Metadata>
           <Atom/>
       </Metadata>
       <JobParameter schemaVersion="V1_0">
           <Metadata>
               <Atom/>
           </Metadata>
           <ParameterName kxe="false" kb="ROR">TargetManagedSystemName</ParameterName>
           <ParameterValue kxe="false" kb="CUR">vrml12-fsp</ParameterValue>
       </JobParameter>
    </JobParameters>
    </JobRequest>
```

Example 2:

```
<JobRequest xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
   <Metadata>
       <Atom/>
   </Metadata>
   <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
       <Metadata>
           <Atom/>
       </Metadata>
       <OperationName kxe="false" kb="ROR">MigrateValidte</OperationName>
       <GroupName kxe="false" kb="ROR">LogicalPartition</GroupName>
       <ProgressType kxe="false" kb="ROR">DISCRETE</ProgressType>
   </RequestedOperation>
   <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
       <Metadata>
           <Atom/>
       </Metadata>
       <JobParameter schemaVersion="V1_0">
           <Metadata>
               <Atom/>
           </Metadata>
           <ParameterName kxe="false" kb="ROR">TargetManagedSystemName</ParameterName>
           <ParameterValue kxe="false" kb="CUR">firebird110</ParameterValue>
       </JobParameter>
       <JobParameter schemaVersion="V1_0">
           <Metadata>
               <Atom/>
           </Metadata>
           <ParameterName kxe="false" kb="ROR">DestinationLparID</ParameterName>
           <ParameterValue kxe="false" kb="CUR">20</ParameterValue>
       </JobParameter>
     </JobParameters>
</JobRequest>
```

Example 3:

```
<JobRequest xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
   <Metadata>
       <Atom/>
   </Metadata>
   <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
       <Metadata>
           <Atom/>
       </Metadata>
       <OperationName kxe="false" kb="ROR">MigrateValidate</OperationName>
       <GroupName kxe="false" kb="ROR">LogicalPartition</GroupName>
       <ProgressType kxe="false" kb="ROR">DISCRETE</ProgressType>
   </RequestedOperation>
   <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
       <Metadata>
           <Atom/>
       </Metadata>
       <JobParameter schemaVersion="V1_0">
           <Metadata>
               <Atom/>
           </Metadata>
           <ParameterName kxe="false" kb="ROR">TargetManagedSystemName</ParameterName>
           <ParameterValue kxe="false" kb="CUR">vrml12-fsp</ParameterValue>
       </JobParameter>
       <JobParameter schemaVersion="V1_0">
           <Metadata>
               <Atom/>
           </Metadata>
           <ParameterName kxe="false" kb="ROR">TargetRemoteHMCIPAddress</ParameterName>
           <ParameterValue kxe="false" kb="CUR">9.124.63.149</ParameterValue>
       </JobParameter>
       <JobParameter schemaVersion="V1_0">
           <Metadata>
               <Atom/>
           </Metadata>
           <ParameterName kxe="false" kb="ROR">TargetRemoteHMCUserID</ParameterName>
           <ParameterValue kxe="false" kb="CUR">hscroot</ParameterValue>
       </JobParameter>
       <JobParameter schemaVersion="V1_0">
           <Metadata>
               <Atom/>
           </Metadata>
           <ParameterName kxe="false" kb="ROR">DestinationLparID</ParameterName>
           <ParameterValue kxe="false" kb="CUR">20</ParameterValue>
       </JobParameter>
       <JobParameter schemaVersion="V1_0">
           <Metadata>
               <Atom/>
           </Metadata>
           <ParameterName kxe="false" kb="ROR">VirtualSCSIMappings</ParameterName>
           <ParameterValue kxe="false" kb="CUR">1/vrml12-vios1//3</ParameterValue>
       </JobParameter>
       <JobParameter schemaVersion="V1_0">
           <Metadata>
               <Atom/>
           </Metadata>
           <ParameterName kxe="false" kb="ROR">SourceMSPIPaddr</ParameterName>
           <ParameterValue kxe="false" kb="CUR">9.3.46.29</ParameterValue>
       </JobParameter>
       <JobParameter schemaVersion="V1_0">
           <Metadata>
               <Atom/>
           </Metadata>
           <ParameterName kxe="false" kb="ROR">DestMSPIPaddr</ParameterName>
           <ParameterValue kxe="false" kb="CUR">9.3.46.39</ParameterValue>
```

```
      </JobParameter>
    </JobParameters>
  </JobRequest>
```

PowerOff_LogicalPartition Job

Operation to Power Off a particular Logical Partition of a Managed System.

Resource

```
/rest/api/uom/LogicalPartition/{LogicalPartition_UUID}/do/PowerOff
```

Overview of PowerOff_LogicalPartition Job

On any Logical Partition of a ManagedSystem, we can invoke this job to power off the partition.

The possible values for Job Parameter "Operation" are:

- shutdown
- osshutdown
- dumprestart
- retrydump

Sample Job Request(s)

```
 <JobRequest
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kxe="false" kb="ROR">PowerOff</OperationName>
        <GroupName kxe="false" kb="ROR">LogicalPartition</GroupName>
        <ProgressType kxe="false" kb="ROR">DISCRETE</ProgressType>
    </RequestedOperation>
    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">immediate</ParameterName>
            <ParameterValue kxe="false" kb="CUR">false</ParameterValue>
        </JobParameter>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">restart</ParameterName>
            <ParameterValue kxe="false" kb="CUR">false</ParameterValue>
        </JobParameter>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">operation</ParameterName>
```

```
            <ParameterValue kxe="false" kb="CUR">shutdown</ParameterValue>
        </JobParameter>
    </JobParameters>
</JobRequest>
```

PowerOn_LogicalPartition Job

The PowerOn_LogicalPartition job is used to power on a logical partition.

Resource

`/rest/api/uom/LogicalPartition/LogicalPartitionUid/do/PowerOn`

Request

You must power on a logical partition to start managing it.

Table 67. Request parameters

Request param	Description
bootmode	The boot mode to use when activating an AIX, Linux, or Virtual I/O Server partition. Valid values are for `norm` for normal, `dd` for diagnostic with default boot list, `ds` for diagnostic with stored boot list, `of` for Open Firmware OK prompt, or `sms` for System Management Services.
iIPLsource	The inital program load (IPL) source to use when activating an IBM i partition. Valid values are a, b, c, or d.
iIPV4address	The IPv4 address of the Ethernet adapter or HEA logical port that is assigned as the alternate restart device of the IBM i partition on which tthe network installation operation will be performed.
IPAddress	The IPv4 or IPv6 address of the Ethernet adapter or HEA logical port that is assigned as the alternate restart device of the IBM i partition on which the network installation operation will be performed.
SubnetMask	The network mask associated with the IPv4 address specified with the --ip option.
iNetmask	The network mask associated with the IP address specified with the --ip option.
iGateway	The IPv4 address of the gateway to use when performing a network installation of an IBM i partition.
Gateway	The IPv4 or IPv6 address of the gateway to use when performing a network installation of an IBM i partition.
iServerIPV4address	The IPv4 address of the server that contains the network installation images for an IBM i partition.
ServerIPAddress	The IPv4 or IPv6 address of the server that contains the network installation images for an IBM i partition.
iServerdir	The directory on the server that contains the network installation images for an IBM i partition.
iSpeed	The speed setting to use when performing a network installation of an IBM i partition. Valid values are: • auto - automatically detect and set speed • 1 - 1 Mbps • 10 - 10 Mbps • 100 - 100 Mbps • 1000 - 1000 Mbps If this option is not specified, it defaults to auto.

Table 67. Request parameters (continued)

Request param	Description
ConnectionSpeed	The speed setting to use when performing a network installation of an IBM i partition. Valid values are: • auto - automatically detect and set speed • 1 - 1 Mbps • 10 - 10 Mbps • 100 - 100 Mbps • 1000 - 1000 Mbps If this option is not specified, it defaults to auto.
iDuplex	The duplex setting to use when performing a network installation of an IBM i partition. Valid values are: • auto - automatically detect and set duplex • half - half duplex • full - full duplex If this option is not specified, it defaults to auto.
DuplexMode	The duplex setting to use when performing a network installation of an IBM i partition. Valid values are: • auto - automatically detect and set duplex • half - half duplex • full - full duplex If this option is not specified, it defaults to auto.
SlotPhysicalLocationCode	The physical location code to be used for a Netboot operation (-l option in CLI).
BootImageFileName	Network boot image file name, mandatory parameter only for IPv6 Netboot (-B option in CLI).
IBMiImageServerDirectory	The server directory containing IBM i image.
iMtu	The maximum transmission unit, in bytes, to use when performing a network installation of an IBM i partition. Valid values are 1500 bytes or 9000 bytes. If this option is not specified, it defaults to 1500 bytes.
MaximumTransmissionUnit	The maximum transmission unit, in bytes, to use when performing a network installation of an IBM i partition. Valid values are 1500 bytes or 9000 bytes. The default value is 1500 bytes.
Timeout	If this value is not specified, by default the value is set to 3600000 milli seconds (60 minutes).
novsi	When activating a partition that uses virtual Ethernet adapter, Virtual Station Interface (VSI) profiles, use this option to allow the partition to be activated without VSI profiles. Note that when this option is specified, the partition continues to be activated with VSI profiles, if possible.
VLAN	The VLAN ID to be used when performing a network installation of an IBM i partition. The valid values are 1 (lowest) - 4094 (highest). This attribute is optional.

Table 67. Request parameters (continued)

Request param	Description
force	This option allows you to shut down a Virtual IO Server (VIOS) partition which is the only VIOS partition providing access to the paging device or reserved storage device for one or more partitions. This option also allows you to activate a shared memory partition during the following conditions: • The partition is configured to use redundant paging VIOS partitions, but only one paging VIOS partition is currently available and that paging VIOS partition has access to an available paging space device in the shared memory pool. • The partition is configured to use redundant paging VIOS partitions, but no paging space is available device in the shared memory pool that can be accessed by both paging VIOS partitions. However, a paging space device is available in the shared memory pool that can be accessed by one of the paging VIOS partitions. • The partition is configured to use a single paging VIOS partition that is not currently available or does not have access to an available paging space device in the shared memory pool, but another paging VIOS partition is available in the shared memory pool that has access to an available paging space device in the shared memory pool.
keylock	The keylock position to set. Valid values are `manual` and `norm` for normal. This option is required while setting the keylock position for a partition or a managed system. This option is optional when powering on a managed system or activating a partition.
LogicalPartitionProfile	The name of the profile against which the partition will be powered on.

Response

This job returns the job ID to be used to poll the iob status.

Sample Job Request(s)

```
<JobRequest:JobRequest
 xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_1">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kb="ROR" kxe="false">PowerOn</OperationName>
        <GroupName kb="ROR" kxe="false">LogicalPartition</GroupName>
    </RequestedOperation>
    <JobParameters kb="CUR" kxe="false" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
   <JobParameter schemaVersion="V1_0">
           <Metadata>
               <Atom/>
           </Metadata>
           <ParameterName kxe="false" kb="ROR">force</ParameterName>
           <ParameterValue kxe="false" kb="CUR">false</ParameterValue>
      </JobParameter>
      <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
```

```
            </Metadata>
            <ParameterName kxe="false" kb="ROR">LogicalPartitionProfile</ParameterName>
            <ParameterValue kxe="false" kb="CUR">lpar_prof</ParameterValue>
        </JobParameter>
  <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">novsi</ParameterName>
            <ParameterValue kxe="false" kb="CUR">true</ParameterValue>
        </JobParameter>
  <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">bootmode</ParameterName>
            <ParameterValue kxe="false" kb="CUR">norm</ParameterValue>
        </JobParameter>
  <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">keylock</ParameterName>
            <ParameterValue kxe="false" kb="CUR">manual</ParameterValue>
        </JobParameter>
   </JobParameters>
</JobRequest:JobRequest>
```

ConfigDevice_VirtualIOServer Job

Configure device Job on Virtual IO Server will help to configure the devices on VirtualIOServers.

Resource

`/rest/api/uom/VirtualIOServer/{VirtualIOServer_UUID}/do/ConfigDevice`

Request

Configure device Job on Virtual IO Server will configure the devices. If there are any devices detected that have no device software installed when configuring devices, this Job returns a warning message with the name or a list of possible names for the device package that must be installed.

Table 68. Request Parameters

Request param	Description
devName	Specifies the named device to be configured. This is a optional attribute if left blank, all devices are tried to configure.

Response

This job would return message obtained from VIOS on standard output and error console and also the valid Job status like Complete_Ok, Failed_Before_Complete and so on.

Table 69. Response params

Response Parameters	Description
StdError	Error message obtained from VIOS on standard error console.
StdOut	Message obtained from VIOS on standard output console.

Sample Job Request(s)

```
<JobRequest:JobRequest
 xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kb="ROR" kxe="false">Delete</OperationName>
        <GroupName kb="ROR" kxe="false">Cluster</GroupName>
    </RequestedOperation>
    <JobParameters kb="CUR" kxe="false" schemaVersion="V1_1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <JobParameter schemaVersion="V1_0">
      <Metadata>
          <Atom/>
      </Metadata>
      <ParameterName kb="ROR" kxe="false">devName</ParameterName>
      <ParameterValue kxe="false" kb="CUR">scsi0</ParameterValue>
  </JobParameter>
    </JobParameters>
</JobRequest:JobRequest>
```

Related reference:

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

GetNetworkBootDevices_VirtualIOServer Job

Get Network Boot devices job is used to get network devices from the profile of Virtual I/O Server (VIOS).

Resource

```
/rest/api/uom/VirtualIOServer/{VirtualIOServerUUID}/do/GetNetworkBootDevices
```

Request

User gets the list of network boot devices.

Table 70. Request Parameters

Request parameter	Description
VirtualIOServerUUID	Virtual I/O Server UUID of partition to get network boot devices.

Response

This job returns the list of network boot devices that belong to the VIOS.

Sample Job Request(s)

```
<JobRequest:JobRequest
 xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
```

```
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_1">
<Metadata>
 <Atom />
</Metadata>
<RequestedOperation kb="CUR" kxe="false"
 schemaVersion="V1_1">
 <Metadata>
  <Atom />
 </Metadata>
 <OperationName kb="ROR" kxe="false">GetNetworkBootDevices
 </OperationName>
 <GroupName kb="ROR" kxe="false">VirtualIOServer</GroupName>
</RequestedOperation>
<JobParameters kb="CUR" kxe="false" schemaVersion="V1_1">
 <Metadata>
  <Atom />
 </Metadata>
 <JobParameter schemaVersion="V1_1">
  <Metadata>
   <Atom />
  </Metadata>
  <ParameterName kxe="false" kb="ROR">LogicalPartitionProfileUUID
  </ParameterName>
  <ParameterValue kxe="false" kb="CUR">7778f188-bdb0-3fed-8ac6-2b255b8a11a7
  </ParameterValue>
 </JobParameter>
</JobParameters>
</JobRequest:JobRequest>
```

PowerOff_VirtualIOServer Job

Operation to power off a logical partition of a Managed System.

Resource

```
/rest/api/uom/VirtualIOServer/{VirtualIOServer_UUID}/do/PowerOff
```

Overview of PowerOff_VirtualIOServer Job

You can invoke this job on any Virtual I/O Server of a Managed System to power it off.

The possible values for the **Operation** job parameter are:

- shutdown
- osshutdown
- dumprestart
- retrydump

Sample Job Request(s)

```
<JobRequest
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kxe="false" kb="ROR">PowerOff</OperationName>
        <GroupName kxe="false" kb="ROR">VirtualIOServer</GroupName>
        <ProgressType kxe="false" kb="ROR">DISCRETE</ProgressType>
```

```
    </RequestedOperation>
    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">immediate</ParameterName>
            <ParameterValue kxe="false" kb="CUR">false</ParameterValue>
        </JobParameter>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">restart</ParameterName>
            <ParameterValue kxe="false" kb="CUR">false</ParameterValue>
        </JobParameter>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">operation</ParameterName>
            <ParameterValue kxe="false" kb="CUR">shutdown</ParameterValue>
        </JobParameter>
    </JobParameters>
</JobRequest>
```

PowerOn_VirtualIOServer Job

The PowerOn_VirtualIOServer Job is used to power on a Virtual I/O Server.

Resource

`/rest/api/uom/VirtualIOServer/{VirtualIOServer_UUID}/do/PowerOn`

Request

User has to power on a Virtual I/O Server to start managing it.

Table 71. Request parameters

Request param	Description
bootmode	The boot mode to use when activating an AIX, Linux, or Virtual I/O Server partition. Valid values are: **norm** normal **dd** diagnostic with default boot list **ds** diagnostic with stored boot list **of** Open Firmware OK prompt **sms** System Management Services.
novsi	When activating a partition that uses virtual Ethernet adapter Virtual Station Interface (VSI) profiles, use this option to allow the partition to be activated without VSI profiles. Note that when this option is specified, the partition will still be activated with VSI profiles if possible.

Table 71. Request parameters (continued)

Request param	Description
force	This option allows you to shut down a Virtual I/O Server (VIOS) partition which is the only VIOS partition providing access to the paging device or reserved storage device for one or more partitions. This option also allows you to activate a shared memory partition under any of the following conditions: • The partition is configured to use redundant paging VIOS partitions, but only one paging VIOS partition is currently available and that paging VIOS partition has access to an available paging space device in the shared memory pool. • The partition is configured to use redundant paging VIOS partitions, but there is no available paging space device in the shared memory pool that can be accessed by both paging VIOS partitions. However, there is an available paging space device in the shared memory pool that can be accessed by one of the paging VIOS partitions. • The partition is configured to use a single paging VIOS partition which is not currently available or does not have access to an available paging space device in the shared memory pool, but there is another paging VIOS partition in the shared memory pool that is available and that has access to an available paging space device in the shared memory pool.
keylock	The keylock position to set. Valid values are manual and normal. This option is required when setting the keylock position for a partition or a managed system. This option is optional when powering on a managed system or activating a partition.
LogicalPartitionProfile	The name of the profile against which the partition will be powered on

Response

This job would return the job ID to be used to poll the Job status.

Sample Job Request(s)

```
<JobRequest:JobRequest
 xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kb="ROR" kxe="false">PowerOn</OperationName>
        <GroupName kb="ROR" kxe="false">VirtualIOServer</GroupName>
    </RequestedOperation>
    <JobParameters kb="CUR" kxe="false" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
```

```
      </Metadata>
   <JobParameter schemaVersion="V1_0">
          <Metadata>
              <Atom/>
          </Metadata>
          <ParameterName kxe="false" kb="ROR">force</ParameterName>
          <ParameterValue kxe="false" kb="CUR">false</ParameterValue>
      </JobParameter>
      <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">LogicalPartitionProfile</ParameterName>
            <ParameterValue kxe="false" kb="CUR">vios_profile</ParameterValue>
        </JobParameter>
  <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">novsi</ParameterName>
            <ParameterValue kxe="false" kb="CUR">true</ParameterValue>
        </JobParameter>
  <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">bootmode</ParameterName>
            <ParameterValue kxe="false" kb="CUR">norm</ParameterValue>
        </JobParameter>
  <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">keylock</ParameterName>
            <ParameterValue kxe="false" kb="CUR">manual</ParameterValue>
        </JobParameter>
   </JobParameters>
</JobRequest:JobRequest>
```

GetFreePhysicalVolumes_VirtualIOServer Job

The GetFreePhysicalVolumes job returns the free physical volumes that belong to Virtual I/O Server (VIOS) on which it was queried. It also provides the **FibreChannelBackedOnly** option to filter the Fibre-Channel-backed physical volumes.

Resource

`/rest/api/uom/VirtualIOServer/{VirtualIOServer_UUID}/do/GetFreePhysicalVolumes`

Since: Version 1_3_0

Request

The **FibreChannelBackedOnly** is the only optional attribute. When this parameter value is set to true, the job returns a list of physical volumes that are fibre channel backed and available for usage. If the value is set to false, the job returns a list of all physical volumes that are available for usage.

Table 72. Request parameter

Request parameter	Description
FibreChannelBackedOnly	To filter only Fibre-Channel-backed physical volume (PV).

Response

This job returns the **PhysicalVolumeCollection** parameter that contains free physical volumes and also the valid job status values such as `Complete_Ok` and `Failed_Before_Complete`.

Table 73. Response parameters

Response parameter	Description
result	**PhysicalVolumeCollection** that contains free PVs.

Sample Job Request(s)

```
 <JobRequest:JobRequest xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"

 xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"

xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_3_0">

    <Metadata>

        <Atom/>

    </Metadata>

    <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_3_0">

        <Metadata>

            <Atom/>

        </Metadata>

        <OperationName kb="ROR" kxe="false">GetFreePhysicalVolumes</OperationName>

        <GroupName kb="ROR" kxe="false">VirtualIOServer</GroupName>

    </RequestedOperation>

    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_3_0">

        <Metadata>

            <Atom/>

        </Metadata>

        <JobParameter schemaVersion="V1_3_0">

      <Metadata>

          <Atom/>

      </Metadata>

      <ParameterName kb="ROR" kxe="false">FibreChannelBackedOnly</ParameterName>

      <ParameterValue kxe="false" kb="CUR">true</ParameterValue>

  </JobParameter>

    </JobParameters>

</JobRequest:JobRequest>
```

Related reference:

"Virtual I/O Server" on page 18

The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical

partitions within the server.

Create_Cluster Job

The cluster create job is used to create a cluster with shared storage pool. The cluster and shared storage pool details are available in an XML format.

Resource

`/rest/api/uom/Cluster/do/Create`

Since: Version 1_1_0

Request

Because the cluster and shared storage pool coexists, a cluster cannot be created without a shared storage pool or vice versa. You must create both cluster and shared storage pool at same time.

Table 74. Request Parameters

Request parameter	Description
clusterXml	Cluster details containing node and repository disk information in an XML format.
sspXml	Shared storage pool details containing pool disk information in an XML format.

Response

This job returns valid Job status values such as `Complete_Ok` and `Failed_Before_Complete`.

Sample Job Request(s)

```
<JobRequest:JobRequest xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kb="ROR" kxe="false">Create</OperationName>
        <GroupName kb="ROR" kxe="false">Cluster</GroupName>
    </RequestedOperation>
    <JobParameters kb="CUR" kxe="false" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
 <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kb="ROR" kxe="false" >clusterXml</ParameterName>
<ParameterValue kxe="false" kb="CUR" ><![CDATA[<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Cluster:Cluster
            xmlns:Cluster="http://www.ibm.com/xmlns/systems/power/firmware/uom/mc/2012_10/"
            xmlns="http://www.ibm.com/xmlns/systems/power/firmware/uom/mc/2012_10/"
            xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
            <Metadata>
              <Atom/>
            </Metadata>
            <ClusterName kb="COR" kxe="false">testCluster</ClusterName>
```

```
            <RepositoryDisk kb="CUD" kxe="false" schemaVersion="V1_0">
                <Metadata>
                    <Atom/>
                </Metadata>
                <PhysicalVolume schemaVersion="V1_0">
                    <Metadata>
                        <Atom/>
                    </Metadata>
                    <Description kb="CUD" kxe="false">MPIO IBM 2076 FC Disk</Description>
<UniqueDeviceID kb="ROR" kxe="false">3321360050768028087920000000000003804214503IBMfcp
</UniqueDeviceID>
                    <VolumeCapacity kb="CUR" kxe="false">20480</VolumeCapacity>
                    <VolumeName kb="CUR" kxe="false">hdisk8</VolumeName>
                    <VolumeState kb="ROR" kxe="false">active</VolumeState>
                    <IsFibreChannelBacked kxe="false" kb="ROR">true</IsFibreChannelBacked>
                </PhysicalVolume>
            </RepositoryDisk>
            <Node kb="CUD" kxe="false" schemaVersion="V1_0">
                <Metadata>
                    <Atom/>
                </Metadata>
                <Node schemaVersion="V1_0">
                    <Metadata>
                        <Atom/>
                    </Metadata>
                    <HostName kb="CUD" kxe="false">pfwsdmc77.in.ibm.com</HostName>
                    <PartitionID kb="CUD" kxe="false">1</PartitionID>
                    <MachineTypeModelAndSerialNumber kb="CUD" kxe="false" schemaVersion="V1_0">
                        <Metadata>
                            <Atom/>
                        </Metadata>
                        <MachineType kb="CUR" kxe="false">7895</MachineType>
                        <Model kb="CUR" kxe="false">22X</Model>
                        <SerialNumber kb="CUR" kxe="false">105A6DB</SerialNumber>
                    </MachineTypeModelAndSerialNumber>
                    <VirtualIOServerLevel kb="CUD" kxe="false">2.2.3.0</VirtualIOServerLevel>
                    <VirtualIOServer kb="CUD" kxe="false"
href="https://9.124.63.53:12443/rest/api/uom/ManagedSystem/b957a114-1b0f-3d45-ac51-49c991a8ac58/
VirtualIOServer/0927928F-BBC2-4686-A391-DB73D6D9DA3B"
rel="related"/>
                </Node>
            </Node>
        </Cluster:Cluster>]]></ParameterValue>
        </JobParameter>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kb="ROR" kxe="false" >sspXml</ParameterName>
            <ParameterValue kxe="false" kb="CUR" >
<![CDATA[<?xml version="1.0" encoding="UTF-8" standalone="yes"?><SharedStoragePool
 xmlns="http://www.ibm.com/xmlns/systems/power/firmware/uom/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0"><Metadata><Atom/></Metadata>
<PhysicalVolumes kb="CUR" kxe="false" schemaVersion="V1_0">
                <Metadata>
                    <Atom/>
                </Metadata>
                <PhysicalVolume schemaVersion="V1_0">
                    <Metadata>
                        <Atom/>
                    </Metadata>
                    <Description kb="CUD" kxe="false">MPIO IBM 2076 FC Disk</Description>
<UniqueDeviceID kxe="false" kb="ROR">3321360050768028087920000000000003704214503IBMfcp</UniqueDeviceID>
                    <VolumeCapacity kb="CUR" kxe="false">20480</VolumeCapacity>
                    <VolumeName kb="CUR" kxe="false">hdisk7</VolumeName>
                    <VolumeState kb="ROR" kxe="false">active</VolumeState>
                    <IsFibreChannelBacked kb="ROR" kxe="false">false</IsFibreChannelBacked>
```

```
            </PhysicalVolume>
        </PhysicalVolumes>
        <Capacity kxe="false" kb="CUR">19.88</Capacity>
        <FreeSpace kxe="false" kb="CUR">19.52</FreeSpace>
        <OverCommitSpace kb="CUR" kxe="false">0.000000</OverCommitSpace>
        <AlertThreshold kb="CUD" kxe="false">35%</AlertThreshold>
        <StoragePoolName kb="CUR" kxe="false">testSP</StoragePoolName></SharedStoragePool>]]
></ParameterValue>
      </JobParameter>
  </JobParameters>
</JobRequest:JobRequest>
```

Related concepts:

"Cluster" on page 52
A Cluster is a set of one or more networked Virtual I/O Server (VIOS) partitions, where each VIOS within the cluster has access to a common set of physical volumes.

Related tasks:

"Shared Storage Pool" on page 54
Shared storage pool contains a logical organization of one or more physical volumes that will be used to provide block storage.

Create_Cluster2 Job

The create cluster job is used to create a cluster with shared storage pool, and with one or more tiers. The details of the cluster, shared storage pool, and the tier are available in an XML format.

Resource

`/rest/api/uom/Cluster/do/Create`

Since: Version 1_3_0

Request

Becaues the cluster and the shared storage pool coexists, a cluster cannot be created without a shared storage pool or vice versa. You must create both the cluster and the shared storage pool at the same time. With this job, every cluster is created with a minimum of one tier and with one or more failure groups.

Table 75. Request Parameters

Request parameter	Description
clusterXml	Cluster details containing node and repository disk information in an XML format.
sspXml	Shared storage pool details containing pool disk information in an XML format.
tierXml	Tier details containing failure group information in XML format.

Response

This job returns valid job status values such as `Complete_Ok` and `Failed_Before_Complete`.

Sample Job Request(s)

```
<JobRequest:JobRequest xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"

xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"

xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_3_0">

    <Metadata>

        <Atom/>
```

```
    </Metadata>
    <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_3_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kb="ROR" kxe="false">Create</OperationName>
        <GroupName kb="ROR" kxe="false">Cluster</GroupName>
    </RequestedOperation>
    <JobParameters kb="CUR" kxe="false" schemaVersion="V1_3_0">
        <Metadata>
            <Atom/>
        </Metadata>
 <JobParameter schemaVersion="V1_3_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kb="ROR" kxe="false" >clusterXml</ParameterName>
<ParameterValue kxe="false" kb="CUR" ><![CDATA[<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Cluster:Cluster
            xmlns:Cluster="http://www.ibm.com/xmlns/systems/power/firmware/uom/mc/2012_10/"
            xmlns="http://www.ibm.com/xmlns/systems/power/firmware/uom/mc/2012_10/"
            xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_3_0">
            <Metadata>
              <Atom/>
            </Metadata>
            <ClusterName kb="COR" kxe="false">testCluster</ClusterName>
            <RepositoryDisk kb="CUD" kxe="false" schemaVersion="V1_3_0">
                <Metadata>
                    <Atom/>
                </Metadata>
                <PhysicalVolume schemaVersion="V1_3_0">
                    <Metadata>
                        <Atom/>
```

```
                    </Metadata>
                    <Description kb="CUD" kxe="false">MPIO DS5100/5300 Disk</Description>
<UniqueDeviceID kb="ROR" kxe="false">01M0lCTTE4MTggICAgICBGQVN0VDYwMEEwQjgwMDA2RTFEMkUwMDAwMjE3
MDUzMzRENkND</UniqueDeviceID>
                    <VolumeCapacity kb="CUR" kxe="false">5120</VolumeCapacity>
                    <VolumeName kb="CUR" kxe="false">hdisk19</VolumeName>
                    <VolumeState kb="ROR" kxe="false">active</VolumeState>
                    <IsFibreChannelBacked kxe="false" kb="ROR">true</IsFibreChannelBacked>
                </PhysicalVolume>
            </RepositoryDisk>
            <Node kxe="false" kb="CUD" schemaVersion="V1_2_0">
    <Metadata>
     <Atom />
    </Metadata>
    <Node schemaVersion="V1_2_0">
     <Metadata>
      <Atom />
     </Metadata>
     <HostName kb="CUD" kxe="false">saturn2vios2.blr.stglabs.ibm.com</HostName>
     <PartitionID kb="CUD" kxe="false">2</PartitionID>
     <MachineTypeModelAndSerialNumber kxe="false" kb="CUD" schemaVersion="V1_2_0">
      <Metadata>
       <Atom />
      </Metadata>
      <MachineType kb="CUR" kxe="false">9117</MachineType>
      <Model kb="CUR" kxe="false">MMB</Model>
      <SerialNumber kb="CUR" kxe="false">06297AP</SerialNumber>
     </MachineTypeModelAndSerialNumber>
     <VirtualIOServerLevel kb="CUD" kxe="false">2.2.4.0</VirtualIOServerLevel>
    </Node>
   </Node>
        </Cluster:Cluster>]]></ParameterValue>
        </JobParameter>
        <JobParameter schemaVersion="V1_3_0">
```

```
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kb="ROR" kxe="false" >sspXml</ParameterName>
<ParameterValue kxe="false" kb="CUR" ><![CDATA[<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<SharedStoragePool
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/uom/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_3_0"><Metadata><Atom/></Metadata>
            <Capacity kxe="false" kb="CUR">19.88</Capacity>
            <FreeSpace kxe="false" kb="CUR">19.52</FreeSpace>
            <OverCommitSpace kb="CUR" kxe="false">0.000000</OverCommitSpace>
            <AlertThreshold kb="CUD" kxe="false">35%</AlertThreshold>
            <StoragePoolName kb="CUR" kxe="false">testSP</StoragePoolName></SharedStoragePool>]]
></ParameterValue>
        </JobParameter>
        <JobParameter schemaVersion="V1_3_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kb="ROR" kxe="false" >tierXml</ParameterName>
<ParameterValue kxe="false" kb="CUR" ><![CDATA[<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Tier xmlns="http://www.ibm.com/xmlns/systems/power/firmware/uom/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_3_0">
       <Metadata>
           <Atom/>
       </Metadata>
    <Name ksv="V1_3_0" kb="CUR" kxe="false">testTier</Name>
       <Type ksv="V1_3_0" kxe="false" kb="CUD">UnrestrictedSystemTier</Type>
       <MirrorState ksv="V1_3_0" kxe="false" kb="ROR">NotMirrored</MirrorState>
    <FailureGroups ksv="V1_3_0" kb="CUD" kxe="false" schemaVersion="V1_3_0">
            <Metadata>
                <Atom/>
            </Metadata>
```

```
            <FailureGroup schemaVersion="V1_3_0">
                <Metadata>
                    <Atom/>
                </Metadata>
      <Name ksv="V1_3_0" kb="CUR" kxe="false">test_FG</Name>
                <PhysicalVolumes ksv="V1_3_0" kxe="false" kb="CUD" schemaVersion="V1_3_0">
                   <Metadata>
                       <Atom/>
                    </Metadata>
                   <PhysicalVolume schemaVersion="V1_3_0">
                       <Metadata>
                           <Atom/>
                       </Metadata>
<UniqueDeviceID kb="ROR" kxe="false">01M0lCTTE4MTggICAgICBGQVN0VDYwMEEwQjgwMDA2
RTFEMkUwMDAwRUM1OTRGRkNFNEIy</UniqueDeviceID>
                       <AvailableForUsage kb="CUD" kxe="false">true</AvailableForUsage>
                       <VolumeCapacity kb="CUR" kxe="false">30720</VolumeCapacity>
                       <VolumeName kb="CUR" kxe="false">hdisk17</VolumeName>
                   </PhysicalVolume>
               </PhysicalVolumes>
           </FailureGroup>
      </FailureGroups>
  </Tier>
  ></ParameterValue>
  </JobParameter>
    </JobParameters>
</JobRequest:JobRequest>
```

Related concepts:

"Cluster" on page 52
A Cluster is a set of one or more networked Virtual I/O Server (VIOS) partitions, where each VIOS within the cluster has access to a common set of physical volumes.

Related tasks:

"Shared Storage Pool" on page 54
Shared storage pool contains a logical organization of one or more physical volumes that will be used to provide block storage.

Delete_Cluster Job

Delete_Cluster Job is used to delete a cluster with Shared Storage Pool (SSP).

Resource

```
/rest/api/uom/Cluster/{Cluster_UUID}/do/Delete
```

Request

This Job does not expect any job parameter.

Response

This job would only return the valid job status values such as Complete_Ok or Failed_Before_Complete.

Sample Job Request(s)

```
<JobRequest:JobRequest
 xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kb="ROR" kxe="false">Delete</OperationName>
        <GroupName kb="ROR" kxe="false">Cluster</GroupName>
    </RequestedOperation>
    <JobParameters kb="CUR" kxe="false" schemaVersion="V1_1_0">
        <Metadata>
            <Atom/>
        </Metadata>
    </JobParameters>
</JobRequest:JobRequest>
```

Related concepts:

"Cluster" on page 52
A Cluster is a set of one or more networked Virtual I/O Server (VIOS) partitions, where each VIOS within the cluster has access to a common set of physical volumes.

Related tasks:

"Shared Storage Pool" on page 54
Shared storage pool contains a logical organization of one or more physical volumes that will be used to provide block storage.

CreateLogicalUnit_Cluster Job

CreateLogicalUnit_Cluster Job is used to create a logical unit (LU) in a cluster or Shared Storage Pool (SSP).

Resource

```
/rest/api/uom/Cluster/{Cluster_UUID}/do/CreateLogicalUnit
```

Request

The LU details such as name, type, device type, and size are required. Also, you can mention the cloning details to create a clone LU.

Table 76. Request parameters

Request parameter	Description
TierUDID	The value of Tier UDID to be left blank as default.
LUName	Name of the LU to be created.
LUSize	Size of the LU to be created in GB.
LUType	Type of provisioning allowed. This can take only THICK / THIN as value.
DeviceType	Type of LU to be created. This can take only VirtualIO_Disk / VirtualIO_Image as value.
ClonedFrom	UDID of LU from which newly created LU is cloned.

Response

This job would return UDID of newly created Logical unit and also the valid Job status like Complete_Ok, Failed_Before_Complete and so on.

Table 77. Response params

Response param	Description
LUCreated	UDID of logical unit of newly created Logical Unit

Sample Job Request(s)

```
<JobRequest:JobRequest
xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_1_0">
   <Metadata>
       <Atom/>
   </Metadata>
   <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_1_0">
       <Metadata>
           <Atom/>
       </Metadata>
       <OperationName kb="ROR" kxe="false">CreateLogicalUnit</OperationName>
       <GroupName kb="ROR" kxe="false">Cluster</GroupName>
   </RequestedOperation>
   <JobParameters kb="CUR" kxe="false" schemaVersion="V1_1_0">
       <Metadata>
           <Atom/>
       </Metadata>
       <JobParameter schemaVersion="V1_0">
     <Metadata>
         <Atom/>
     </Metadata>
     <ParameterName kb="ROR" kxe="false">TierUDID</ParameterName>
     <ParameterValue kxe="false" kb="CUR">2548dd0c98b66211e3840f0000c9f843185194ee2ce6ef2ce6
   </ParameterValue>
 </JobParameter>
 <JobParameter schemaVersion="V1_0">
     <Metadata>
         <Atom/>
     </Metadata>
     <ParameterName kb="ROR" kxe="false">LUName</ParameterName>
     <ParameterValue kxe="false" kb="CUR">newLU</ParameterValue>
 </JobParameter>
 <JobParameter schemaVersion="V1_0">
     <Metadata>
         <Atom/>
     </Metadata>
```

```
      <ParameterName kb="ROR" kxe="false">LUSize</ParameterName>
      <ParameterValue kxe="false" kb="CUR">18</ParameterValue>
  </JobParameter>
  <JobParameter schemaVersion="V1_0">
      <Metadata>
          <Atom/>
      </Metadata>
      <ParameterName kb="ROR" kxe="false">LUType</ParameterName>
      <ParameterValue kxe="false" kb="CUR">THIN</ParameterValue>
  </JobParameter>
  <JobParameter schemaVersion="V1_0">
      <Metadata>
          <Atom/>
      </Metadata>
      <ParameterName kb="ROR" kxe="false">DeviceType</ParameterName>
      <ParameterValue kxe="false" kb="CUR">VirtualIO_Disk</ParameterValue>
  </JobParameter>
  <JobParameter schemaVersion="V1_0">
      <Metadata>
          <Atom/>
      </Metadata>
      <ParameterName kb="ROR" kxe="false">ClonedFrom</ParameterName>
      <ParameterValue kxe="false" kb="CUR">276f4ba9fcd40dc1eecedef802d6172d27</ParameterValue>
  </JobParameter>
    </JobParameters>
</JobRequest:JobRequest>
```

Related concepts:

"Cluster" on page 52
A Cluster is a set of one or more networked Virtual I/O Server (VIOS) partitions, where each VIOS within the cluster has access to a common set of physical volumes.

Related tasks:

"Shared Storage Pool" on page 54
Shared storage pool contains a logical organization of one or more physical volumes that will be used to provide block storage.

"Logical Unit" on page 58
Logical unit is a file in the pool that provides file-backed storage for Power systems.

DeleteLogicalUnit_Cluster Job

The DeleteLogicalUnit_Cluster job is used to remove or delete a logical unit from the cluster or a shared storage pool.

Resource

```
/rest/api/uom/Cluster/{Cluster_UUID}/do/DeleteLogicalUnit
```

Since: Version 1_1_0

Request

The details about the logical unit that needs to be deleted must be specified.

Table 78. Request parameters

Request parameter	Description
LogicalUnitUDID	The unique device identifier (UDID) value of the logical unit that needs to be deleted.

Response

This job returns valid job status values such as `Complete_Ok` and `Failed_Before_Complete`.

Sample Job Request(s)

```
<JobRequest:JobRequest

xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"

xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"

xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_3_0">

    <Metadata>

        <Atom/>

    </Metadata>

    <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_3_0">

        <Metadata>

            <Atom/>

        </Metadata>

        <OperationName kb="ROR" kxe="false">DeleteLogicalUnit</OperationName>

        <GroupName kxe="false" kb="ROR">Cluster</GroupName>

    </RequestedOperation>

    <JobParameters kb="CUR" kxe="false" schemaVersion="V1_3_0">

        <Metadata>

            <Atom/>

        </Metadata>

   <JobParameter schemaVersion="V1_3_0">

            <Metadata>

                <Atom/>

            </Metadata>

            <ParameterName kb="ROR" kxe="false" >LogicalUnitUDID</ParameterName>

<ParameterValue kxe="false" kb="CUR" >27fbe6472e0de511e5a52e40f2e91330242644023a

5eb8be48c1cfb278e9164a1b</ParameterValue>

        </JobParameter>

    </JobParameters>

</JobRequest:JobRequest>
```

Related concepts:

"Cluster" on page 52

A Cluster is a set of one or more networked Virtual I/O Server (VIOS) partitions, where each VIOS within the cluster has access to a common set of physical volumes.

Related tasks:

"Shared Storage Pool" on page 54
Shared storage pool contains a logical organization of one or more physical volumes that will be used to provide block storage.

"Logical Unit" on page 58
Logical unit is a file in the pool that provides file-backed storage for Power systems.

MigrateLogicalUnit_Cluster Job

The MigrateLogicalUnit_Cluster job is used to move a logical unit from one tier to another tier within the same cluster.

Resource

`/rest/api/uom/Cluster/{Cluster_UUID}/do/MoveLogicalUnit`

Since: Version 1_3_0

Request

The details of the logical unit that needs to be migrated and details of the target tier must be specified.

Table 79. Request parameters

Request parameter	Description
LogicalUnitUDID	The unique device identifier (UDID) value of the logical unit that needs to be migrated.
TargetTierUUID	The UUID of the target tier to which the logical unit needs to be migrated.
IsRecursive	When the value of this parameter is `true`, the entire tree of derived logical units including input logical unit is moved. When the value is `false`, only the input logical unit is moved.

Response

This job returns the valid job status values such as `Complete_Ok` and `Failed_Before_Complete`.

Sample Job Request(s)

```
<JobRequest:JobRequest

xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"

xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"

xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_1_0">

    <Metadata>

        <Atom/>

    </Metadata>

    <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_1_0">

        <Metadata>

            <Atom/>

        </Metadata>
```

```
        <OperationName kb="ROR" kxe="false">MoveLogicalUnit</OperationName>
        <GroupName kb="ROR" kxe="false">Cluster</GroupName>
    </RequestedOperation>
    <JobParameters kb="CUR" kxe="false" schemaVersion="V1_1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <JobParameter schemaVersion="V1_0">
      <Metadata>
          <Atom/>
      </Metadata>
      <ParameterName kb="ROR" kxe="false">LogicalUnitUDID</ParameterName>
<ParameterValue kxe="false" kb="CUR">27fbe6472e0de511e5a52e40f2e91330242644023a
5eb8be48c1cfb278e9164a1b</ParameterValue>
  </JobParameter>
  <JobParameter schemaVersion="V1_0">
      <Metadata>
          <Atom/>
      </Metadata>
      <ParameterName kb="ROR" kxe="false">TargetTierUUID</ParameterName>
 <ParameterValue kxe="false" kb="CUR">191c94bb-6a7e-3c5a-bce6-1c7d558e0923</ParameterValue>
  </JobParameter>
    </JobParameters>
</JobRequest:JobRequest>
```

Related concepts:

"Cluster" on page 52
A Cluster is a set of one or more networked Virtual I/O Server (VIOS) partitions, where each VIOS within the cluster has access to a common set of physical volumes.

Related tasks:

"Shared Storage Pool" on page 54
Shared storage pool contains a logical organization of one or more physical volumes that will be used to provide block storage.

"Logical Unit" on page 58
Logical unit is a file in the pool that provides file-backed storage for Power systems.

LULinkedClone_Cluster Job

Logical Unit Clone job is used to create a clone of link clone of logical units.

Resource

```
/rest/api/uom/Cluster/{Cluster_UUID}/do/LULinkedClone
```

Request

The UDID of Source and Destination Logical Unit are required to create a link clone. HMC does not support creating a full clone of logical units.

Table 80. Request Parameters

Request Parameters	Description
SourceUDID	UDID of logical unit from which clone has to be performed.
DestinationUDID	UDID of logical unit on which clone will be performed.

Response

This job would only return the valid Job status like Complete_Ok, Failed_Before_Complete and so on.

Sample Job Request(s)

```
<JobRequest
 xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kxe="false" kb="ROR">LULinkedClone</OperationName>
        <GroupName kxe="false" kb="ROR">Cluster</GroupName>
        <ProgressType kxe="false" kb="ROR">DISCRETE</ProgressType>
    </RequestedOperation>
    <JobParameters kb="CUR" kxe="false" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
  <JobParameter schemaVersion="V1_0">
      <Metadata>
          <Atom/>
      </Metadata>
      <ParameterName kb="ROR" kxe="false">SourceUDID</ParameterName>
      <ParameterValue kxe="false" kb="CUR">276f4ba9fcd40dc1eecedef802d6172d27</ParameterValue>
  </JobParameter>
  <JobParameter schemaVersion="V1_0">
      <Metadata>
          <Atom/>
      </Metadata>
      <ParameterName kb="ROR" kxe="false">DestinationUDID</ParameterName>
      <ParameterValue kxe="false" kb="CUR">27c0fae17fca3336fdcedaefcbcad6541d</ParameterValue>
  </JobParameter>
    </JobParameters>
</JobRequest>
```

Related concepts:

"Cluster" on page 52
A Cluster is a set of one or more networked Virtual I/O Server (VIOS) partitions, where each VIOS within the cluster has access to a common set of physical volumes.

Related tasks:

"Shared Storage Pool" on page 54
Shared storage pool contains a logical organization of one or more physical volumes that will be used to provide block storage.

"Logical Unit" on page 58
Logical unit is a file in the pool that provides file-backed storage for Power systems.

Cluster_Replace_Repository_Disk Job

The Cluster_Replace_Repository_Disk job is used to replace the existing repository disk of the cluster with a new one.

Resource

```
/rest/api/uom/Cluster/{Cluster_UUID}/do/ReplaceRepository
```

Request

This job has only one parameter which is the marshalled XML of the new repository disk.

Table 81. List of Job parameters

Request param	Description
REPO_DISK	Details of the new repository disk

Response

This job returns valid Job status values such as Complete_Ok and Failed_Before_Complete.

Sample Job Request(s)

```
<JobRequest

xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"

xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">

    <Metadata>

        <Atom/>

    </Metadata>

    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">

        <Metadata>

            <Atom/>

        </Metadata>

        <OperationName kxe="false" kb="ROR">ReplaceRepository</OperationName>

        <GroupName kxe="false" kb="ROR">Cluster</GroupName>

        <ProgressType kxe="false" kb="ROR">DISCRETE</ProgressType>

    </RequestedOperation>

    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">

        <Metadata>

            <Atom/>
```

```
        </Metadata>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">REPO_DISK</ParameterName>
<ParameterValue kxe="false" kb="CUR"><![CDATA[<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<PhysicalVolume xmlns="http://www.ibm.com/xmlns/systems/power/firmware/uom/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_3_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <Description kb="CUD" kxe="false">MPIO DS5100/5300 Disk</Description>
<UniqueDeviceID kb="ROR" kxe="false">01M0lCTTE4MTggICAgICBGQVN0VDYwMEEwQjgwMDA
2RTFEMkUwMDAwQTM0RDU0NjQwMTJB</UniqueDeviceID>
 <VolumeCapacity kb="CUR" kxe="false">512</VolumeCapacity>
 <VolumeName kb="CUR" kxe="false">hdisk14</VolumeName>
 <VolumeState kb="ROR" kxe="false">active</VolumeState>
 <IsFibreChannelBacked kb="ROR" kxe="false">true</IsFibreChannelBacked>
</PhysicalVolume>]]</ParameterValue>
        </JobParameter>
    </JobParameters>
</JobRequest>
```

Related concepts:

"Cluster" on page 52
A Cluster is a set of one or more networked Virtual I/O Server (VIOS) partitions, where each VIOS within the cluster has access to a common set of physical volumes.

Related tasks:

"Shared Storage Pool" on page 54
Shared storage pool contains a logical organization of one or more physical volumes that will be used to provide block storage.

GetFreePhysicalVolumes_Cluster Job

The GetFreePhysicalVolumes_Cluster job returns the free physical volumes that can be added to the cluster.

Resource

```
/rest/api/uom/Cluster/{Cluster_UUID}/do/GetFreePhysicalVolumes
```

Since: Version 1_3_0

Request

This job does not need any job parameters.

Response

This job returns the **PhysicalVolumeCollection** that contains the free physical volumes that can be added to the cluster on which it was queried. It also returns valid job status values such as `Complete_Ok` and `Failed_Before_Complete`.

Table 82. Response parameters

Response parameters	Description
result	**PhysicalVolumeCollection** that contains free physical volumes that can be added to the cluster.

Sample Job Request(s)

```
<JobRequest:JobRequest xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"

xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"

xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_3_0">

    <Metadata>

        <Atom/>

    </Metadata>

    <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_3_0">

        <Metadata>

            <Atom/>

        </Metadata>

        <OperationName kb="ROR" kxe="false">GetFreePhysicalVolumes</OperationName>

        <GroupName kb="ROR" kxe="false">VirtualIOServer</GroupName>

    </RequestedOperation>

    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_3_0">

        <Metadata>

            <Atom/>

        </Metadata>

  </JobParameter>

    </JobParameters>

</JobRequest:JobRequest>
```

Related concepts:

“Cluster” on page 52
A Cluster is a set of one or more networked Virtual I/O Server (VIOS) partitions, where each VIOS within the cluster has access to a common set of physical volumes.

ManagePhysicalVolume_Cluster Job

The ManagePhysicalVolume_Cluster job allows you to add, remove, or replace the physical volumes that are associated to the failure group of the specified tier.

Resource

`/rest/api/uom/Cluster/{Cluster_UUID}/do/ManagePhysicalVolume`

Since: Version 1_3_0

Request

The job request must contain either of **TIER_UDID** parameter or the **TIER_UUID** parameter, and the **FAILURE_GROUP_UDID** parameter is mandatory. If the **PhysicalVolumeCollection** parameter is located in the **PVS_TO_BE_ADDED** parameter, then those physical volumes are added to the specified failure group. If the **PhysicalVolumeCollection** parameter is specified under **PVS_TO_BE_REMOVED** parameter, then those physical volumes are removed from the specified failure group. If both **PVS_TO_BE_ADDED** parameter and the **PVS_TO_BE_REMOVED** parameter are specified, then the physical volumes are replaced in the specified failure group.

Table 83. Request parameters

Request parameter	Description
TIER_UDID	The value of tier UDID.
FAILURE_GROUP_UDID	The value of failure group UDID.
PVS_TO_BE_ADDED	List of physical volumes that must be added.
PVS_TO_BE_REMOVED	List of physical volumes that must be removed.

Response

This job returns the valid job status such as `Complete_Ok` and `Failed_Before_Complete`.

Sample Job Request(s)

```
<JobRequest:JobRequest

xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"

xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"

xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_3_0">

    <Metadata>

        <Atom/>

    </Metadata>

    <RequestedOperation kb="CUR" kxe="false" schemaVersion="V1_3_0">

        <Metadata>

            <Atom/>

        </Metadata>
```

```
        <OperationName kb="ROR" kxe="false">GetFreePhysicalVolumes</OperationName>
        <GroupName kb="ROR" kxe="false">VirtualIOServer</GroupName>
    </RequestedOperation>
    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_3_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <JobParameter schemaVersion="V1_3_0">
      <Metadata>
          <Atom/>
      </Metadata>
      <ParameterName kb="ROR" kxe="false">TIER_UDID</ParameterName>
<ParameterValue kxe="false" kb="CUR">256f140d7631f411e59866e41f139ec
750b3720f5ee4867db8</ParameterValue>
  </JobParameter>
  <JobParameter schemaVersion="V1_3_0">
      <Metadata>
          <Atom/>
      </Metadata>
      <ParameterName kb="ROR" kxe="false">TIER_UUID</ParameterName>
<ParameterValue kxe="false" kb="CUR">94a23340-ec8d-3f7b-8e7e-9d9feb046d23</ParameterValue>
  </JobParameter>
  <JobParameter schemaVersion="V1_3_0">
      <Metadata>
          <Atom/>
      </Metadata>
     <ParameterName kb="ROR" kxe="false">FAILURE_GROUP_UDID</ParameterName>
<ParameterValue kxe="false" kb="CUR">266f140d7631f411e59866e41f139ec75032686d962ce51c29
</ParameterValue>
  </JobParameter>
  <JobParameter schemaVersion="V1_3_0">
      <Metadata>
          <Atom/>
```

```
      </Metadata>
      <ParameterName kb="ROR" kxe="false">PVS_TO_BE_ADDED</ParameterName>
<ParameterValue kxe="false" kb="CUR"><![CDATA[<PhysicalVolume_Collection schemaVersion="V1_3_0">
      <PhysicalVolume schemaVersion="V1_3_0">
                                <Metadata>
                                    <Atom/>
                                </Metadata>
                            <Description kb="CUD" kxe="false">MPIO DS5100/5300 Disk</Description>
<UniqueDeviceID kb="ROR" kxe="false">01M0lCTTE4MTggICAgICBGQVN0VDYwMEEwQjgwMDA
2RTE5OTIwMDAwMzIxNzUxMkI3NTY3</UniqueDeviceID>
                                <VolumeCapacity kb="CUR" kxe="false">51200</VolumeCapacity>
                                <VolumeName kb="CUR" kxe="false">hdisk10</VolumeName>
                                <VolumeState kb="ROR" kxe="false">active</VolumeState>
                           <IsFibreChannelBacked kxe="false" kb="ROR">true</IsFibreChannelBacked>
<StorageLabel ksv="V1_3_0" kb="ROR" kxe="false">cGFnaW5nX2Rpc2tfNTBfMTI=</StorageLabel>
                             </PhysicalVolume>
      <PhysicalVolume schemaVersion="V1_3_0">
                                <Metadata>
                                    <Atom/>
                                </Metadata>
                        <Description kb="CUD" kxe="false">MPIO DS5100/5300 Disk</Description>
<UniqueDeviceID kb="ROR" kxe="false">01M0lCTTE4MTggICAgICBGQVN0VDYwMEEwQjgwMDA
2RTE5OTIwMDAwMzIxNzUxMkI3NTY4</UniqueDeviceID>
                                <VolumeCapacity kb="CUR" kxe="false">51200</VolumeCapacity>
                                <VolumeName kb="CUR" kxe="false">hdisk11</VolumeName>
                                <VolumeState kb="ROR" kxe="false">active</VolumeState>
                             <IsFibreChannelBacked kxe="false" kb="ROR">true</IsFibreChannelBacked>
            <StorageLabel ksv="V1_3_0" kb="ROR" kxe="false">cGFnaW5nX2Rpc2tfNTBfMTI=</StorageLabel>
                              </PhysicalVolume>
      </PhysicalVolume_Collection>]]
></ParameterValue>
  </JobParameter>
  <JobParameter schemaVersion="V1_3_0">
      <Metadata>
```

```
          <Atom/>
      </Metadata>
      <ParameterName kb="ROR" kxe="false">PVS_TO_BE_REMOVED</ParameterName>
<ParameterValue kxe="false" kb="CUR"><![CDATA[<PhysicalVolume_Collection schemaVersion="V1_3_0">
      <PhysicalVolume schemaVersion="V1_3_0">
                                <Metadata>
                                    <Atom/>
                                </Metadata>
                            <Description kb="CUD" kxe="false">MPIO DS5100/5300 Disk</Description>
<UniqueDeviceID kb="ROR" kxe="false">01M0lCTTE4MTggICAgICBGQVN0VDYwMEEwQjgwMDA
2RTE5OTIwMDAwMzIxNzUxMkI3NTY5</UniqueDeviceID>
                                <VolumeCapacity kb="CUR" kxe="false">51200</VolumeCapacity>
                                <VolumeName kb="CUR" kxe="false">hdisk12</VolumeName>
                                <VolumeState kb="ROR" kxe="false">active</VolumeState>
                                <IsFibreChannelBacked kxe="false" kb="ROR">true</IsFibreChannelBacked>
                <StorageLabel ksv="V1_3_0" kb="ROR" kxe="false">cGFnaW5nX2Rpc2tfNTBfMTI=</StorageLabel>
                            </PhysicalVolume>
      <PhysicalVolume schemaVersion="V1_3_0">
                                <Metadata>
                                    <Atom/>
                                </Metadata>
                                <Description kb="CUD" kxe="false">MPIO DS5100/5300 Disk</Description>
<UniqueDeviceID kb="ROR" kxe="false">01M0lCTTE4MTggICAgICBGQVN0VDYwMEEwQjgwMDA
2RTE5OTIwMDAwMzIxNzUxMkI3NTY6</UniqueDeviceID>
                                <VolumeCapacity kb="CUR" kxe="false">51200</VolumeCapacity>
                                <VolumeName kb="CUR" kxe="false">hdisk13</VolumeName>
                                <VolumeState kb="ROR" kxe="false">active</VolumeState>
                                <IsFibreChannelBacked kxe="false" kb="ROR">true</IsFibreChannelBacked>
                <StorageLabel ksv="V1_3_0" kb="ROR" kxe="false">cGFnaW5nX2Rpc2tfNTBfMTI=</StorageLabel>
                            </PhysicalVolume>
      </PhysicalVolume_Collection>]]></ParameterValue>
  </JobParameter>
```

```
    </JobParameters>

</JobRequest:JobRequest>
```

Related concepts:

"Cluster" on page 52
A Cluster is a set of one or more networked Virtual I/O Server (VIOS) partitions, where each VIOS within the cluster has access to a common set of physical volumes.

ClearStatistics_SRIOVEthernetLogicalPort Job

Operation to clear Statistical data available on a particular SRIOV Ethernet Logical Port of Logical Partition / VirtualIOServer.

Resource

```
/rest/api/uom/LogicalPartition/{LogicalPartition_UUID}/SRIOVEthernetLogicalPort/
{SRIOVEthernetLogicalPort_UUID}/do/ClearStatistics
```

```
/rest/api/uom/VirtualIOServer/{VirtualIOServer_UUID}/SRIOVEthernetLogicalPort/
{SRIOVEthernetLogicalPort_UUID}/do/ClearStatistics
```

Overview of ClearStatistics_SRIOVEthernetLogicalPort Job

For a ManagedSystem which has SRIOV Adapter Card installed on it, we can configure one or more SRIOV Ethernet Logical Ports to a LogicalPartition. For each of these configured Ethernet Logical Ports, we can invoke this job to reset the statistics maintained regarding its Usage/Traffic-flow.

Sample Job Request(s)

```
  <JobRequest:JobRequest
 xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
     <Metadata>
         <Atom/>
     </Metadata>
     <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
         <Metadata>
             <Atom/>
         </Metadata>
         <OperationName kxe="false" kb="ROR">ClearStatistics</OperationName>
         <GroupName kxe="false" kb="ROR">SRIOVEthernetLogicalPort</GroupName>
     </RequestedOperation>
     <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
         <Metadata>
             <Atom/>
         </Metadata>
     </JobParameters>
 </JobRequest:JobRequest>
```

ClearStatistics_SRIOVFibreChannelOverEthernetLogicalPort Job

Operation to clear Statistical data available on a particular SRIOV FibreChannelOverEthernet Logical Port of Logical Partition / VirtualIOServer.

Resource

```
/rest/api/uom/LogicalPartition/{LogicalPartition_UUID}/SRIOVFibreChannelOverEthernetLogicalPort
/{SRIOVFibreChannelOverEthernetLogicalPort_UUID}/do/ClearStatistics
```

```
/rest/api/uom/VirtualIOServer/{VirtualIOServer_UUID}/SRIOVFibreChannelOverEthernetLogicalPort
/{SRIOVFibreChannelOverEthernetLogicalPort_UUID}/do/ClearStatistics
```

Overview of ClearStatistics_SRIOVFibreChannelOverEthernetLogicalPort Job

For a ManagedSystem which has SRIOV Adapter Card installed on it, we can configure one or more SRIOV FibreChannelOverEthernet Logical Ports to a LogicalPartition. For each of these configured FibreChannelOverEthernet Logical Ports, we can invoke this job to reset the statistics maintained regarding its Usage/Traffic-flow.

Sample Job Request(s)

```
 <JobRequest:JobRequest
xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kxe="false" kb="ROR">ClearStatistics</OperationName>
        <GroupName kxe="false" kb="ROR">SRIOVFibreChannelOverEthernetLogicalPort</GroupName>
    </RequestedOperation>
    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
    </JobParameters>
</JobRequest:JobRequest>
```

CreatePowerEnterprisePool_PowerEnterprisePool Job

Creates a Capacity on Demand (CoD) Power enterprise pool.

Resource

```
/rest/api/uom/PowerEnterprisePool/do/CreatePowerEnterprisePool
```

Request

Invoke this job to create a Capacity on Demand (CoD) Power enterprise pool.

Table 84. Request Parameters

Request param	Description
FileID	File ID number for the pool configuration file after the file has been uploaded to the HMC.
PoolName	User specified pool name for the Power enterprise pool.
HMCList	List of Hardware Management Console that has to be part being created

Response

This job would return the job id to be used to poll the job status.

Sample Job Request(s)

Sample Job Request Example 1: Since Version 1_4_0.

```
<JobRequest:JobRequest
   xmlns:JobRequest="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
   xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
   xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_3_0">
   <Metadata>
       <Atom />
   </Metadata>
   <RequestedOperation kb="CUR" kxe="false"
       schemaVersion="V1_3_0">
       <Metadata>
           <Atom />
       </Metadata>
       <OperationName kb="ROR" kxe="false">CreatePowerEnterprisePool
       </OperationName>
       <GroupName kb="ROR" kxe="false">PowerEnterprisePool</GroupName>
   </RequestedOperation>
   <JobParameters kxe="false" kb="CUR" schemaVersion="V1_3_0">
       <Metadata>
           <Atom />
       </Metadata>
       <JobParameter schemaVersion="V1_3_0">
           <Metadata>
               <Atom />
           </Metadata>
           <ParameterName kxe="false" kb="ROR">FileID</ParameterName>
           <ParameterValue kxe="false" kb="CUR">codPool_02C3_signed1.xml
           </ParameterValue>
       </JobParameter>
       <JobParameter schemaVersion="V1_3_0">
           <Metadata>
               <Atom />
           </Metadata>
           <ParameterName kxe="false" kb="ROR">PoolName</ParameterName>
           <ParameterValue kxe="false" kb="CUR">codPool_02C3
```

```
            </ParameterValue>
        </JobParameter>
        <JobParameter schemaVersion="V1_3_0">
            <Metadata>
                <Atom />
            </Metadata>
            <ParameterName kxe="false" kb="ROR">HMCList</ParameterName>
            <ParameterValue kxe="false" kb="CUR"><![CDATA[<PowerEnterprisePoolManagementConsole_Collection
xmlns:PowerEnterprisePoolManagementConsole_Collection=
"http://www.ibm.com/xmlns/systems/power/firmware/uom/mc/2012_10/"
xmlns="http://www.ibm.com/xmlns/systems/power/firmware/uom/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_3_0">
        <Metadata>
        <Atom/>
        </Metadata>
        <PowerEnterprisePoolManagementConsole schemaVersion="V1_3_0">
        <Metadata>
        <Atom/>
        </Metadata>
        <ManagementConsoleName kxe="false" kb="UOR">HMC194</ManagementConsoleName>
        <ManagementConsoleMachineTypeModelSerialNumber kb="UOR" kxe="false" schemaVersion="V1_3_0">
        <Metadata>
        <Atom/>
        </Metadata>
        <MachineType kxe="false" kb="CUR">V637</MachineType>
        <Model kb="CUR" kxe="false">f8d</Model>
        <SerialNumber kxe="false" kb="CUR">9d1efad</SerialNumber>
        </ManagementConsoleMachineTypeModelSerialNumber>
        <IsMasterConsole kxe="false" kb="UOR">false</IsMasterConsole>
        <IsBackupMasterConsole kb="UOR" kxe="false">false</IsBackupMasterConsole>
        <ManagementConsoleIPAddress ksv="V1_4_0" kb="COD" kxe="false">9.124.63.182
</ManagementConsoleIPAddress>
        <ManagementConsoleUserID ksv="V1_4_0" kxe="false" kb="COR">hscroot</ManagementConsoleUserID>
```

```
        <ManagementConsolePassword ksv="V1_4_0" kxe="false" kb="COR">abc123</ManagementConsolePassword>
        </PowerEnterprisePoolManagementConsole>
        <PowerEnterprisePoolManagementConsole schemaVersion="V1_3_0">
        <Metadata>
        <Atom/>
        </Metadata>
        <ManagementConsoleName kxe="false" kb="UOR">HMC194</ManagementConsoleName>
        <ManagementConsoleMachineTypeModelSerialNumber kb="UOR" kxe="false" schemaVersion="V1_3_0">
        <Metadata>
        <Atom/>
        </Metadata>
        <MachineType kxe="false" kb="CUR">V637</MachineType>
        <Model kb="CUR" kxe="false">f8d</Model>
        <SerialNumber kxe="false" kb="CUR">9d1efad</SerialNumber>
        </ManagementConsoleMachineTypeModelSerialNumber>
        <IsMasterConsole kxe="false" kb="UOR">false</IsMasterConsole>
        <IsBackupMasterConsole kb="UOR" kxe="false">false</IsBackupMasterConsole>
        <ManagementConsoleIPAddress ksv="V1_4_0" kb="COD" kxe="false">9.124.63.182
</ManagementConsoleIPAddress>
        <ManagementConsoleUserID ksv="V1_4_0" kxe="false" kb="COR">hscroot</ManagementConsoleUserID>
        <ManagementConsolePassword ksv="V1_4_0" kxe="false" kb="COR">abc123</ManagementConsolePassword>
        </PowerEnterprisePoolManagementConsole>
        </PowerEnterprisePoolManagementConsole_Collection>
</ParameterValue>
        </JobParameter>
    </JobParameters>
</JobRequest:JobRequest>
```

Sample Job Request Example 2: Prior to 1_4_0.

```
<JobRequest xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
 xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
```

```
        </Metadata>
        <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <OperationName kxe="false" kb="ROR">CreatePowerEnterprisePool</OperationName>
            <GroupName kxe="false" kb="ROR">PowerEnterprisePool</GroupName>
            <ProgressType kxe="false" kb="ROR">LINEAR</ProgressType>
        </RequestedOperation>
        <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <JobParameter schemaVersion="V1_0">
                <Metadata>
                    <Atom/>
                </Metadata>
                <ParameterName kxe="false" kb="ROR">FileID</ParameterName>
                <ParameterValue kxe="false" kb="CUR">12345678</ParameterValue>
            </JobParameter>
            <JobParameter schemaVersion="V1_0">
                <Metadata>
                    <Atom/>
                </Metadata>
                <ParameterName kxe="false" kb="ROR">PoolName</ParameterName>
                <ParameterValue kxe="false" kb="CUR">SamplePoolName</ParameterValue>
            </JobParameter>
        </JobParameters>
    </JobRequest>
```

SetMasterConsole_PowerEnterprisePool Job

Change the master management console for a Power enterprise pool.

Resource

```
/rest/api/uom/PowerEnterprisePool/do/SetMasterConsole
```

Request

Invoke this job to change the master management console for a Power enterprise pool.

Table 85. Request Parameters

Request param	Description
ManagementConsoleIPAddress	Provide IP Address of the management console to set it as master.
Force	The valid values are true or false. If this parameter is not set by default force parameter will be false.

Response

This job would return the job id to be used to poll the Job's status.

Sample Job Request(s)

```
  <JobRequest xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
     <Metadata>
         <Atom/>
     </Metadata>
     <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
         <Metadata>
             <Atom/>
         </Metadata>
         <OperationName kxe="false" kb="ROR">SetMasterConsole</OperationName>
         <GroupName kxe="false" kb="ROR">PowerEnterprisePool</GroupName>
         <ProgressType kxe="false" kb="ROR">LINEAR</ProgressType>
     </RequestedOperation>
     <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
         <Metadata>
             <Atom/>
         </Metadata>
         <JobParameter schemaVersion="V1_0">
             <Metadata>
                 <Atom/>
             </Metadata>
             <ParameterName kxe="false" kb="ROR">ManagementConsoleIPAddress</ParameterName>
             <ParameterValue kxe="false" kb="CUR">9.124.63.171</ParameterValue>
```

```
        </JobParameter>
        <JobParameter schemaVersion="V1_0">
               <Metadata>
                   <Atom/>
               </Metadata>
               <ParameterName kxe="false" kb="ROR">Force</ParameterName>
               <ParameterValue kxe="false" kb="CUR">true</ParameterValue>
           </JobParameter>
    </JobParameters>
</JobRequest>
```

SyncSystemPool_PowerEnterprisePool Job

Synchronize the pool information between the master management console for the pool and the systems in the pool.

Resource

```
/rest/api/uom/PowerEnterprisePool/do/SyncSystemPool
```

Request

Invoke this job to synchronize the pool information between the master management console for the pool and the systems in the pool.

Response

This job would return the job id to be used to poll the Job status.

Sample Job Request(s)

```
  <JobRequest xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kxe="false" kb="ROR">SyncSystemPool</OperationName>
        <GroupName kxe="false" kb="ROR">PowerEnterprisePool</GroupName>
        <ProgressType kxe="false" kb="ROR">LINEAR</ProgressType>
    </RequestedOperation>
    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
    </JobParameters>
</JobRequest>
```

UpdatePowerEnterprisePool_PowerEnterprisePool Job

Update a Power enterprise pool with the specified pool configuration file.

Resource

/rest/api/uom/PowerEnterprisePool/do/UpdatePowerEnterprisePool

Request

Invoke this job to update a power enterprise pool with the specified pool configuration file.

Table 86. Request Parameters

Request param	Description
FileID	File ID number for the pool configuration file after the file has been uploaded to the HMC.

Response

This job would return the job id to be used to poll the Job's status.

Sample Job Request(s)

```
 <JobRequest xmlns="http://www.ibm.com/xmlns/systems/power/firmware/web/mc/2012_10/"
xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_0">
    <Metadata>
        <Atom/>
    </Metadata>
    <RequestedOperation kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <OperationName kxe="false" kb="ROR">UpdatePowerEnterprisePool</OperationName>
        <GroupName kxe="false" kb="ROR">PowerEnterprisePool</GroupName>
        <ProgressType kxe="false" kb="ROR">LINEAR</ProgressType>
    </RequestedOperation>
    <JobParameters kxe="false" kb="CUR" schemaVersion="V1_0">
        <Metadata>
            <Atom/>
        </Metadata>
        <JobParameter schemaVersion="V1_0">
            <Metadata>
                <Atom/>
            </Metadata>
            <ParameterName kxe="false" kb="ROR">FileID</ParameterName>
            <ParameterValue kxe="false" kb="CUR">12345678</ParameterValue>
        </JobParameter>
    </JobParameters>
</JobRequest>
```

Job status

When a Job is invoked, a status is returned. The status provides information about the result of the Job and other related details.

Table provides information about the different job statuses along with a description of each status.

Table 87. List of Job statuses.

Job status	Description
CANCELED_BEFORE_START	The **Cancel** command was received and was processed successfully before the Job moved out of the NOT_STARTED state.
CANCELED_WHILE_RUNNING	A **Cancel** command was received and processed successfully while the Job was in the RUNNING state.

Table 87. List of Job statuses (continued).

Job status	Description
COMPLETED_WITH_WARNINGS	The Job completed with partial success, with some subtasks completed successfully, and some others failed. Manual intervention is required either to complete the intended task or to revert the system to its original state. For further details, see the JobResponse.Progress.CompletedTasks for the Job.
COMPLETED_WITH_ERROR	The Job ended naturally. However, due to one of the following conditions listed below, the Job was not moved to COMPLETED_OK state. • Some precondition was not met and the Job should not be attempted again with the system in the current state • The Job would need to be rerun in its entirety due to an error, possibly after manual cleanup For further details, see the JobResponse.Progress.CompletedTasks for the Job.
FAILED_TO_START	The Job could not be started due to an unexpected internal error in the Job processing.
FAILED_BEFORE_COMPLETION	The Job did not complete. This usually means that the Job method threw an exception or was interrupted, without the **Cancel** command.
NOT_STARTED	The Job has not yet been initiated by the framework. **Note:** If a Job remains in this state for a significant amount of time, you might want to check what is causing the delay in initiating the Job.
RUNNING	The Job is in progress.
FAILED_BEFORE_COMPLETION_RETRY	The Job cannot perform the requested operation. The virtual I/O server (VIOS) is busy performing other tasks. Try the operation after some time.
COMPLETED_OK	The Job completed successfully and subsequent user intervention is not required. **Note:** The Job might still have issued warnings and there might still be some minor cleanup required. For further details, see the JobResponse.Progress.CompletedTasks for the Job.

Events

HMC REST API Client must use this API to read a feed of events.

Resource

`/rest/api/uom/Event`

Every Event lists the following properties

Event Attribute Type	Event Attribute	Definition of the Event Attribute.
EventTypeEnum	EventType	Defines the type of Event.
String	EventID	Defines the abstract unique id for the event.

Event Attribute Type	Event Attribute	Definition of the Event Attribute.
String	EventData	Defines the Primary data for the event.
String	EventDetail	Defines the detailed data for the event.

Event's Generic Purpose would be one among the following:

1. INVALID_URI – A resource has been invalidated or replaced in the ElementHandle2Cache, and the client should reload it if they care.
2. CACHE_CLEARED – essentially everything invalidated since the cache is cleared.
3. MISSING_EVENTS – the client didn't pull events fast enough.
4. ADD_URI – a repository object, mapped to a resource was added.
5. MODIFY_URI – a repository object, mapped to a resource was changed.
6. DELETE_URI – a repository object, mapped to a resource was deleted.

Event semantics

1. Events are Non-persistent.
2. Events are timely (not meaningful to read long past when they occurred).
3. Most events use a URI to specify which REST Resource they relate to.
4. Reading events out of order is not meaningful.

Performance and Capacity Monitoring

REST based web service APIs for Performance and Capacity Monitoring (PCM) of the IBM Power Systems servers.

Ongoing monitoring of systems is critical and vital need for continual and optimal business operations. The need for monitoring is more crucial in virtualized environments due to the shared nature of resources. A PowerVM administrator can understand the capacity distribution of the physical resources among virtual servers and monitor continuously the utilization levels and performances of these resources to ensure physical resources are distributed evenly and used optimally. The admin also need to proactively take actions before a performance problem occurs.

The Performance and Capacity Monitoring (PCM) APIs aims at addressing two primary issues.

1. To bridge the gap in the PowerVM performance monitoring for the new virtualization features.
2. To provide an easy and uniform interface to access the performance data as against making them available via various commands and utilities.

Note: The user must have **`ManageUtilizationData`** task access to be able to change the PCM preference settings.

Related reference:

"Managed System" on page 10
Managed System API provides a list of all system information managed by Hardware Management Console (HMC), such as system name, system machine type, model and serial number, system state, system capabilities, IP address, system migration information, system processor, system memory, and system I/O adapters information for all of the managed system. In addition, each managed system has a list of all child objects, such as:

"Logical Partition" on page 13
Logical Partition provides information about AIX, Linux or IBM i partitions.

"Virtual I/O Server" on page 18
The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources between client logical partitions within the server.

Management Console PCM Preferences

Management console PCM preferences are configuration options provided to the REST API consumers to manage the PCM data collection for managed systems that are connected to a hardware management console (HMC).

The API provides a list of managed systems along with their PCM data collection configuration. It provides details of the following status values:

1. EnergyMonitoringCapable: This is a read-only attribute and provides information on whether the managed system supports Energy data collection or not. For non-supported managed systems, this value is false. This attribute is available from HMC 860 SP1 release onwards.
2. LongTermMonitorEnabled: Long term monitoring configuration value
3. AggregationEnabled: Utilization data aggregation configuration value
4. ShortTermMonitorEnabled: Short term monitoring configuration value
5. ComputeLTMEnabled: Processor and memory metrics monitoring configuration value
6. EnergyMonitorEnabled: Energy monitoring status

For aggregation of the PCM data it is essential to have Long Term Monitoring (LTM) enabled. In case the user enables only aggregation and does not enable LTM, the application implicitly enables LTM and Energy Monitoring, if the managed system supports Energy monitoring.

The Energy Monitoring flag gets enabled when user enables aggregation. The Energy Monitoring flag can also be enabled explicitly without enabling aggregation. In such a case, only raw metrics is available.

The API user can enable or disable the PCM data collection and aggregation for one or more managed systems. The HMC stores aggregated metrics for configured duration. The preferences APIs allows users to specify the storage duration for the aggregated data with **`AggregatedMetricsStorageDuration`** property. The configured storage duration is applicable across all the managed systems.

There is restriction on number of managed system for which PCM functionality can be enabled. The preferences APIs also provides information for how many managed systems the PCM functionality can be enabled. These values are computed based on HMC configuration and they are read only.

1. MaximumManagedSystemsForLongTermMonitor: Indicates maximum number of managed systems for which Long Term Monitoring (LTM) collection is possible.
2. MaximumManagedSystemsForComputeLTM: Indicates maximum number of managed systems for which processor and memory metrics collection is possible.
3. MaximumManagedSystemsForAggregation: Indicates maximum number of managed systems for with Aggregation is possible.
4. MaximumManagedSystemsForShortTermMonitor: Indicates maximum number of managed systems for which Short Term Monitoring (STM) collection is possible.
5. MaximumManagedSystemsForEnergyMonitor: Indicates the maximum number of managed systems for which Energy Monitoring collection is possible. The Energy Monitoring flag is available from HMC Version 8 onwards.

Resource

`/rest/api/pcm/preferences`

Since: Version 1_1_0

Table 88. Supported methods

Method	Description	Internet media type
GET	This API is used to retrieve the preferences set for all the managed systems connected to the HMC.	`application/xml`, `application/vnd.ibm.powervm.pcm.dita`
POST	This API is used to set/update the preferences set for one or more managed systems connected to the HMC. It also allows user to change aggregated metrics storage duration.	`application/xml`, `application/vnd.ibm.powervm.pcm.dita`

Related reference:

"Managed System PCM Preferences"
The managed system PCM preference API is a subset of ManagementConsolePCMpreferences API. This API works on a particular managed system that is connected to HMC and provides APIs enable / disable utilization data collection.

"Long Term Monitor Metrics (LTM)" on page 127
Long term monitor metrics, as the name indicates, are the metrics that are collected for a resource for a longer term monitoring perspective. They are useful for continuous ongoing monitoring of the managed system.

"Short Term Monitor Metrics" on page 129
Short term monitor (STM) provides very detailed metrics on the performance of PowerVM resources. They are primarily useful for troubleshooting purpose. The STM metrics consumes additional server resources. So, the user can disable STM after troubleshooting is completed.

"Processed Metrics for Managed System" on page 131
The processed metrics for Managed System are generated every processing duration (30 seconds), same as the Long Term monitor (LTM) frequency and they are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

"Processed Metrics for Logical Partition" on page 133
The processed metrics for Logical Partition are generated every processing duration (30 seconds), same as the Long Term Monitoring (LTM) frequency and they are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

"Aggregated Metrics for Managed System" on page 135
To provide access to the utilization data for longer duration HMC processes the raw metrics and performs data aggregation. The aggregated metrics are retained in HMC for long duration and duration is configurable by PCM preferences APIs.

"Aggregated Metrics for Logical Partition" on page 137
To provide access to the utilization data for longer duration, the hardware managemetn console (HMC) processes raw metrics and performs data aggregation. The aggregated metrics are retained in the HMC for long duration and duration is configurable by PCM preferences APIs.

Managed System PCM Preferences

The managed system PCM preference API is a subset of ManagementConsolePCMpreferences API. This API works on a particular managed system that is connected to HMC and provides APIs enable / disable utilization data collection.

A managed system has the following PCM preferences:

1. EnergyMonitoringCapable: This is a read-only attribute and provides information on whether the managed system supports Energy data collection or not. For non-supported managed systems, the value of EnergyMonitoringCapable attribute is false. This attribute is available from HMC 860 SP1 release onwards.
2. LongTermMonitorEnabled: Long term monitoring configuration value

3. AggregationEnabled: Utilization data aggregation configuration value
4. ShortTermMonitorEnabled: Short term monitoring configuration value
5. ComputeLTMEnabled: Processor and memory metrics monitoring configuration value
6. EnergyMonitorEnabled: Energy monitoring status configuration value

.

For aggregation of the PCM data, it is essential to have Long Term Monitoring (LTM) enabled. If the user enables only aggregation and does not enable LTM, the application implicitly enables LTM and Energy Monitoring, if the managed system supports Energy monitoring.

The Energy Monitoring flag is available from HMC Version 8 onwards.

Resource

`/rest/api/pcm/ManagedSystem/{uuid}/preferences`

Since: Version 1_1_0

Table 89. Supported methods

Method	Description	Internet media type
GET	This API is used to retrieve the preferences set for a managed system that is connected to the HMC.	`application/xml`, `application/vnd.ibm.powervm.pcm.dita`
POST	This API is used to set/update the preferences set for a managed system that is connected to the HMC.	`application/xml`, `application/vnd.ibm.powervm.pcm.dita`

Related reference:

"Management Console PCM Preferences" on page 124
Management console PCM preferences are configuration options provided to the REST API consumers to manage the PCM data collection for managed systems that are connected to a hardware management console (HMC).

"Long Term Monitor Metrics (LTM)" on page 127
Long term monitor metrics, as the name indicates, are the metrics that are collected for a resource for a longer term monitoring perspective. They are useful for continuous ongoing monitoring of the managed system.

"Short Term Monitor Metrics" on page 129
Short term monitor (STM) provides very detailed metrics on the performance of PowerVM resources. They are primarily useful for troubleshooting purpose. The STM metrics consumes additional server resources. So, the user can disable STM after troubleshooting is completed.

"Processed Metrics for Managed System" on page 131
The processed metrics for Managed System are generated every processing duration (30 seconds), same as the Long Term monitor (LTM) frequency and they are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

"Processed Metrics for Logical Partition" on page 133
The processed metrics for Logical Partition are generated every processing duration (30 seconds), same as the Long Term Monitoring (LTM) frequency and they are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

"Aggregated Metrics for Managed System" on page 135
To provide access to the utilization data for longer duration HMC processes the raw metrics and performs data aggregation. The aggregated metrics are retained in HMC for long duration and duration is configurable by PCM preferences APIs.

"Aggregated Metrics for Logical Partition" on page 137
To provide access to the utilization data for longer duration, the hardware managemetn console (HMC) processes raw metrics and performs data aggregation. The aggregated metrics are retained in the HMC for long duration and duration is configurable by PCM preferences APIs.

Long Term Monitor Metrics (LTM)

Long term monitor metrics, as the name indicates, are the metrics that are collected for a resource for a longer term monitoring perspective. They are useful for continuous ongoing monitoring of the managed system.

Long term monitor metrics API is used to fetch the Long Term Monitoring (LTM) metrics from the hardware management console (HMC). The LTM metrics provides data collected from Power Hypervisor (PHYP) and Virtual I/O Server (VIOS). The LTM metrics are collected at every collection interval (30 seconds) and they are retained in HMC for 30 minutes. After 30 minutes they are completely purged from the system.

Resource

```
/rest/api/pcm/ManagedSystem/{uuid}/RawMetrics/LongTermMonitor
/rest/api/pcm/ManagedSystem/{uuid}/RawMetrics/LongTermMonitor?StartTS={StartTS}&EndTS={EndTS}
```

Since: Version 1_1_0

Quick properties

The API accepts start time stamp and end time stamp as it quick properties. If they are specified the API responds within the specified duration.

Table 90. Quick properties

Quick property	Description	Format
StartTS	Start timestamp is an optional parameter. If specified, the API returns available LTM metrics after the specified time.	yyyy-MM-ddTHH:mm:ssZ
EndTS	End timestamp is an optional parameter. If specified, the API returns available LTM metrics on or before the specified time.	yyyy-MM-ddTHH:mm:ssZ

Table 91. Supported methods

Method	Description	Internet media type
GET	This API is used to retrieve the long term monitoring metrics for a managed system. The response is returned as an Atom feed to the consumer. The Atom Feed contains the Atom Links to the LTM JSON files. There is one JSON file PHYP and one JSON file for each VIOS on the managed system. The Atom link is a web link to the actual JSON file. The content can be retrieved by using the Atom Link.	`application/atom+xml`

Response

Successful execution of the request is returned with response code 200. In response the application returns Atom feed and Atom entries. Attributes of the Atom feed are as follows:

- **Id:** UUID of the managed system.
- **Updated:** Time stamp of the last available LTM metrics data collection in the response.
- **Title:** Name of metrics category.
- **Subtitle:** UUID of the managed system.
- **Link:** Type application or JSON link to the LTM JSON data.
- **Generator:** Name of console.

The Atom feed has series of Atom entries. Attributes of the Atom entry are as follows:

- **Id:** UUID of the managed system
- **Updated:** Start time stamp of the LTM JSON
- **Title:** Name of metrics category and system type
- **Published:** End time stamp of the LTM JSON
- **Link:** Type application JSON, link to the LTM JSON data
- **Author:** Name of console
- **Category:** Indicates the source of the metrics. There are two main sources for the metrics:
  1. **PHYP:** For any managed systems, there is only one Hypervisor. The utilization metrics are gathered from the PHYP.
  2. **VIOS:** For any managed systems, there can be 0 or more VIOS. The utilization metrics are gathered from VIOS. The ID field has a value of VIOS id. For example, if the id = 1, then the category is vios_1. The metrics is generated by VIOS with VIOS id 1.

The application collects data at every collection interval and generates one snapshop for the system's utilization. Each snapshot corresponds to one or more Atom entries.

The metric data are supplied in the JSON format. The following links provide JSON specification from PHYP and VIOS:

- "LTM Power Hypervisor JSON Specification" on page 178
- "LTM Virtual IO Server JSON Specification" on page 182

Related reference:

"Short Term Monitor Metrics" on page 129
Short term monitor (STM) provides very detailed metrics on the performance of PowerVM resources. They are primarily useful for troubleshooting purpose. The STM metrics consumes additional server resources. So, the user can disable STM after troubleshooting is completed.

"Processed Metrics for Managed System" on page 131
The processed metrics for Managed System are generated every processing duration (30 seconds), same as the Long Term monitor (LTM) frequency and they are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

"Processed Metrics for Logical Partition" on page 133
The processed metrics for Logical Partition are generated every processing duration (30 seconds), same as the Long Term Monitoring (LTM) frequency and they are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

"Aggregated Metrics for Managed System" on page 135
To provide access to the utilization data for longer duration HMC processes the raw metrics and performs data aggregation. The aggregated metrics are retained in HMC for long duration and duration is configurable by PCM preferences APIs.

"Aggregated Metrics for Logical Partition" on page 137
To provide access to the utilization data for longer duration, the hardware managemetn console (HMC) processes raw metrics and performs data aggregation. The aggregated metrics are retained in the HMC for long duration and duration is configurable by PCM preferences APIs.

"Managed System PCM Preferences" on page 125
The managed system PCM preference API is a subset of ManagementConsolePCMpreferences API. This API works on a particular managed system that is connected to HMC and provides APIs enable / disable utilization data collection.

"Management Console PCM Preferences" on page 124
Management console PCM preferences are configuration options provided to the REST API consumers to manage the PCM data collection for managed systems that are connected to a hardware management console (HMC).

Short Term Monitor Metrics

Short term monitor (STM) provides very detailed metrics on the performance of PowerVM resources. They are primarily useful for troubleshooting purpose. The STM metrics consumes additional server resources. So, the user can disable STM after troubleshooting is completed.

Short term monitor metrics API is used to fetch the STM metrics from the hardware management console (HMC). Short term monitor metrics provides more granular data collected from Power Hypervisor (PHYP) and Virtual I/O Server (VIOS). The STM data are collected every 5 seconds)once enabled from Preferences and they are retained in HMC for 30 minutes. After 30 minutes they are completely purged from the system.

Resource

```
/rest/api/pcm/ManagedSystem/{uuid}/RawMetrics/ShortTermMonitor
/rest/api/pcm/ManagedSystem/{uuid}/RawMetrics/ShortTermMonitor?StartTS={StartTS}&EndTS={EndTS}
```

Since: Version 1_1_0

Quick properties

The API accepts start timestamp and end timestamp as it quick properties. If they are specified the API responds within the specified duration.

Table 92. Quick properties

Quick property	Description	Format
StartTS	Start timestamp is an optional parameter. If specified, the API returns available STM metrics after the specified time.	yyyy-MM-ddTHH:mm:ssZ
EndTS	End timestamp is an optional parameter. If specified, the API returns available STM metrics on or before the specified time.	yyyy-MM-ddTHH:mm:ssZ

Table 93. Supported methods

Method	Description	Internet media type
GET	This API is used to retrieve the short term monitoring metrics for a managed system. The response is returned as an Atom feed to the consumer. The Atom Feed contains the Atom Links to the STM JSON files. There is one JSON file PHYP and one JSON file for each VIOS on the managed system. The Atom link is a web link to the actual JSON file. The content can be retrieved by using the Atom Link.	`application/atom+xml`

Response

Successful execution of the request is returned with response code 200. In response the application returns Atom feed and Atom entries. Attributes of the Atom feed are as follows:

- **Id:** UUID of the managed system.
- **Updated:** Timestamp of the last available STM metrics data collection in the response.
- **Title:** Name of metrics category.
- **Subtitle:** UUID of the managed system.
- **Link:** Type application/json, link to the STM JSON data.
- **Generator:** Name of console.

The Atom feed has series of Atom entries. Attributes of the Atom entry are as follows:

- **Id:** UUID of the managed system.
- **Updated:** Timestamp of the STM JSON
- **Title:** Name of metrics category and system type.
- **Published:** End timestamp of the STM JSON.
- **Link:** Type application/json, Link to the STM JSON data
- **Author:** Name of console.
- **Category:** Indicates the source of the metrics. There are two main sources for the metrics:
  1. **Power Hypervisor (PHYP):** For any managed systems, there is only one hypervisor. The utilization metrics are gathered from the PHYP.
  2. **Virtual IO Server (VIOS):** For any managed systems, there can be 0 or more VIOSes. The utilization metrics are gathered from VIOS. The ID field has a value of VIOS id. For example, if the id = 1, then the category is vios_1. The metrics is generated by VIOS with VIOS id 1.

The application collects data every at every collection interval and generates one snapshop for the system's utilization. Each snapshot corresponds to one or more Atom entries.

The metric data are supplied in the JSON format. The following links provide JSON specification from PHYP and VIOS.

- "STM Power Hypervisor JSON Specification" on page 184
- "STM Virtual I/O Server JSON Specification" on page 188

Related reference:

"Long Term Monitor Metrics (LTM)" on page 127
Long term monitor metrics, as the name indicates, are the metrics that are collected for a resource for a longer term monitoring perspective. They are useful for continuous ongoing monitoring of the managed system.

"Processed Metrics for Managed System"
The processed metrics for Managed System are generated every processing duration (30 seconds), same as the Long Term monitor (LTM) frequency and they are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

"Processed Metrics for Logical Partition" on page 133
The processed metrics for Logical Partition are generated every processing duration (30 seconds), same as the Long Term Monitoring (LTM) frequency and they are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

"Aggregated Metrics for Managed System" on page 135
To provide access to the utilization data for longer duration HMC processes the raw metrics and performs data aggregation. The aggregated metrics are retained in HMC for long duration and duration is configurable by PCM preferences APIs.

"Aggregated Metrics for Logical Partition" on page 137
To provide access to the utilization data for longer duration, the hardware managemetn console (HMC) processes raw metrics and performs data aggregation. The aggregated metrics are retained in the HMC for long duration and duration is configurable by PCM preferences APIs.

"Managed System PCM Preferences" on page 125
The managed system PCM preference API is a subset of ManagementConsolePCMpreferences API. This API works on a particular managed system that is connected to HMC and provides APIs enable / disable utilization data collection.

"Management Console PCM Preferences" on page 124
Management console PCM preferences are configuration options provided to the REST API consumers to manage the PCM data collection for managed systems that are connected to a hardware management console (HMC).

Processed Metrics for Managed System

The processed metrics for Managed System are generated every processing duration (30 seconds), same as the Long Term monitor (LTM) frequency and they are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

This API is used to retrieve the processed metrics for a managed system. Processed Metrics works on the LTM metrics and performs a series of computations on the available LTM data to generate processed data. This metric is computed every processing duration (30 seconds), once aggregation is enabled from preferences.

Resource

```
/rest/api/pcm/ManagedSystem/{uuid}/ProcessedMetrics
/rest/api/pcm/ManagedSystem/{uuid}/ProcessedMetrics?StartTS={StartTS}&EndTS={EndTS}&NoOfSamples={n}
```

Since: Version 1_1_0

Table 94. Quick properties

Quick property	Description	Format
StartTS	Start timestamp is a mandatory parameter. The API returns available Processed metrics after the specified time.	yyyy-MM-ddTHH:mm:ssZ

Table 94. Quick properties (continued)

Quick property	Description	Format
EndTS	End timestamp is an optional parameter. If specified, the API returns available Processed metrics on or before the specified time.	yyyy-MM-ddTHH:mm:ssZ
NoOfSamples	Number of samples is an optional parameter. If specified, the API returns the specified number of Processed metrics.	int > 0
FileFormat	File format is an optional parameter. If the parameter is not specified, the API returns the Processed metrics in JSON format.	JSON or CSV

Table 95. Supported methods

Method	Description	Internet media type
GET	This API is used to retrieve the processed metrics for a managed system. The response is returned as an Atom feed to the consumer. The Atom Feed contains the Atom Links to all the Processed Json files for the managed system. The Atom link is a web link to the actual json file. The content can be retrieved by using the Atom Link.	`application/atom+xml`

Response

Successful execution of the request is returned with response code 200. In response the application returns Atom feed and Atom entries. Attributes of the Atom feed are as follows:

- **Id:** UUID of the managed system.
- **Updated:** Timestamp of the last available processed metrics data
- **Title:** Name of metrics category.
- **Subtitle:** UUID of the managed system.
- **Link:** Type application/json, link to the processed JSON data.
- **Generator:** Name of console.

The Atom feed has series of Atom entries. Attributes of the Atom entry are as follows:

- **Id:** UUID of the managed system.
- **Updated:** End timestamp of the processed JSON returned.
- **Title:** Name of metrics category and system type.
- **Published:** Start timestamp of the processed JSON returned.
- **Link:** Type application/json, link to the processed JSON data.
- **Author:** Name of console.
- **Category:** Indicates the metrics category, in this case ManagedSystem.
  - **Frequency:** Indicates metrics processing frequency in seconds.

The application processes data every at every processing interval and generates one snapshop for the Managed System's utilization. Each snapshot corresponds to an Atom entry.

The metric data are supplied in the JSON format. The following link provide JSON specification for Managed System processed metrics.

- "Managed System Processed and Aggregated Metrics JSON Specification" on page 191

Related reference:

"Long Term Monitor Metrics (LTM)" on page 127
Long term monitor metrics, as the name indicates, are the metrics that are collected for a resource for a longer term monitoring perspective. They are useful for continuous ongoing monitoring of the managed system.

"Short Term Monitor Metrics" on page 129
Short term monitor (STM) provides very detailed metrics on the performance of PowerVM resources. They are primarily useful for troubleshooting purpose. The STM metrics consumes additional server resources. So, the user can disable STM after troubleshooting is completed.

"Processed Metrics for Logical Partition"
The processed metrics for Logical Partition are generated every processing duration (30 seconds), same as the Long Term Monitoring (LTM) frequency and they are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

"Aggregated Metrics for Managed System" on page 135
To provide access to the utilization data for longer duration HMC processes the raw metrics and performs data aggregation. The aggregated metrics are retained in HMC for long duration and duration is configurable by PCM preferences APIs.

"Aggregated Metrics for Logical Partition" on page 137
To provide access to the utilization data for longer duration, the hardware managemetn console (HMC) processes raw metrics and performs data aggregation. The aggregated metrics are retained in the HMC for long duration and duration is configurable by PCM preferences APIs.

"Managed System PCM Preferences" on page 125
The managed system PCM preference API is a subset of ManagementConsolePCMpreferences API. This API works on a particular managed system that is connected to HMC and provides APIs enable / disable utilization data collection.

"Management Console PCM Preferences" on page 124
Management console PCM preferences are configuration options provided to the REST API consumers to manage the PCM data collection for managed systems that are connected to a hardware management console (HMC).

Processed Metrics for Logical Partition

The processed metrics for Logical Partition are generated every processing duration (30 seconds), same as the Long Term Monitoring (LTM) frequency and they are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

This API is used to retrieve the processed metrics for a logical partition. Processed Metrics works on the LTM metrics and performs a series of computations on the available LTM data to generate the processed data. This metric is computed every processing duration (30 seconds), once aggregation is enabled from preferences.

Resource

```
/rest/api/pcm/LogicalPartition/{uuid}/ProcessedMetrics
/rest/api/pcm/LogicalPartition/{uuid}/ProcessedMetrics?StartTS={StartTS}&EndTS={EndTS}&NoOfSamples={n}
```

Since: Version 1_1_0

Table 96. Quick properties

Quick property	Description	Format
StartTS	Start timestamp is a mandatory parameter. The API returns available Processed metrics after the specified time.	yyyy-MM-ddTHH:mm:ssZ
EndTS	End timestamp is an optional parameter. If specified, the API returns available Processed metrics on or before the specified time.	yyyy-MM-ddTHH:mm:ssZ
NoOfSamples	Number of samples is an optional parameter. If specified, the API returns the specified number of Processed metrics.	int > 0

Table 97. Supported methods

Method	Description	Internet media type
GET	This API is used to retrieve the processed metrics for a logical partition. The response is returned as an Atom feed to the consumer. The Atom Feed contains the Atom Links to all the Processed JSON files for the logical partition. The Atom link is a web link to the actual JSON file. The content can be retrieved by using the Atom Link.	`application/atom+xml`

Response

Successful execution of the request is returned with response code 200. In response the application returns Atom feed and Atom entries. Attributes of the Atom feed are as follows:

- **Id:** UUID of the logical partition.
- **Updated:** Timestamp of the last available processed metrics data.
- **Title:** Name of metrics category.
- **Subtitle:** UUID of the managed system.
- **Link:** Type application/json, link to the processed JSON data.
- **Generator:** Name of console.

The Atom feed has series of Atom entries. Attributes of the Atom entry are as follows:

- **Id:** UUID of the managed system.
- **Updated:** End timestamp of the Processed JSON returned.
- **Title:** Name of metrics category and system type.
- **Published:** Start timestamp of the Processed JSON returned.
- **Link:** Type application/json, Link to the Processed JSON data.
- **Author:** Name of console.
- **Category:** Indicates the metrics category, in this case LogicalPartition.
  - **Frequency:** Indicates metrics processing frequency in seconds.

The application processes data every at every processing interval and generates one snapshot for the logical partition utilization. Each snapshot corresponds to an Atom entry.

The metric data are supplied in the JSON format. The following link provide JSON specification for Logical Partition processed metrics.

- "Logical Paritition Processed/Aggregated Metrics JSON Specification" on page 199

Related reference:

"Long Term Monitor Metrics (LTM)" on page 127
Long term monitor metrics, as the name indicates, are the metrics that are collected for a resource for a longer term monitoring perspective. They are useful for continuous ongoing monitoring of the managed system.

"Short Term Monitor Metrics" on page 129
Short term monitor (STM) provides very detailed metrics on the performance of PowerVM resources. They are primarily useful for troubleshooting purpose. The STM metrics consumes additional server resources. So, the user can disable STM after troubleshooting is completed.

"Processed Metrics for Managed System" on page 131
The processed metrics for Managed System are generated every processing duration (30 seconds), same as the Long Term monitor (LTM) frequency and they are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

"Aggregated Metrics for Managed System"
To provide access to the utilization data for longer duration HMC processes the raw metrics and performs data aggregation. The aggregated metrics are retained in HMC for long duration and duration is configurable by PCM preferences APIs.

"Aggregated Metrics for Logical Partition" on page 137
To provide access to the utilization data for longer duration, the hardware managemetn console (HMC) processes raw metrics and performs data aggregation. The aggregated metrics are retained in the HMC for long duration and duration is configurable by PCM preferences APIs.

"Managed System PCM Preferences" on page 125
The managed system PCM preference API is a subset of ManagementConsolePCMpreferences API. This API works on a particular managed system that is connected to HMC and provides APIs enable / disable utilization data collection.

"Management Console PCM Preferences" on page 124
Management console PCM preferences are configuration options provided to the REST API consumers to manage the PCM data collection for managed systems that are connected to a hardware management console (HMC).

Aggregated Metrics for Managed System

To provide access to the utilization data for longer duration HMC processes the raw metrics and performs data aggregation. The aggregated metrics are retained in HMC for long duration and duration is configurable by PCM preferences APIs.

This API is used to retrieve the aggregated metrics for a managed system. To retain the processed performance utilization data for longer duration they are aggregated. The levels of aggregation are:

1. Tier 1 roll up: The processed metrics collected over 15 minutes forms 1 aggregated metrics sample. The tier 1 metrics are retained for 2 hours.
2. Tier 2 roll up: The tier 1 metrics are further rolled up to have one sample for 2 hours. The tier 2 metrics are retained for 1 week.
3. Tier 3 roll up: The tier 2 metrics are further rolled up to have one sample per day. The tier 3 metrics are retained until the retention duration set by preferences API.

In the rolled up data, for example, for the aggregated metric, the system maintains 3 values: Average, Minimum, and Maximum of the underlying processed metrics.

Resource

```
/rest/api/pcm/ManagedSystem/{uuid}/AggregatedMetrics
```

```
/rest/api/pcm/ManagedSystem/{uuid}/AggregatedMetrics?StartTS={StartTS}&EndTS={EndTS}&NoOfSamples=
{n}&Feed={feed}
```

Since: Version 1_1_0

Table 98. Quick properties

Quick property	Description	Format
StartTS	Start timestamp is a mandatory parameter. The API returns available Aggregated metrics after the specified time.	yyyy-MM-ddTHH:mm:ssZ
EndTS	End timestamp is an optional parameter. If specified, the API returns available Aggregated metrics on or before the specified time.	yyyy-MM-ddTHH:mm:ssZ
NoOfSamples	Number of samples is an optional parameter. If specified, the API returns the specified number of Aggregated metrics.	int > 0
Feed	Feed is an optional parameter. • **bySource:** the API returns Aggregated metrics for Managed system and links to fetch Aggregated metrics for its logical paritions. • **byTier:** the API returns consolidated Aggregated metrics for Managed system and all its logical partitions.	Specify one of the two values: • bySource (default) • byTier

Table 99. Supported methods

Method	Description	Internet media type
GET	This API is used to retrieve the aggregated metrics for a managed system. The response is returned as an Atom feed to the consumer. The Atom Feed contains the Atom Links to all the Aggregated JSON files for the managed system. The Atom link is a web link to the actual JSON file. The content can be retrieved by using the Atom Link.	`application/atom+xml`

Response

Successful execution of the request is returned with response code 200. In response the application returns Atom feed and Atom entries. Attributes of the Atom feed follow:

- **Id:** UUID of the Managed System
- **Updated:** Timestamp of the last available Aggregated metrics data

The Atom feed has series of Atom entries. Attributes of the Atom entry follow:

- **Updated:** End timestamp of the Aggregated JSON returned
- **Published:** Start timestamp of the Aggregated JSON returned

- **Link:** Type application/json, Link to the Aggregated JSON data
- **Category:** Indicates the metrics category, in this case ManagedSystem
  - **Frequency:** Indicates metrics aggregation frequency in seconds

The application aggregates data every at every aggregation interval and generates snapshots for the managed system utilization. Each snapshot corresponds to an Atom entry.

The metric data are supplied in the JSON format. The following link provide JSON specification for managed system aggregated metrics.

- "Managed System Processed and Aggregated Metrics JSON Specification" on page 191

Related reference:

"Long Term Monitor Metrics (LTM)" on page 127
Long term monitor metrics, as the name indicates, are the metrics that are collected for a resource for a longer term monitoring perspective. They are useful for continuous ongoing monitoring of the managed system.

"Short Term Monitor Metrics" on page 129
Short term monitor (STM) provides very detailed metrics on the performance of PowerVM resources. They are primarily useful for troubleshooting purpose. The STM metrics consumes additional server resources. So, the user can disable STM after troubleshooting is completed.

"Processed Metrics for Managed System" on page 131
The processed metrics for Managed System are generated every processing duration (30 seconds), same as the Long Term monitor (LTM) frequency and they are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

"Processed Metrics for Logical Partition" on page 133
The processed metrics for Logical Partition are generated every processing duration (30 seconds), same as the Long Term Monitoring (LTM) frequency and they are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

"Aggregated Metrics for Logical Partition"
To provide access to the utilization data for longer duration, the hardware managemetn console (HMC) processes raw metrics and performs data aggregation. The aggregated metrics are retained in the HMC for long duration and duration is configurable by PCM preferences APIs.

"Managed System PCM Preferences" on page 125
The managed system PCM preference API is a subset of ManagementConsolePCMpreferences API. This API works on a particular managed system that is connected to HMC and provides APIs enable / disable utilization data collection.

"Management Console PCM Preferences" on page 124
Management console PCM preferences are configuration options provided to the REST API consumers to manage the PCM data collection for managed systems that are connected to a hardware management console (HMC).

Aggregated Metrics for Logical Partition

To provide access to the utilization data for longer duration, the hardware managemetn console (HMC) processes raw metrics and performs data aggregation. The aggregated metrics are retained in the HMC for long duration and duration is configurable by PCM preferences APIs.

This API is used to retrieve the aggregated metrics for a logical partition. To retain the processed performance utilization data for longer duration they are aggregated. There are three levels of aggregation:

1. Tier 1 roll up: The processed metrics collected over 15 minutes forms 1 aggregated metrics sample. The tier 1 metrics are retained for 2 hours.
2. Tier 2 roll up: The tier 1 metrics are further rolled up to have one sample for 2 hours. The tier 2 metrics are retained for 1 week.

3. Tier 3 roll up: The tier 2 metrics are further rolled up to have one sample per day. The tier 3 metrics are retained until the retention duration set by preferences API.

In the rolled up data, for example, for the aggregated metric, the system maintains 3 values: Average, Minimum, and Maximum of the underlying processed metrics.

Resource

```
/rest/api/pcm/LogicalPartition/{uuid}/AggregatedMetrics
/rest/api/pcm/LogicalPartition/{uuid}/AggregatedMetrics?StartTS={StartTS}&EndTS={EndTS}&NoOfSamples={n}
```

Since: Version 1_1_0

Table 100. Quick properties

Quick property	Description	Format
StartTS	Start timestamp is a mandatory parameter. The API returns available Aggregated metrics after the specified time.	yyyy-MM-ddTHH:mm:ssZ
EndTS	End timestamp is an optional parameter. If specified, the API returns available Aggregated metrics on or before the specified time.	yyyy-MM-ddTHH:mm:ssZ
NoOfSamples	Number of samples is an optional parameter. If specified, the API returns the specified number of Aggregated metrics.	int > 0

Table 101. Supported methods

Method	Description	Internet media type
GET	This API is used to retrieve the aggregated metrics for a logical partition. The response is returned as an Atom feed to the consumer. The Atom Feed contains the Atom Links to all the Aggregated JSON files for the logical partition. The Atom link is a web link to the actual JSON file. The content can be retrieved by using the Atom Link.	`application/atom+xml`

Response

Successful execution of the request is returned with response code 200. In response the application returns Atom feed and Atom entries. Attributes of the Atom feed follow:

- **Id:** UUID of the Logical Partition
- **Updated:** Timestamp of the last available Aggregated metrics data

The Atom feed has series of Atom entries. Attributes of the Atom entry follow:

- **Updated:** End timestamp of the Aggregated JSON returned
- **Published:** Start timestamp of the Aggregated JSON returned
- **Link:** Type application/json, Link to the Aggregated JSON data
- **Category:** Indicates the metrics category, in this case LogicalPartition

– **Frequency:** Indicates metrics aggregation frequency in seconds

The application aggregates data every at every aggregation interval and generates snapshops for the Logical Partition's utilization. Each snapshot corresponds to an Atom entry.

The metric data are supplied in the JSON format. The following link provide JSON specification for Logical Partition aggregated metrics.

- "Logical Paritition Processed/Aggregated Metrics JSON Specification" on page 199

Related reference:

"Long Term Monitor Metrics (LTM)" on page 127
Long term monitor metrics, as the name indicates, are the metrics that are collected for a resource for a longer term monitoring perspective. They are useful for continuous ongoing monitoring of the managed system.

"Short Term Monitor Metrics" on page 129
Short term monitor (STM) provides very detailed metrics on the performance of PowerVM resources. They are primarily useful for troubleshooting purpose. The STM metrics consumes additional server resources. So, the user can disable STM after troubleshooting is completed.

"Processed Metrics for Managed System" on page 131
The processed metrics for Managed System are generated every processing duration (30 seconds), same as the Long Term monitor (LTM) frequency and they are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

"Processed Metrics for Logical Partition" on page 133
The processed metrics for Logical Partition are generated every processing duration (30 seconds), same as the Long Term Monitoring (LTM) frequency and they are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

"Aggregated Metrics for Managed System" on page 135
To provide access to the utilization data for longer duration HMC processes the raw metrics and performs data aggregation. The aggregated metrics are retained in HMC for long duration and duration is configurable by PCM preferences APIs.

"Managed System PCM Preferences" on page 125
The managed system PCM preference API is a subset of ManagementConsolePCMpreferences API. This API works on a particular managed system that is connected to HMC and provides APIs enable / disable utilization data collection.

"Management Console PCM Preferences" on page 124
Management console PCM preferences are configuration options provided to the REST API consumers to manage the PCM data collection for managed systems that are connected to a hardware management console (HMC).

Energy Monitoring

Energy monitoring provides information about the power consumption data of a Managed System, along with generated heat from inlet, CPU, and baseboards. These details are helpful in finding whether a system is overloaded, is generating more heat, or use more power than expected. The metrics also give where the load can be distributed.

This metric data is collected at 30-seconds interval when the energy monitoring flag is set to true. These metrics are stored in the HMC for 30 minutes and are then purged from the system. The raw data is used to generate Processed and Aggregated metrics by further rolling up.

Energy monitoring is supported by Managed Systems having FSP level 860.1 and above.

Energy monitoring RAW API

This API is used to retrieve the RAW metrics for Energy Monitoring of a Managed System. The RAW metrics data is collected from the Flexible Service Processor (FSP) and it is collected at every collection

interval (30 seconds). The metrics data is retained in HMC for 30 minutes. After 30 minutes, the data is completely purged from the system. The RAW data is used to generate the Processed and Aggregated metrics by further rolling up.

The collection of Raw metrics job runs every 30 seconds. It starts at xx:xx:00 and continues at xx:xx:30, and so on. If the Raw collection process gets errors for five consecutive times, the collection is paused for 15 minutes and then is resumed automatically.

Resource

```
/rest/api/pcm/ManagedSystem/<ManagedSystemUuid>/RawMetrics/EnergyMonitor
/rest/api/pcm/ManagedSystem/<ManagedSystemUuid>/RawMetrics/EnergyMonitor?
StartTS=yyyy-MM-dd'T'hh:mm:ssZ&EndTS= yyyy-MM-dd'T'hh:mm:ssZ
```

An example for date is 2017-05-22T18:43:29+0000.

Table 102. Quick properties

Quick property	Description	Format
StartTS	The start time stamp is an optional parameter. If specified, the API returns the available Energy metrics from the specified time.	yyyy-MM-dd'T'HH:mm:ssZ
EndTS	The end time stamp is an optional parameter. If specified, the API returns available Energy metrics on or before the specified time.	yyyy-MM-dd'T'HH:mm:ssZ

Table 103. Supported methods

Method	Description	Internet media type
GET	This API is used to retrieve the energy monitoring metrics for a managed system.	`application/xml`

Response

```
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:ns2="http://a9.com/-/spec/opensearch/1.1/"
xmlns:ns3="http://www.w3.org/1999/xhtml">
 <id>FEED_UNIQUE_ID</id>
 <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>
 <title type="text">EnergyMonitorMetrics</title>
 <subtitle type="text">ManagedSystem MANAGED_SYSTEM_UUID</subtitle>
 <link rel="self" href="https://HMC_IP_OR_HOST_NAME/rest/api/pcm/ManagedSystem/
MANAGED_SYSTEM_UUID/RawMetrics/EnergyMonitor"/>
 <generator uri="IBM Power Systems Management Console" version="1"/>
 <entry>
  <id>ENTRY_UNIQUE_ID</id>
  <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>
```

```
  <title type="text">LTM_MANAGED_SYSTEM_MTMS_fsp_ yyyyMMDDThhmmss +0000.json</title>
  <published>YYYY-MM-DDThh:mm:ss.sssZ</published>
  <link type="application/vnd.ibm.powervm.pcm.json" href="https://HMC_IP_OR_HOST_NAME:
              443/rest/api/pcm/ManagedSystem/MANAGED_SYSTEM_UUID/RawMetrics/EnergyMonitor/
      FILE_NAME.json"/>
  <author>
   <name>IBM Power Systems Management Console</name>
  </author>
  <category term="fsp"/>
 </entry>
 <entry>
  <id>ENTRY_UNIQUE_ID</id>
  <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>
  <title type="text">LTM_MANAGED_SYSTEM_MTMS_fsp_ yyyyMMDDThhmmss +0000.json</title>
  <published>YYYY-MM-DDThh:mm:ss.sssZ</published>
  <link type="application/vnd.ibm.powervm.pcm.json" href="https://HMC_IP_OR_HOST_NAME:
              443/rest/api/pcm/ManagedSystem/MANAGED_SYSTEM_UUID/RawMetrics/EnergyMonitor/
      FILE_NAME.json"/>
  <author>
   <name>IBM Power Systems Management Console</name>
  </author>
  <category term="fsp"/>
 </entry>
</feed>
```

Successful execution of the request is returned with response code 200. In response, the application returns feed and entries.

Each entries have the link to the RAW metric data that is provided in the JSON format. See "JSON specification for Energy monitoring RAW metrics" on page 146 to view the JSON specification for the Energy RAW metrics.

Processed Metrics for Energy Monitoring

When the **Aggregation** flag is enabled in PCM Preferences, the processed metrics for Energy Monitoring are generated for the specific processing duration of 30 seconds. The metric data is retained up to 2 hours. After 2 hours, tithe data is rolled up by the aggregation logic. Processed metrics uses the Raw metrics and performs a series of computations on the raw data to generate the processed data.

Resource

The processing of Raw metrics job runs every 30 seconds. For example, the interval can be xx:xx:00 and then at xx:xx:30 and so on. Processed metrics give utilization data per time interval (30 seconds).

```
/rest/api/pcm/ManagedSystem/<ManagedSystemUuid>/ProcessedMetrics?Type=Energy
/rest/api/pcm/ManagedSystem/<ManagedSystemUuid>/ProcessedMetrics?Type=Energy&
StartTS={StartTS}&EndTS={EndTS}&NoOfSamples={n}
```

An example for date is 2017-05-22T18:43:29+0000

Table 104. Quick properties

Quick property	Description	Format
Type	If the value is specified, the API returns available Processed metrics only for Energy Monitoring.	
StartTS	The start time stamp is an optional parameter. The API returns available Processed metrics from the specified time.	yyyy-MM-dd'T'HH:mm:ssZ
EndTS	The end timestamp is an optional parameter. If specified, the API returns available Processed metrics on or before the specified time.	yyyy-MM-dd'T'HH:mm:ssZ
NoOfSamples	Number of samples is an optional parameter. If specified, the API returns the specified number of Processed metrics.	Integer > 0

Table 105. Supported methods

Method	Description	Internet media type
GET	This API is used to retrieve the processed metrics for a managed system. The response is returned as an Atom feed to the consumer.	`application/xml`

Response

```
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:ns2="http://a9.com/-/spec/opensearch/1.1/"
            xmlns:ns3="http://www.w3.org/1999/xhtml">
 <id>FEED_UNIQUE_ID</id>
 <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>
 <title type="text">ProcessedMetrics</title>
 <subtitle type="text">ManagedSystem MANAGED_SYSTEM_UUID</subtitle>
 <link rel="self" href="https://HMC_IP_OR_HOST_NAME/rest/api/pcm/ManagedSystem/
MANAGED_SYSTEM_UUID/ProcessedMetrics?Type=Energy"/>
 <generator uri="IBM Power Systems Management Console" version="1"/>
 <entry>
```

```
  <id>ENTRY_UNIQUE_ID</id>

  <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>

  <title type="text">JSON file name</title>

  <published>YYYY-MM-DDThh:mm:ss.sssZ</published>

  <link type="application/json" href="https://HMC_IP_OR_HOST_NAME:

              443/rest/api/pcm/ProcessedMetrics/FILE_NAME.json"/>

  <author>

   <name>IBM Power Systems Management Console</name>

  </author>

  <category term="ManagedSystem" frequency="30"/>

 </entry>

</feed>
```

Successful execution of the request is returned with response code 200. In response, the application returns Atom feed and Atom entries.

The Entry includes the link of the Processed data file that is in the JSON format. See "JSON specification for Energy Monitoring Processed and Aggregated Metrics" on page 148 to view the JSON specification for the Energy Processed and Aggregated metrics.

Aggregated Metrics for Energy Monitoring

The Hardware Management Console (HMC) processes the Energy Monitoring Processed metrics and performs data aggregation to provide access to the utilization data for longer duration. The Aggregated metrics are retained in HMC for a specified duration and duration can be configured using the PCM Management Console preferences APIs.

This API is used to retrieve the Aggregated metrics for Energy Monitoring. To retain the processed performance utilization data for a longer duration, the data is aggregated. The levels of aggregation are:

1. Tier 1 roll up: The first roll-up happens every 5 minutes. The API uses the Processed metrics of the last 5 minutes and forms an aggregated metrics sample. The Tier 1 metrics are retained for 24 hours. The roll up job starts at 00:01:00 hours and continues to 00:06:00 hours, 00:11:00 hours, and so on.
2. Tier 2 roll up: The second roll-up happens every 2 hours. The API uses the Tier 1 metrics of the last 2 hours and forms an aggregated metrics sample. The Tier 2 metrics are retained for 1 week. The roll up job starts at 00:01:30 hours and continues to 02:01:30 hours, 04:01:30 hours, and so on.
3. Tier 3 roll up: The tier 2 metrics are rolled up to have one sample per day. The Tier 2 metrics are retained for the retention duration set by the Management Console preferences API. This roll-up job starts each day at 00:02:00 hours.

For example, in the rolled-up data for the Aggregated metric, the system maintains the **Average**, **Minimum**, and **Maximum** values of the underlying processed metrics.

Resource

```
/rest/api/pcm/ManagedSystem/<ManagedSystemUuid>/AggregatedMetrics?Type=Energy
/rest/api/pcm/ManagedSystem/<ManagedSystemUuid>/AggregatedMetrics?

Type=Energy& StartTS={StartTS}&EndTS={EndTS}&NoOfSamples={n}
```

An example for date is 2017-05-22T18:43:29+0000.

Table 106. Quick properties

Quick property	Description	Format
Type	If the value is specified as Energy, the API returns available Aggregated metrics only for Energy Monitoring.	
StartTS	The start time stamp is an optional parameter. The API returns available Aggregated metrics from the specified time.	yyyy-MM-dd'T'HH:mm:ssZ
EndTS	The end time stamp is an optional parameter. If the value is specified, the API returns available Aggregated metrics on or before the specified time.	yyyy-MM-dd'T'HH:mm:ssZ
NoOfSamples	Number of samples is an optional parameter. If the value is specified, the API returns the specified number of Aggregated metrics.	Integer > 0

Table 107. Supported methods

Method	Description	Internet media type
GET	This API is used to retrieve the aggregated metrics for a managed system. The response is returned as an Atom feed to the consumer.	`application/xml`

Response

```
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:ns2="http://a9.com/-/spec/opensearch/1.1/"

 xmlns:ns3="http://www.w3.org/1999/xhtml">

 <id>FEED_UNIQUE_ID</id>

 <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>

 <title type="text">AggregatedMetrics</title>

 <subtitle type="text">ManagedSystem MANAGED_SYSTEM_UUID</subtitle>

 <link rel="self" href="https://HMC_IP_OR_HOST_NAME/rest/api/pcm/ManagedSystem/

MANAGED_SYSTEM_UUID/AggregatedMetrics?Type=Energy"/>

 <generator uri="IBM Power Systems Management Console" version="1"/>

 <entry>

  <id>ENTRY_UNIQUE_ID</id>

  <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>

  <title type="text">JSON file name</title>

  <published>YYYY-MM-DDThh:mm:ss.sssZ</published>

  <link type="application/json" href="https://HMC_IP_OR_HOST_NAME:

              443/rest/api/pcm/AggregatedMetrics/FILE_NAME.json"/>
```

```
 <author>
  <name>IBM Power Systems Management Console</name>
 </author>
 <category term="ManagedSystem" frequency="300"/>
</entry>
<entry>
 <id>ENTRY_UNIQUE_ID</id>
 <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>
 <title type="text">JSON file name</title>
 <published>YYYY-MM-DDThh:mm:ss.sssZ</published>
 <link type="application/json" href="https://HMC_IP_OR_HOST_NAME:
             443/rest/api/pcm/AggregatedMetrics/FILE_NAME.json"/>
 <author>
  <name>IBM Power Systems Management Console</name>
 </author>
 <category term="ManagedSystem" frequency="7200"/>
</entry>
<entry>
 <id>ENTRY_UNIQUE_ID</id>
 <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>
 <title type="text">JSON file name</title>
 <published>YYYY-MM-DDThh:mm:ss.sssZ</published>
 <link type="application/json" href="https://HMC_IP_OR_HOST_NAME:
             443/rest/api/pcm/AggregatedMetrics/FILE_NAME.json"/>
 <author>
  <name>IBM Power Systems Management Console</name>
 </author>
 <category term="ManagedSystem" frequency="86400"/>
</entry>
</feed>
```

Successful execution of the request is returned with response code 200. In response, the application returns Atom feed and Atom entries for each Tier.

The Feed includes file links for the Aggregated metrics for each Tier. The metric data is supplied in the JSON format. See “JSON specification for Energy Monitoring Processed and Aggregated Metrics” on page 148

148 view the JSON specification for Energy Monitoring Aggregated metrics.

JSON specification for Energy monitoring RAW metrics

The PCM Raw metric for Energy monitoring URI provides the following utilization data that is fetched by the Hardware Management Console (HMC) from the Flexible Service Processor (FSP) when the **EnergyMonitorEnabled** parameter is enabled for a managed system.

Resource

```
/rest/api/pcm/ManagedSystem/[Managed System Uuid]/RawMetrics/EnergyMonitor/

LTM_[Managed System MTMS]_fsp_yyyyMMDDThhmmss+0000.json
```

Response

```
{

 "timeStamp": "String",

 "status": Number,

 "errorInfo": [{

   "errId": "String",

   "errMsg": "String"

  }],

 "timeBasedCycles": Number,

 "utilInfo": {

  "version": "String",

  "metricType": "String",

  "monitoringType": "String",

  "mtms": "String",

  "name": "String"

 },

 "powerEnergyReading": {

  "currentPowerReading": Float,

  "minimumPowerReading": Float,

  "maximumPowerReading": Float,

  "averagePowerReading": Float,

  "ipmiTimeStamp": "String",

  "reportingPeriod": Number,

  "powerReadingState": "String"

 },

 "thermalEnergyReading": {

  "inletTemperatures": [{
```

```
   "entityId": "String",
   "entityInstance": "String",
   "temperatureData": Float
  }],
  "cpuTemperatures": [{
   "entityId": "String",
   "entityInstance": "String",
   "temperatureData": Float
  }],
  "baseboardTemperatures": [{
   "entityId": "String",
   "entityInstance": "String",
   "temperatureData": Float
  }]
 }
}
```

The following table provides details of the metrics.

Table 108. Description of metric parameters.

Metric name	Description
timeStamp	Timestamp of that sample
status	The valid values are: • Status 0: all collections were success • Status 1: all collections were failed • Status 2: part of the collection was success
errorInfo	Information about any error that occurred
errId	ID of the error
errMsg	Detailed information of the error
utilInfo	Information about the sample
Version	Version of JSON format for Energy Monitoring
metricType	Type of the sample. For Raw metrics, the value is **Raw**.
monitoringType	Type of the monitoring. For Raw metrics, the value is **Raw**.
name	Name of the Managed System
mtms	Machine Type, Model, and Serial number of the Managed System
powerEnergyReading	Power consumption data
currentPowerReading	Power consumption at the current time
minimumPowerReading	Minimum Power consumption for the time interval (30 seconds)

Table 108. Description of metric parameters (continued).

Metric name	Description
maximumPowerReading	Maximum Power consumption for the time interval (30 seconds)
averagePowerReading	Average Power consumption for the time interval (30 seconds)
ipmiTimeStamp	Timestamp reported by the ipmitool
reportingPeriod	Number of seconds for which the data is reported
powerReadingState	State of the Power Reading from FSP. It can be active or inactive
thermalEnergyReading	Data related to heat generated by the managed system
inletTemperatures	Generated heat data for the Inlet
cpuTemperatures	Generated heat data for the CPUs
baseboardTemperatures	Generated heat data for the Baseboards
entityId	The ID of the entity such as Inlet or CPU or Baseboard
entityInstance	Instance name of the entity like Inlet or CPU or Baseboard
temperatureReading	Heat generated by the entities for the time interval in Celsius

JSON specification for Energy Monitoring Processed and Aggregated Metrics

All the parameters for the Processed and Aggregated metrics are same except that the Processed metrics provides only Average metrics, and it does not provide minimum and maximum records. The PCM Processed or Aggregated metrics URI for Energy Monitoring provides the following power consumption and thermal data when the **AggregationEnabled** parameter is enabled for a managed system.

Resource

```
/rest/api/pcm/ProcessedMetrics/EnergyMetrics_ManagedSystem_[Managed System Uuid]_

yyyyMMDDThhmmss+0000_yyyyMMDDThhmmss+0000_30.json
/rest/api/pcm/AggregatedMetrics/EnergyMetrics_ManagedSystem_[Managed System Uuid]_

yyyyMMDDThhmmss+0000_yyyyMMDDThhmmss+0000_[Tier frequency in seconds].json
```

Response

```
{

 "systemUtil": {

  "utilInfo": {

   "version": "String",

   "metricType": "String",

   "frequency": Number,

   "startTimeStamp": "String",

   "endTimeStamp": "String",

   "mtms": "String",

   "name": "String",
```

```
 "metricArrayOrder": ["AVG", "MIN", "MAX"],
 "uuid": "String"
},
"utilSamples": [{
 "sampleType": "ManagedSystem",
 "energyUtil": {
  "powerUtil": {
   "powerReading": [Float, Float, Float]
  },
  "thermalUtil": {
   "inletTemperatures": [{
     "entityId": "String",
     "entityInstance": "String",
     "temperatureReading": [Float, Float, Float]
   }],
   "cpuTemperatures": [{
     "entityId": "String",
     "entityInstance": "String",
     "temperatureReading": [Float, Float, Float]
   }],
   "baseboardTemperatures": [{
     "entityId": "String",
     "entityInstance": "String",
     "temperatureReading": [Float, Float, Float]
   }]
  }
 },
 "sampleInfo": {
  "timeStamp": "String",
  "numOfSamplesAggregated": Number,
  "status": Number
 }
}]
```

}

}

Table 109. Description of metric parameters.

Metric name	Description
Version	Version of JSON format for Energy Monitoring.
metricType	Type of the sample. The values are **Processed** or **Aggregated**.
frequency	The collection cycle in seconds.
startTimeStamp	Start Timestamp of the samples collected
endTimeStamp	End Timestamp of the samples collected
uuid	UUID of the Managed System
name	name of the Managed System
mtms	Machine Type, Model, and Serial number of the Managed System
metricArrayOrder	**AVG**, **MIN**, and **MAX**. For Processed metrics, the value is **AVG**
utilSamples: Containing sample of Processed or Aggregated Energy Monitoring metrics.	
sampleType	Type of the sample. The value is **ManagedSystem**
energyUtil	Contains the Energy & Power related data
powerUtil	Power consumption data
powerReading	Power consumed by the system for the specific interval, in watt
thermalUtil	Data related to heat generated by the managed system
inletTemperatures	Generated heat data for the Inlet
cpuTemperatures	Generated heat data for the CPUs
baseboardTemperatures	Generated heat data for the Baseboards
entityId	The ID of the entity. The value can be **Inlet**, **CPU**, or **Baseboard**.
entityInstance	Instance name of the entity. The value can be **Inlet**, **CPU**, or **Baseboard**.
temperatureReading	Heat generated by the entities for the time interval, in Celsius
utilSamples: sampleInfo	
timeStamp	Timestamp of the sample
numOfSamplesAggregated	Number of samples that are rolled up to get the sample. This attribute is available only for Aggregated metrics.
status	The valid values are: • Status 0: all collections were success • Status 1: all collections were failed • Status 2: part of the collection was success

Shared Storage Pool Monitoring

The Shared Storage Pool (SSP) monitoring REST API provides options to set collection preferences, to get raw data that is collected every 5 minutes, and to consume Processed and Aggregated metrics.

SSP monitoring is used to collect performance data about the SSP and process them to generate aggregated metrics. This API is supported from Hardware Management Console (HMC) Version 8.6, and later and Virtual I/O Server (VIOS) Version 2.2.2.5.00, and later.

SSP Preferences

The Shared Storage Pool (SSP) PCM preference allows you to enable or disable the option for SSP PCM data collection and data Aggregation for one or more SSPs. The data collection for SSP is done by using the node or the virtual I/O server (VIOS) that are connected to the Hardware Management Console (HMC).

SSP has the following PCM preferences:

- **MonitorEnabled**: Raw Metrics collection status
- **AggregationEnabled**: Utilization data aggregation status

Note: When the Aggregation metrics is enabled, the Monitoring metrics is automatically enabled.

Resource

/rest/api/pcm/preferences/SSP

Table 110. Supported methods

Method	Description	Internet media type
GET	This API is used to retrieve the preferences set for SSPs that are managed by the HMC.	application/xml
POST	This API is used to update the preferences set for SSPs that are managed by the HMC.	application/xml

Resource

/rest/api/pcm/SSP/<SSPUuId>/preferences

Table 111. Supported methods

Method	Description	Internet media type
GET	This API is used to retrieve the preferences set for a single SSP that is managed by the HMC.	application/xml
POST	This API is used to update the preferences set for a single SSP that is managedby the HMC.	application/xml

GET Response

/rest/api/pcm/preferences/SSP

```
<feed xmlns=http://www.w3.org/2005/Atom xmlns:ns2=http://a9.com/-/spec/opensearch/1.1/

          xmlns:ns3="http://www.w3.org/1999/xhtml">

 <id>FEED_UNIQUE_ID</id>

 <title type="text">Performance and Capacity Monitoring Preferences</title>

 <subtitle type="text"/>

 <link rel="self" href="https://HMC_IP_OR_HMC_HOST_NAME/rest/api/pcm/preferences/SSP"/>

 <generator uri="IBM Power Systems Management Console" version="1"/>
```

```
<entry>
 <id>ENTRY_UNIQUE_ID</id>
 <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>
 <title type="text">Performance and Capacity Monitoring Preferences</title>
 <published>YYYY-MM-DDThh:mm:ss.sssZ</published>
 <author>
  <name>IBM Power Systems Management Console</name>
 </author>
 <content type="application/xml">
 <ManagementConsolePCMSSPPreference:ManagementConsolePCMSSPPreference xmlns:ManagementConsolePCM
              SSPPreference=http://www.ibm.com/xmlns/systems/power/firmware/pcm/mc/2012_10/
              xmlns=http://www.ibm.com/xmlns/systems/power/firmware/pcm/mc/2012_10/
              xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_5_1">
  <Metadata>
   <Atom/>
  </Metadata>
  <ManagementConsoleSSPPreference kb="UOD" kxe="false" schemaVersion="V1_5_1">
   <Metadata>
   <Atom>
    <AtomID>SSP_UUID</AtomID>
    <AtomCreated>TIME in Milliseconds</AtomCreated>
   </Atom>
   </Metadata>
   <ClusterName kb="ROR" kxe="false">ClusterName</ClusterName>
   <ClusterUdId kxe="false" kb="ROR">ClusterUdId</ClusterUdId>
   <SSPName kb="ROR" kxe="false">SSPName</SSPName>
   <SSPUdId kxe="false" kb="ROR">SSPUdid</SSPUdId>
   <SSPUuId kb="ROR" kxe="false">SSPUuId</SSPUuId>
   <AggregationEnabled kxe="false" kb="UOD">true/false</AggregationEnabled>
   <MonitorEnabled kb="UOD" kxe="false">true/false</MonitorEnabled>
   <RemoteHMCCollectionStatus kb="ROR" kxe="false">true/false</RemoteHMCCollectionStatus>
   <AssociatedSharedStoragePool kxe="false" kb="ROR" href="https://HMC_IP_OR_HMC_HOST_NAME:
                           443/rest/api/uom/SharedStoragePool/SSP UUID" rel="related"/>
   <ManagedNodeSSPPreferences kb="ROR" kxe="false" schemaVersion="V1_5_1">
    <Metadata>
     <Atom/>
```

```
      </Metadata>
      <NodeMTMS kxe="false" kb="ROR">NodeMTMS</NodeMTMS>
      <NodeCollectionStatus kxe="false" kb="ROR">On/Off</NodeCollectionStatus>
      <NodeName kxe="false" kb="ROR">NodeName</NodeName>
      <NodeId kxe="false" kb="ROR">NodeId</NodeId>
      <NodeIPAddress kb="ROR" kxe="false">NodeIPAddress</NodeIPAddress>
     </ManagedNodeSSPPreferences>
    </ManagementConsoleSSPPreference>
   </ManagementConsolePCMSSPPreference:ManagementConsolePCMSSPPreference>
   </content>
  </entry>
</feed>
```

Successful execution of the request is returned with response code 200. In response, the application returns the following repeating attributes based on the number of SSPs, and the node or VIOS that are configured.

- **ManagementConsoleSSPPreference**: The SSP level preferences
  - **AtomId**: UUID of the SSP
  - **AtomCreated**: Creation time of the feed in milliseconds
  - **ClusterName**: Name of the cluster
  - **ClusterUUID**: A unique ID that is associated with the cluster
  - **SSPName**: Name of the SSP
  - **SSPUDID**: Unique ID of SSP from VIOS
  - **SSPUUID**: The UUID of SSP, this ID is used for further references
  - **AggregationEnabled**: This attribute indicates whether an Aggregation is enabled or not
  - **MonitorEnabled**: This attribute indicates whether any Collection is enabled or not
  - **RemoteHMCCollectionStatus**: This field indicates whether a different HMC is collecting the SSP data
  - **AssociatedSharedStoragePool**: Unique REST URI for the SSP
  - **ManagedNodeSSPPreferences**: The Node or VIOS that are connected to the HMC.
    - **NodeMTMS**: The MTMS of the Node or VIOS
    - **NodeCollectionStatus**: This attribute indicates whether the collection is enabled through this node
    - **NodeName**: Name of the Node or VIOS
    - **NodeId**: The ID of the Node in the SSP pool.
    - **NodeIPAddress**: IP address of the pool.

    Timestamp of the last available Aggregated metrics data

The **ManagementConsoleSSPPreference** properties repeat for every SSP that the HMC is managing.

The **ManagedNodeSSPPreferences** properties repeat for every connected Node or VIOS that is a part of the SSP.

/rest/api/pcm/SSP/*SSPUuId*/preferences

```
<feed xmlns=http://www.w3.org/2005/Atom xmlns:ns2=http://a9.com/-/spec/opensearch/1.1/
             xmlns:ns3="http://www.w3.org/1999/xhtml">
 <id>FEED_UNIQUE_ID</id>
 <title type="text">Performance and Capacity Monitoring Preferences</title>
 <subtitle type="text"/>
 <link rel="self" href="https://HMC_IP_OR_HMC_HOST_NAME/rest/api/pcm/preferences/SSP"/>
 <generator uri="IBM Power Systems Management Console" version="1"/>
 <entry>
  <id>ENTRY_UNIQUE_ID</id>
  <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>
  <title type="text">Performance and Capacity Monitoring Preferences</title>
  <published>YYYY-MM-DDThh:mm:ss.sssZ</published>
  <author>
   <name>IBM Power Systems Management Console</name>
  </author>
  <content type="application/xml">
   <ManagementConsoleSSPPreference:ManagementConsoleSSPPreference
                     xmlns:ManagementConsoleSSPPreference="http://www.ibm.com/xmlns/systems/power/firmware/
                     pcm/mc/2012_10/"
                     xmlns="http://www.ibm.com/xmlns/systems/power/firmware/pcm/mc/2012_10/"
                     xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_6_0">
    <Metadata>
     <Atom>
      <AtomID>SSP_UUID</AtomID>
      <AtomCreated>TIME in Milliseconds</AtomCreated>
     </Atom>
    </Metadata>
    <ClusterName kb="ROR" kxe="false">ClusterName</ClusterName>
    <ClusterUdId kxe="false" kb="ROR">ClusterUdId</ClusterUdId>
    <SSPName kb="ROR" kxe="false">SSPName</SSPName>
    <SSPUdId kxe="false" kb="ROR">SSPUdid</SSPUdId>
    <SSPUuId kb="ROR" kxe="false">SSPUuId</SSPUuId>
    <AggregationEnabled kxe="false" kb="UOD">true/false</AggregationEnabled>
    <MonitorEnabled kb="UOD" kxe="false">true/false</MonitorEnabled>
    <RemoteHMCCollectionStatus kb="ROR" kxe="false">true/false</RemoteHMCCollectionStatus>
    <AssociatedSharedStoragePool kxe="false" kb="ROR" href="https://HMC_IP_OR_HMC_HOST_NAME:
```

```
                              443/rest/api/uom/SharedStoragePool/SSP UUID" rel="related"/>
    <ManagedNodeSSPPreferences kb="ROR" kxe="false" schemaVersion="V1_5_1">
     <Metadata>
      <Atom/>
     </Metadata>
     <NodeMTMS kxe="false" kb="ROR">NodeMTMS</NodeMTMS>
     <NodeCollectionStatus kxe="false" kb="ROR">On/Off</NodeCollectionStatus>
     <NodeName kxe="false" kb="ROR">NodeName</NodeName>
     <NodeId kxe="false" kb="ROR">NodeId</NodeId>
     <NodeIPAddress kb="ROR" kxe="false">NodeIPAddress</NodeIPAddress>
    </ManagedNodeSSPPreferences>
   </ManagementConsoleSSPPreference:ManagementConsoleSSPPreference>
  </content>
 </entry>
</feed>
```

Sample payload for POST request

User can update only the **AggregationEnabled** and **MonitorEnabled** attributes. All the other attributes are read-only.

/rest/api/pcm/preferences/SSP

```
<ManagementConsolePCMSSPPreference:ManagementConsolePCMSSPPreference
             xmlns:ManagementConsolePCMSSPPreference=http://www.ibm.com/xmlns/systems/power/firmware/
             pcm/mc/2012_10/
             xmlns=http://www.ibm.com/xmlns/systems/power/firmware/pcm/mc/2012_10/
             xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_5_1">
 <Metadata>
  <Atom/>
 </Metadata>
 <ManagementConsoleSSPPreference kb="UOD" kxe="false" schemaVersion="V1_5_1">
  <Metadata>
   <Atom>
    <AtomID>SSP_UUID</AtomID>
    <AtomCreated>TIME in Milliseconds</AtomCreated>
   </Atom>
  </Metadata>
  <ClusterName kb="ROR" kxe="false">ClusterName</ClusterName>
  <ClusterUdId kxe="false" kb="ROR">ClusterUdId</ClusterUdId>
```

```
  <SSPName kb="ROR" kxe="false">SSPName</SSPName>
  <SSPUdId kxe="false" kb="ROR">SSPUdid</SSPUdId>
  <SSPUuId kb="ROR" kxe="false">SSPUuId</SSPUuId>
  <AggregationEnabled kxe="false" kb="UOD">true/false</AggregationEnabled>
  <MonitorEnabled kb="UOD" kxe="false">true/false</MonitorEnabled>
  <RemoteHMCCollectionStatus kb="ROR" kxe="false">true/false</RemoteHMCCollectionStatus>
  <AssociatedSharedStoragePool kxe="false" kb="ROR" href="https://HMC_IP_OR_HMC_HOST_NAME:
              443/rest/api/uom/SharedStoragePool/SSP UUID" rel="related"/>
  <ManagedNodeSSPPreferences kb="ROR" kxe="false" schemaVersion="V1_5_1">
   <Metadata>
    <Atom/>
   </Metadata>
   <NodeMTMS kxe="false" kb="ROR">NodeMTMS</NodeMTMS>
   <NodeCollectionStatus kxe="false" kb="ROR">On/Off</NodeCollectionStatus>
   <NodeName kxe="false" kb="ROR">NodeName</NodeName>
   <NodeId kxe="false" kb="ROR">NodeId</NodeId>
   <NodeIPAddress kb="ROR" kxe="false">NodeIPAddress</NodeIPAddress>
  </ManagedNodeSSPPreferences>
 </ManagementConsoleSSPPreference>
</ManagementConsolePCMSSPPreference:ManagementConsolePCMSSPPreference>
```

/rest/api/pcm/SSP/<SSPUuId>/preferences

```
<ManagementConsoleSSPPreference:ManagementConsoleSSPPreference xmlns:ManagementConsole
              SSPPreference="http://www.ibm.com/xmlns/systems/power/firmware/pcm/mc/2012_10/"
              xmlns="http://www.ibm.com/xmlns/systems/power/firmware/pcm/mc/2012_10/"
              xmlns:ns2="http://www.w3.org/XML/1998/namespace/k2" schemaVersion="V1_6_0">
 <Metadata>
  <Atom>
   <AtomID>SSP_UUID</AtomID>
   <AtomCreated>TIME in Milliseconds</AtomCreated>
  </Atom>
 </Metadata>
 <ClusterName kb="ROR" kxe="false">ClusterName</ClusterName>
 <ClusterUdId kxe="false" kb="ROR">ClusterUdId</ClusterUdId>
 <SSPName kb="ROR" kxe="false">SSPName</SSPName>
 <SSPUdId kxe="false" kb="ROR">SSPUdid</SSPUdId>
 <SSPUuId kb="ROR" kxe="false">SSPUuId</SSPUuId>
 <AggregationEnabled kxe="false" kb="UOD">true/false</AggregationEnabled>
```

```
 <MonitorEnabled kb="UOD" kxe="false">true/false</MonitorEnabled>

 <RemoteHMCCollectionStatus kb="ROR" kxe="false">true/false</RemoteHMCCollectionStatus>

 <AssociatedSharedStoragePool kxe="false" kb="ROR" href="https://HMC_IP_OR_HMC_HOST_NAME:

        443/rest/api/uom/SharedStoragePool/SSP UUID" rel="related"/>

 <ManagedNodeSSPPreferences kb="ROR" kxe="false" schemaVersion="V1_5_1">

  <Metadata>

   <Atom/>

  </Metadata>

  <NodeMTMS kxe="false" kb="ROR">NodeMTMS</NodeMTMS>

  <NodeCollectionStatus kxe="false" kb="ROR">On/Off</NodeCollectionStatus>

  <NodeName kxe="false" kb="ROR">NodeName</NodeName>

  <NodeId kxe="false" kb="ROR">NodeId</NodeId>

  <NodeIPAddress kb="ROR" kxe="false">NodeIPAddress</NodeIPAddress>

 </ManagedNodeSSPPreferences>

</ManagementConsoleSSPPreference:ManagementConsoleSSPPreference>
```

Raw Metrics for SSP Preference

The Raw metrics provides data collected from Virtual I/O Server (VIOS) that acts as a Node of the SSP and is connected to the Hardware Management Console (HMC). The Raw metrics are collected at every collection interval (5 minutes) and they are retained in the HMC for 30 minutes. After 30 minutes, they are purged from the system. This raw data is used to generate Processed and Aggregated metrics by further rolling up.

The collection of Raw metrics job runs every 5 minutes. It starts at xx:00 hours, continues to xx:05 hours, and so on.

Note: Any configuration changes to the SSP is reflected after 5 to 10 minutes.

Resource

```
/rest/api/pcm/SharedStoragePool/<SSPUuId>/RawMetrics
/rest/api/pcm/SharedStoragePool/<SSPUuId>/RawMetrics?

StartTS=yyyy-MM-dd'T'hh:mm:ssZ&EndTS= yyyy-MM-dd'T'hh:mm:ssZ
```

An example for date is 2017-05-22T18:43:29+0000

Quick properties

The API accepts start time stamp and the end time stamp as quick properties. If they are specified, the API responds within the specified duration.

Table 112. Quick properties

Quick property	Description	Format
StartTS	Start timestamp is an optional parameter. The API returns available LTM metrics from the specified time.	yyyy-MM-dd'T'HH:mm:ssZ

Table 112. Quick properties (continued)

Quick property	Description	Format
EndTS	End timestamp is an optional parameter. If specified, the API returns available LTM metrics on or before the specified time.	yyyy-MM-dd'T'HH:mm:ssZ

Table 113. Supported methods

Method	Description	Internet media type
GET	This API is used to retrieve the Raw metrics for an SSP that is connected to the HMC	`application/xml`

Response

```
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:ns2="http://a9.com/-/spec/opensearch/1.1/"

xmlns:ns3="http://www.w3.org/1999/xhtml">

 <id>FEED_UNIQUE_ID</id>

 <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>

 <title type="text">SharedStoragePool</title>

 <subtitle type="text">SharedStoragePool SSPUuiD</subtitle>

 <link rel="self" href="https://HMC_IP_OR_HOST_NAME/rest/api/pcm/SharedStoragePool/

SSPUuID/RawMetrics"/>

 <generator uri="IBM Power Systems Management Console" version="1"/>

 <entry>

  <id>ENTRY_UNIQUE_ID</id>

  <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>

  <title type="text">JSON file name</title>

  <published>YYYY-MM-DDThh:mm:ss.sssZ</published>

  <link type="application/vnd.ibm.powervm.pcm.json" href="https://

               HMC_IP_OR_HOST_NAME/rest/api/pcm/SharedStoragePool/

               SSPUuiD/RawMetrics/file name.json"/>

  <author>

   <name>IBM Power Systems Management Console</name>

  </author>

  <category term="ssp"/>

 </entry>

 <entry>

  <id>ENTRY_UNIQUE_ID</id>
```

```
  <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>

  <title type="text">JSON file name</title>

  <published>YYYY-MM-DDThh:mm:ss.sssZ</published>

  <link type="application/vnd.ibm.powervm.pcm.json"href="https://HMC_IP_OR_HOST_NAME

              /rest/api/pcm/SharedStoragePool/SSPUuiD/RawMetrics/file name.json"/>

  <author>

   <name>IBM Power Systems Management Console</name>

  </author>

  <category term="ssp"/>

 </entry>

</feed>
```

Successful execution of the request is returned with response code 200. In response the application returns Feed and Entries.

Each entry has the link to the raw metric data that is supplied in the JSON format. The following link provides the JSON specification for SSP RAW metrics: "JSON specification for the SSP RAW metrics" on page 163

Processed metrics for SSP

If Aggregation flag is enabled in SSP Preferences, the processed metrics for Shared Storage Pool (SSP) are generated for the specific processing duration of 5 minutes. Processed metrics give utilization data per time interval. These metrics are retained for maximum of 2 hours. After 2 hours, they are rolled up by the aggregation logic.

Processed metrics works on the Raw metrics and performs a series of computations on the raw data to generate the processed data. The processing of Raw metrics job runs every 5 minutes. It starts at xx:00 hours, continues to xx:05 hours, and so on. Processed metrics give utilization data per time interval (5 minutes).

Resource

```
/rest/api/pcm/SharedStoragePool/<SSPUuId>/ProcessedMetrics
/rest/api/pcm/SharedStoragePool/${SSPUuid}/ProcessedMetrics?

StartTS={StartTS}&EndTS={EndTS}&NoOfSamples={n}
```

An example for date is 2017-05-22T18:43:29+0000

Table 114. Quick properties

Quick property	Description	Format
StartTS	The start time stamp is an optional parameter. If specified, the API returns available Shared Storage Pool (SSP) Processed metrics from the specified time.	yyyy-MM-dd'T'HH:mm:ssZ

Table 114. Quick properties (continued)

Quick property	Description	Format
EndTS	The end time stamp is an optional parameter. If specified, the API returns available SSP Processed metrics on or before the specified time.	yyyy-MM-dd'T'HH:mm:ssZ
NoOfSamples	Number of samples is an optional parameter. If specified, the API returns the specified number of Processed metrics.	integer > 0

Table 115. Supported methods

Method	Description	Internet media type
GET	This API is used to retrieve the Atom link is a web link to the actual JSON file. The content can be retrieved by using the Atom Link.	`application/atom+xml`

Response

```
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:ns2="http://a9.com/-/spec/opensearch/1.1/"

             xmlns:ns3="http://www.w3.org/1999/xhtml">

 <id>FEED_UNIQUE_ID</id>

 <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>

 <title type="text">ProcessedMetrics</title>

 <subtitle type="text">SharedStoragePool SSPUuid</subtitle>

 <link rel="self" href="https://HMC_IP_OR_HOST_NAME/rest/api/pcm/

       SharedStoragePool/SSPUuid/ProcessedMetrics"/>

 <generator uri="IBM Power Systems Management Console" version="1"/>

 <entry>

  <id>ENTRY_UNIQUE_ID</id>

  <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>

  <title type="text">JSON file name</title>

  <published>YYYY-MM-DDThh:mm:ss.sssZ</published>

  <link type="application/json" href="https://HMC_IP_OR_HOST_NAME/rest/

              api/pcm/ProcessedMetrics/

      file name.json"/>

  <author>

   <name>IBM Power Systems Management Console</name>

  </author>
```

```
  <category term="SharedStoragePool" frequency="300"/>

 </entry>

</feed>
```

Successful execution of the request is returned with response code 200. In response the application returns Feed and Entry.

The Entry includes the link of the Processed data file that is in the JSON format. The following link provides the JSON specification for SSP Processed metrics: "JSON Specification for SSP Processed and Aggregated Metrics" on page 170.

Aggregated metrics for SSP

The Hardware Management Console (HMC) processes the processed metrics and performs data aggregation to provide access to the utilization data for longer duration.

The aggregated metrics are retained in HMC for the duration that can be configured by using the management console preferences. For more information, see "Management Console PCM Preferences" on page 124.

This API is used to retrieve the aggregated metrics for SSP. To retain the processed performance utilization data for longer duration, the data is aggregated. The levels of aggregation are:

- Tier 1 roll up: The first roll-up happens every 2 hours. It takes Processed metrics of last 2 hours and forms 1 aggregated metrics sample. The Tier 1 metrics are retained for 1 week. The roll up job starts at 00:01:30 hours and continues to 02:01:30 hours, 04:01:30 hours, and so on
- Tier 2 roll up: The tier 1 metrics are further rolled up to have one sample per day. The tier 2 metrics are retained for the retention duration that is set by Management Console preferences API. This roll up job starts each day at 00:02:00.

For example, in the rolled up data for the aggregated metric, the system maintains the values Average, Minimum, and Maximum, of the underlying processed metrics.

Resource

```
/rest/api/pcm/SharedStoragePool/<SSPUuId>/AggregatedMetrics
/rest/api/pcm/SharedStoragePool/<SSPUuId>/AggregatedMetrics?

StartTS={StartTS}&EndTS={EndTS}&NoOfSamples={n}
```

An example for date is 2017-05-22T18:43:29+0000

Table 116. Quick properties

Quick property	Description	Format
StartTS	The start time stamp is an optional parameter. If specified, the API returns available SSP Aggregated metrics from the specified time.	yyyy-MM-dd'T'HH:mm:ssZ
EndTS	The end time stamp is an optional parameter. If specified, the API returns available SSP Aggregated metrics on or before the specified time.	yyyy-MM-dd'T'HH:mm:ssZ
NoOfSamples	Number of samples is an optional parameter. If specified, the API returns the specified number of Processed metrics.	integer > 0

Table 117. Supported methods

Method	Description	Internet media type
GET	This API is used to retrieve the Atom link that is a web link to the actual JSON file. The content can be retrieved by using the Atom Link.	`application/xml`

Response

```
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:ns2=http://a9.com/-/

spec/opensearch/1.1/ xmlns:ns3="http://www.w3.org/1999/xhtml">

 <id>FEED_UNIQUE_ID</id>

 <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>

 <title type="text">AggregatedMetrics</title>

 <subtitle type="text">SharedStoragePool SSPUuId</subtitle>

 <link rel="self" href="https://HMC_IP_OR_HOST_NAME/rest/api/pcm/

        SharedStoragePool/SSPUuid/AggregatedMetrics"/>

 <generator uri="IBM Power Systems Management Console" version="1"/>

 <entry>

  <id>ENTRY_UNIQUE_ID</id>

  <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>

  <title type="text">JSON file name</title>

  <published>YYYY-MM-DDThh:mm:ss.sssZ</published>

  <link type="application/json" href=" https://HMC_IP_OR_HOST_NAME/rest/api/

               pcm/AggregatedMetrics/file name.json "/>

  <author>

   <name>IBM Power Systems Management Console</name>

  </author>

  <category term="SharedStoragePool" frequency="7200"/>

 </entry>

 <entry>

  <id>ENTRY_UNIQUE_ID</id>

  <updated>YYYY-MM-DDThh:mm:ss.sssZ</updated>

  <title type="text">JSON file name</title>

  <published>YYYY-MM-DDThh:mm:ss.sssZ</published>

  <link type="application/json" href=" https://HMC_IP_OR_HOST_NAME/rest/api/pcm
```

```
              /AggregatedMetrics/file name.json "/>
  <author>
   <name>IBM Power Systems Management Console</name>
  </author>
  <category term="SharedStoragePool" frequency="86400"/>
 </entry>
</feed>
```

Successful execution of the request is returned with response code 200. In response the application returns Feed and Entries for each Tier.

The Feed includes file links for each Aggregated metrics per Tier. The metric data is supplied in the JSON format. The following link provides the JSON specification for SSP Aggregated metrics: "JSON Specification for SSP Processed and Aggregated Metrics" on page 170.

JSON specification for the SSP RAW metrics

The Shared Storage Pool (SSP) Raw metric URI provides the utilization data that is retrieved by the Hardware Management Console (HMC) from the Virtual IO Server (Node) of the SSP. The data can be obtained when the **MonitorEnabled** attribute for an SSP is enabled in the collection preferences.

Resource

```
/rest/api/pcm/SharedStoragePool/<SSPUuId>/RawMetrics/SSPLTM_ssp_pool_<SSPUuid>_yyyyMMDDThhmmss+0000.json
```

Response

```
{
 "systemUtil":{
  "utilInfo":{
   "version":"String",
   "metricType":"String",
   "monitoringType":"String",
   "mtms":"String"
  },
  "utilSample":{
   "timeStamp":"String",
   "status":"Number",
   "errorInfo":[{
    "errId":"String",
    "errMsg":"String"
   }],
   "sharedStoragePoolUtil":[{
    "name":"String",
```

```
"poolId":"String",
"clusterName":"String",
"clusterId":"String",
"size":"Number",
"free":"Number",
"numOfReads":"Number",
"numOfReadTransfers":"Number",
"numofReadRequestTimeouts":"Number",
"numofReadRequestFailures":"Number",
"numOfWrites":"Number",
"numOfWriteTransfers":"Number",
"numofWriteRequestTimeouts":"Number",
"numofWriteRequestFailures":"Number",
"readBytes":"Number",
"writeBytes":"Number",
"readServiceTime":"Number",
"minReadServiceTime":"Number",
"maxReadServiceTime":"Number",
"writeServiceTime":"Number",
"minWriteServiceTime":"Number",
"maxWriteServiceTime":"Number",
"nodeUtil":[{
 "id":"String",
 "name":"String",
 "mtms":"String",
 "state":"String",
 "poolState":"String",
 "size":"Number",
 "free":"Number",
 "numOfReads":"Number",
 "numOfReadTransfers":"Number",
 "numofReadRequestTimeouts":"Number",
 "numofReadRequestFailures":"Number",
```

```
"numOfWrites":"Number",
"numOfWriteTransfers":"Number",
"numofWriteRequestTimeouts":"Number",
"numofWriteRequestFailures":"Number",
"readBytes":"Number",
"writeBytes":"Number",
"readServiceTime":"Number",
"minReadServiceTime":"Number",
"maxReadServiceTime":"Number",
"writeServiceTime":"Number",
"minWriteServiceTime":"Number",
"maxWriteServiceTime":"Number",
"tierUtil":[{
 "id":"String",
 "name":"String",
 "size":"Number",
 "free":"Number",
 "numOfReads":"Number",
 "numOfReadTransfers":"Number",
 "numofReadRequestTimeouts":"Number",
 "numofReadRequestFailures":"Number",
 "numOfWrites":"Number",
 "numOfWriteTransfers":"Number",
 "numofWriteRequestTimeouts":"Number",
 "numofWriteRequestFailures":"Number",
 "readBytes":"Number",
 "writeBytes":"Number",
 "readServiceTime":"Number",
 "minReadServiceTime":"Number",
 "maxReadServiceTime":"Number",
 "writeServiceTime":"Number",
 "minWriteServiceTime":"Number",
 "maxWriteServiceTime":"Number",
 "failuregroupUtil":[{
```

```
"id":"String",
"name":"String",
"size":"Number",
"free":"Number",
"numOfReads":"Number",
"numOfReadTransfers":"Number",
"numofReadRequestTimeouts":"Number",
"numofReadRequestFailures":"Number",
"numOfWrites":"Number",
"numOfWriteTransfers":"Number",
"numofWriteRequestTimeouts":"Number",
"numofWriteRequestFailures":"Number",
"readBytes":"Number",
"writeBytes":"Number",
"readServiceTime":"Number",
"minReadServiceTime":"Number",
"maxReadServiceTime":"Number",
"writeServiceTime":"Number",
"minWriteServiceTime":"Number",
"maxWriteServiceTime":"Number",
"diskUtil":[{
 "name":"String",
 "id":"String",
 "uid":"String",
 "adapterName":"String",
 "size":"Number",
 "free":"Number",
 "numOfReads":"Number",
 "numOfReadTransfers":"Number",
 "numofReadRequestTimeouts":"Number",
 "numofReadRequestFailures":"Number",
 "numOfWrites":"Number",
 "numOfWriteTransfers":"Number",
```

```
        "numofWriteRequestTimeouts":"Number",
        "numofWriteRequestFailures":"Number",
        "readBytes":"Number",
        "writeBytes":"Number",
        "readServiceTime":"Number",
        "minReadServiceTime":"Number",
        "maxReadServiceTime":"Number",
        "writeServiceTime":"Number",
        "minWriteServiceTime":"Number",
        "maxWriteServiceTime":"Number",
        "timeSpentInWaitQueue":"Number",
        "waitQueueSize":"Number",
        "totalWaitQueueSize":"Number",
        "minTimeSpentInWaitQueue":"Number",
        "maxTimeSpentInWaitQueue":"Number",
        "numOfTimesServiceQueueIsFull":"Number",
        "totalServiceQueueDepth":"Number"
       }]
      }]
     }]
    }]
   }]
  }
 }
}
```

The following table provides the details the metrics.

Table 118. Descriptions of the metric parameters.

Metric parameter	Description
Version	Version of JSON format for SSP. The value is 1.0.0 for the first release.
metricType	Indicates whether the values are Raw, Processed, or Aggregated metrics. ForRraw metrics, the value is Raw.
monitoringType	This indicates the type of monitoring. For SSP the value is SSP.
mtms	Managed system model type and serial number.
timeStamp	Timestamp of the sample.

Table 118. Descriptions of the metric parameters (continued).

Metric parameter	Description
status	Status can have one of the following values: • 0: all collections are successful • 1: all collections have failed • 2: part of the collection is successful
SSPUtil: Metrics in this tag should be the total of all its Nodes utilization data.	
poolId	SSP ID.
name	SSP name.
clusterName	Name of the cluster.
clusterId	UDID of the cluster.
size	Total size of the disks that belong to the SSP in megabytes.
Free	Total free size of the disks that belong to the SSP in megabytes.
DISK METRICS Utilization	The total of the SSP disks utilization. See Disk metrics table for details on each utilization metric.
NodeUtil: Metrics in this tag should be the total of all the SSP Disks utilization in this Node.	
id	VIOS node ID.
mtms	Machine and serial number.
name	VIOS node name.
state	State of the VIOS node. The values are Running and Not Running.
poolState	State of the pool in this node. The values are UP or DOWN.
size	Total size of the disks that belong to the node in megabytes.
Free	Total free size of the disks that belong to the node in megabytes.
DISK METRICS Utilization	The total of all SSP disks utilization in the node. See Disk metric table for details on each utilization metric
TierUtil	
id	Tier ID.
name	Tier name.
Size	Total size of disks in the tier in megabytes. This includes all the disks that belong to the failure groups of the tier.
free	Free size of this disks in the tier in megabytes. This includes all the disks that belong to the failure groups of the tier.
DISK METRICS Utilization	The total of all disks utilization in the tier. See Dis metrics table for details on each utilization metric
FailureGroupUtil	
id	Failure group ID.
name	Failure group name.
size	Total size of disks in the failure group in megabytes.
free	Size of free disks in the failure group in megabytes.

Table 118. Descriptions of the metric parameters (continued).

Metric parameter	Description
DISK METRICS UTILIZATION	The total of all disks utilization metrics in the failure group. See Disk metrics table for details on each utilization metric.
DiskUtil	
udid	Disk unique device identifier.
Uuid	Unique universal identifier.
name	Disk name
adapterName	Name of the adapter to which disk belongs to.
Size	Size of the disk in megabytes.
free	Size of the free disks in megabytes.
DISK METRICS Utilization	Disk utilization metrics of the disk. See Disk metric table for details on each utilization metric.

The following table provides the details the metrics.

Table 119. Descriptions of the Disk metric parameters.

Metric parameter	Description
numofReads	Total number of reads.
numofWrites	Total number of writes.
readBytes	Total bytes read.
writeBytes	Total bytes written.
serviceQueueDepth	Number of times a request is sent to the disks that are not completed yet. This parameter is available only in DiskUtil metric.
numofTimesServiceQueueIsFull	Number of times when the service queue full event has occurred. This parameter is available only in DiskUtil metric.
readServiceTime	Read service time in nanoseconds.
numofReadRequestTimeouts	Number of read request timeouts.
numofReadRequestFailures	Number of failed read requests.
minReadServiceTime	Minimum read service time in nanoseconds.
writeServiceTime	Write Service time in nanoseconds.
minWriteServiceTime	Minimum write service time in Nanoseconds.
maxWriteServiceTime	Maximum write service time in Nanoseconds.
numofWriteRequestTimeouts	Number of write request timeouts.
numofWriteRequestFailures	Number of failed write requests .
waitQueueSize	Number of requests waiting to be sent to disk. This parameter is available only in DiskUtil metric.
totalWaitQueuesize	Accumulated sampled wait queue depth. This parameter is available only in DiskUtil metric.
timeSpentInWaitQueue	Accumulated wait queue time. This parameter is available only in DiskUtil metric.
minTimeSpentInWaitQueue	Minimum wait queue time. This parameter is available only in DiskUtil metric.

Table 119. Descriptions of the Disk metric parameters (continued).

Metric parameter	Description
maxTimeSpentInWaitQueue	Maximum wait queue time. This parameter is available only in DiskUtil metric.
TotalServiceQueueDepth	Accumulated service queue depth. This parameter is available only in DiskUtil metric.
NumOfReadTransfers	Number of read transfers.
NumOfWriteTransfers	Number of write transfers.

JSON Specification for SSP Processed and Aggregated Metrics

The JSON specification for the Shared Storage Pool (SSP) processed and aggregated metrics. All the parameters for the Processed and Aggregated metrics are same except that the Processed metrics provides only Average value, while Aggregated metrics provide Average, Maximum, and Minimum values.

PCM Processed or aggregated metrics URI provides the following SSP utilization data when the **AggregationEnabled** attribute for an SSP is enabled.

Resource

```
/rest/api/pcm/ProcessedMetrics/SharedStoragePool_[SSPUuid]_yyyyMMDDThhmmss+0000_300.json
/rest/api/pcm/AggregatedMetrics/SharedStoragePool_[SSPUuiD]_yyyyMMDDThhmmss+0000_

[Tier frequency in seconds].json
```

Response

```
{

 "systemUtil":{

  "utilInfo":{

   "version":"String",

   "metricType":"String",

   "frequency":"Number",

   "startTimeStamp":"String",

   "endTimeStamp":"String",

   "uuid":"String",

   "name":"String",

   "poolId":"String",

   "clusterId":"String",

   "clusterName":"String",

   "clusterUuid":"String",

   "metricArrayOrder":[ "AVG", "MIN", "MAX"]

  },

  "utilSamples":[{

   "poolUtil":{
```

```
"nodeUtil":[{
 "id":"Number",
 "name":"String",
 "mtms":"String",
 "poolState":"String",
 "tierUtil":[{
  "id":"Number",
  "name":"SYSTEM",
  "failureGrpUtil":[{
   "id":"String",
   "name":"Default",
   "diskUtil":[{
    "name":"String",
    "id":"String",
    "uid":"String",
    "adapterName":"String",
    "timeSpentInWaitQueue":[ "Float", "Float", "Float" ],
    "numofTimesServiceQueueIsFull":[ "Float", "Float", "Float" ],
    "size":["Float", "Float", "Float"],
    "free":["Float", "Float", "Float" ],
    "numOfReads":["Float", "Float", "Float" ],
    "numOfWrites":["Float", "Float", "Float" ],
    "readBytes":["Float", "Float", "Float"],
    "writeBytes":["Float", "Float", "Float" ],
    "transmittedBytes":["Float", "Float", "Float" ],
    "numOfReadRequestTimeouts":["Float", "Float", "Float"],
    "numOfReadRequestFailures":["Float", "Float", "Float"],
    "numOfWriteRequestTimeouts":["Float", "Float", "Float" ],
    "numOfWriteRequestFailures":["Float", "Float", "Float" ],
    "numOfReadTransfers":["Float", "Float", "Float" ],
    "numOfWriteTransfers":["Float", "Float", "Float" ],
    "readServiceTime":["Float", "Float", "Float" ],
    "writeServiceTime":["Float", "Float", "Float" ]
```

```
  }],
  "size":["Float", "Float", "Float" ],
  "free":["Float", "Float", "Float" ],
  "numOfReads":["Float", "Float", "Float" ],
  "numOfWrites":["Float", "Float", "Float"],
  "readBytes":["Float", "Float", "Float" ],
  "writeBytes":["Float", "Float", "Float" ],
  "transmittedBytes":["Float", "Float", "Float" ],
  "numOfReadRequestTimeouts":["Float", "Float", "Float" ],
  "numOfReadRequestFailures":["Float", "Float", "Float" ],
  "numOfWriteRequestTimeouts":["Float", "Float", "Float" ],
  "numOfWriteRequestFailures":["Float", "Float", "Float" ],
  "numOfReadTransfers":["Float", "Float", "Float" ],
  "numOfWriteTransfers":["Float", "Float", "Float" ],
  "readServiceTime":["Float", "Float", "Float" ],
  "writeServiceTime":["Float", "Float", "Float" ]
 }],
 "size":["Float", "Float", "Float" ],
 "free":["Float", "Float", "Float" ],
 "numOfReads":["Float", "Float", "Float" ],
 "numOfWrites":["Float", "Float", "Float" ],
 "readBytes":["Float", "Float", "Float" ],
 "writeBytes":["Float", "Float", "Float"],
 "transmittedBytes":["Float", "Float", "Float" ],
 "numOfReadRequestTimeouts":["Float", "Float", "Float" ],
 "numOfReadRequestFailures":["Float", "Float", "Float"],
 "numOfWriteRequestTimeouts":["Float", "Float", "Float"],
 "numOfWriteRequestFailures":["Float", "Float", "Float"],
 "numOfReadTransfers":["Float", "Float", "Float"],
 "numOfWriteTransfers":["Float", "Float", "Float"],
 "readServiceTime":["Float", "Float", "Float"],
 "writeServiceTime":["Float", "Float", "Float"]
}],
"size":["Float", "Float", "Float"],
```

```
  "free":["Float", "Float", "Float"],
  "numOfReads":["Float", "Float", "Float"],
  "numOfWrites":["Float", "Float", "Float"],
  "readBytes":["Float", "Float", "Float"],
  "writeBytes":["Float", "Float", "Float"],
  "transmittedBytes":["Float", "Float", "Float"],
  "numOfReadRequestTimeouts":["Float", "Float", "Float"],
  "numOfReadRequestFailures":["Float", "Float", "Float"],
  "numOfWriteRequestTimeouts":["Float", "Float", "Float"],
  "numOfWriteRequestFailures":["Float", "Float", "Float"],
  "numOfReadTransfers":["Float", "Float", "Float"],
  "numOfWriteTransfers":["Float", "Float", "Float"],
  "readServiceTime":["Float", "Float", "Float"],
  "writeServiceTime":["Float", "Float", "Float"]
 }],
 "size":["Float", "Float", "Float"],
 "free":["Float", "Float", "Float"],
 "numOfReads":["Float", "Float", "Float"],
 "numOfWrites":["Float", "Float", "Float"],
 "readBytes":["Float", "Float", "Float"],
 "writeBytes":["Float", "Float", "Float"],
 "transmittedBytes":["Float", "Float", "Float"],
 "numOfReadRequestTimeouts":["Float", "Float", "Float"],
 "numOfReadRequestFailures":["Float", "Float", "Float"],
 "numOfWriteRequestTimeouts":["Float", "Float", "Float"],
 "numOfWriteRequestFailures":["Float", "Float", "Float"],
 "numOfReadTransfers":["Float", "Float", "Float"],
 "numOfWriteTransfers":["Float", "Float", "Float"],
 "readServiceTime":["Float", "Float", "Float"],
 "writeServiceTime":["Float", "Float", "Float"]
},
"sampleInfo":{
 "timeStamp":"String",
```

```
    "numOfSamplesAggregated":"Number",
    "status":"Number"
   }
  }]
 }
}
```

The following tables provides the details of the metrics.

Table 120. Descriptions of the metric parameters.

Metric parameter	Description
Version	Version of JSON format for SSP. The value is 1.0.0 for the first release.
metricType	This indicates whether the values are Raw, Processed, or Agregated metrics.
frequency	Indicates the collection cycle time in seconds.
startTimeStamp	Start timestamp of the samples collected.
endTimeStamp	End timestamp of the samples collected.
uuid	UUID of the SSP.
name	SSP name.
poolId	Pool ID.
clusterName	Cluster name.
clusterId	Cluster ID.
clusterUuid	Cluster UUID.
metricArrayOrder	The values are AVG, MIN, and MAX. For Processed metrics, the value is AVG.
utilSamples: poolUtil: Containing sample of Processed and Aggregated SSP metrics.	
size	Average size of the disks that belong to the SSP in megabytes.
Free	Total free size of the disks that belong to the SSP in megabytes.
DISK METRICS Utilization	Total of all disks utilization metrics in the specific failure group. See the Disk metrics table for details on each utilization metric.
utilSamples: poolUtil: NodeUtil: Metrics in this tag should be the total of all the SSP Disks utilization in this Node.	
id	VIOS node ID.
mtms	Machine and serial number.
name	VIOS node name.
state	State of the VIOS node. The values are Running and Not Running.
poolState	State of the pool in this node. The values are UP or DOWN.
FailureGroupUtil	
id	Failure group ID.
name	Failure group Name.

Table 120. Descriptions of the metric parameters (continued).

Metric parameter	Description
size	Size of the disks that belong to this node in megabytes.
free	Free size of the disks that belong to this node in megabytes.
DISK METRICS UTILIZATION	The total of all disks utilization metrics in this failure group. See the Disk metrics table for details on each utilization metric.
utilSamples: poolUtil: NodeUtil:TierUtil	
Id	Tier ID.
Name	Tier name.
Size	Total size of disks in the tier in megabytes. This does not include all the disks that belongs to the failure groups of the tier.
Free	Free size of the disks in this tier in megabytes. This includes all the disks that belongs to the failure groups of the tier.
DISK METRICS Utilization	The total of all disks utilization in this tier. See the Disk metrics table for details on each utilization metric.
utilSamples: poolUtil: NodeUtil: TierUtil :FailureGroupUtil	
Id	Failure group ID.
Name	Failure group name.
Size	Size of disks in the failure group in bytes.
Free	Free size of the disks in this failure group in bytes.
DISK METRICS UTILIZATION	The total of all the disks utilization metrics in this failure group. See the Disk metrics table for details on each utilization metric.
utilSamples: poolUtil: NodeUtil: TierUtil :FailureGroupUtil DiskUtil	
id	Disk-unique device identifier.
UID	Unique universal identifier.
Name	Disk name.
adapterName	Name of the adapter to which the disk belongs.
Size	Size of the disk in megabytes.
free	Free size of the disk in megabytes.
DISK METRICS Utilization	Disk Utilization metrics of this disk. See the Disk metric table for details on each utilization metric.
utilSamples: sampleInfo	
timeStamp	Timestamp of the sample
numOfSamplesAggregated	This attribute is only present for Aggregated metrics.
status	Status can have one of the following values: • 0: all collections are successful • 1: all collections have failed • 2: part of the collection is successful

Table 121. Descriptions of the Disk metric parameters.

Metric parameter	Description
numofReads	Total number of reads.
numofWrites	Total number of writes.
readBytes	Total bytes read.
writeBytes	Total bytes written.
serviceQueueDepth	Number of times a request is sent to the disks that are not completed yet. This parameter is available only in DiskUtil metric.
numofTimesServiceQueueIsFull	Number of times when the service queue full event has occurred. This parameter is available only in DiskUtil metric.
readServiceTime	Read service time in nanoseconds.
numofReadRequestTimeouts	Number of read request timeouts.
numofReadRequestFailures	Number of failed read requests.
minReadServiceTime	Minimum read service time in nanoseconds.
writeServiceTime	Write Service time in nanoseconds.
minWriteServiceTime	Minimum write service time in Nanoseconds.
maxWriteServiceTime	Maximum write service time in Nanoseconds.
numofWriteRequestTimeouts	Number of write request timeouts.
numofWriteRequestFailures	Number of failed write requests .
waitQueueSize	Number of requests waiting to be sent to disk. This parameter is available only in DiskUtil metric.
totalWaitQueuesize	Accumulated sampled wait queue depth. This parameter is available only in DiskUtil metric.
timeSpentInWaitQueue	Accumulated wait queue time. This parameter is available only in DiskUtil metric.
minTimeSpentInWaitQueue	Minimum wait queue time. This parameter is available only in DiskUtil metric.
maxTimeSpentInWaitQueue	Maximum wait queue time. This parameter is available only in DiskUtil metric.
TotalServiceQueueDepth	Accumulated service queue depth. This parameter is available only in DiskUtil metric.
NumOfReadTransfers	Number of read transfers.
NumOfWriteTransfers	Number of write transfers.

Error codes related to SSP monitoring

The table provides the detail of the error codes from the collection provider framework in the node (VIOS) and the description of error codes for SSP monitoring.

Table 122. Error codes and descriptions.

Error code	Description
101	Connection Refused: Too many requests
102	Unable to communicate with perfprovider (Node collection)
103	Problem with VIOS

Table 122. Error codes and descriptions (continued).

Error code	Description
201	Perfprovider (Node collector) not Running
210	Failure to execute the command <incorrect Arguments>
213	Failure to Start SSP monitoring
214	Not a SSP Node
215	SSP is down
216	Failure collecting samples of SSP monitoring
217	SSP monitoring not running

Note: Configuration changes to the SSP are reflected after 5 - 10 minutes.

Performance and Capacity Monitoring JSON Specification

Performance and Capacity (PCM) generates utilization metrics using JSON format. This section provide specification for the utilization metrics.

Note: During data collection, the error codes get logged in the JSON response. The error codes provide the status of data collection.

- Status 0: all collections were success
- Status 1: all collections failed
- Status 2: a part of the collection was success

Related tasks:

"STM Virtual I/O Server JSON Specification" on page 188
Virtual I/O Server JSON Specification for Short Term Monitoring metrics

Related reference:

"LTM Power Hypervisor JSON Specification" on page 178
Power Hypervisor JSON Specification for Long Term Monitoring metrics

"LTM Virtual IO Server JSON Specification" on page 182
Virtual IO Server JSON Specification for Long Term Monitoring metrics

"JSON specification for Energy monitoring RAW metrics" on page 146
The PCM Raw metric for Energy monitoring URI provides the following utilization data that is fetched by the Hardware Management Console (HMC) from the Flexible Service Processor (FSP) when the **EnergyMonitorEnabled** parameter is enabled for a managed system.

"STM Power Hypervisor JSON Specification" on page 184
Power Hypervisor JSON Specification for Short Term Monitoring (STM) metrics

"Managed System Processed and Aggregated Metrics JSON Specification" on page 191
The JSON specification for the managed system processed and aggregated metrics. All the parameters for the processed and aggregated metrics are same except that the Processed metrics provides only Average metrics. It does not provide minimum and maximum records.

"Logical Paritition Processed/Aggregated Metrics JSON Specification" on page 199
The JSON specification for the logical partition Processed and Aggregated metrics. All the parameters for the Processed and Aggregated metrics are same except that the Processed metrics provides only Average metrics. It does not provide Minimum and Maximum records.

"JSON specification for Energy Monitoring Processed and Aggregated Metrics" on page 148
All the parameters for the Processed and Aggregated metrics are same except that the Processed metrics provides only Average metrics, and it does not provide minimum and maximum records. The PCM Processed or Aggregated metrics URI for Energy Monitoring provides the following power consumption and thermal data when the **AggregationEnabled** parameter is enabled for a managed system.

"JSON Specification for SSP Processed and Aggregated Metrics" on page 170
The JSON specification for the Shared Storage Pool (SSP) processed and aggregated metrics. All the parameters for the Processed and Aggregated metrics are same except that the Processed metrics provides only Average value, while Aggregated metrics provide Average, Maximum, and Minimum values.

"JSON specification for the SSP RAW metrics" on page 163
The Shared Storage Pool (SSP) Raw metric URI provides the utilization data that is retrieved by the Hardware Management Console (HMC) from the Virtual IO Server (Node) of the SSP. The data can be obtained when the **MonitorEnabled** attribute for an SSP is enabled in the collection preferences.

LTM Power Hypervisor JSON Specification

Power Hypervisor JSON Specification for Long Term Monitoring metrics

PCM LTM URI provides the following utilization data that is fetched by the Hardware Management Console (HMC) from the Power Hypervisor (PHYP) when the **LongTermMonitorEnabled** configuration value for a managed system is enabled. If **ComputeLTMEnabled** is enabled, the application provides processor and memory metrics; it would not provide network and storage metrics.

```
{
 "systemUtil": {
  "utilInfo": {
   "version": "string",
   "metricType": "string",
   "monitoringType": "string",
   "mtms" : "string",
   "name" : "string"
  },
  "utilSample":
  {
   "timeStamp": "string",
   "status": "number",
   "errorInfo" : [
   { "errId" : "string", "errMsg" : "string"  }
   ],
   "timeBasedCycles" : "number",
   "systemFirmware" :
   {
    "utilizedProcCycles" : "number",
    "assignedMem" : "number"
   },
   "processor" :
   {
    "totalProcUnits" : "number",
    "configurableProcUnits" : "number",
    "availableProcUnits": "number",
    "procCyclesPerSecond" : "number"
   },
   "memory" :
   {
    "totalMem" : "number",
    "availableMem" : "number",
    "configurableMem" : "number"

   },
   "sharedMemoryPool" :
   [
   {
    "id" : "number",
    "name" : "string"
    "totalMem" : "number",
    "assignedMemToLpars" : "number",
    "assignedMemToSysFirmware" : "number",
    "totalIOMem" : "number",
    "mappedIOMemToLpars" : "number"
   }
```

```
],
"sharedProcessorPool":
[
{
 "id" : "number",
 "name" : "string",
 "assignedProcCycles" : "number",
 "utilizedProcCycles" : "number",
 "maxProcUnits" : "number",
 "borrowedProcUnits": "number"
}
],
"network" :
{
 "sriovAdapters":
 [
 {
  "drcIndex"  : "string",
  "physicalPorts" :
  [
  {
   "id": "number",
   "physicalLocation" : "string",
   "receivedPackets" : "number",
   "sentPackets" :  "number",
   "droppedSentPackets" : "number",
   "droppedReceivedPackets" : "number",
   "sentBytes" : "number",
   "receivedBytes" : "number",
   "errorIn" : "number",
   "errorOut" "number"
  }
  ]
 }
 ],
 "HEAdapters": [
 {
  "drcIndex"  : "string",
  "physicalLocation" " "string",
  "physicalPorts" :
  [
  {
   "id": "number",
   "physicalLocation" : "string",
   "receivedPackets" : "number",
   "sentPackets" :  "number",
   "droppedPackets" : "number",
   "sentBytes" : "number",
   "receivedBytes" : "number"
  }
  ]
 }
 ]
},
"lparsUtil" :
[
{
 "id" : "number",
 "uuid" : "string",
 "type" : "string",
 "name":  "string",
 "state" : "string",
 "affinityScore" :  "number",
 "memory":
 {
  "poolId" : "number",
  "weight" : "number",
```

```
 "logicalMem" : "number",
 "backedPhysicalMem" : "number",
 "totalIOMem" : "number",
 "mappedIOMem" : "number"
},
"processor":
{
 "poolId" :  "number",
 "mode"   :  "string"
 "maxVirtualProcessors" : "number",
 "maxProcUnits" : "number",
 "weight" : "number",
 "entitledProcCycles": "number",
 "utilizedCappedProcCycles" :"number",
 "utilizedUnCappedProcCycles" : "number",
 "idleProcCycles" : "number",
 "donatedProcCycles" : "number",
 "timeSpentWaitingForDispatch" : "number",
 "totalInstructions" : "number",
 "totalInstructionsExecutionTime" : "number"
},
"network" :
{
 "virtualEthernetAdapters":[
 {
  "vlanId" :  "number",
  "vswitchId" :  "number",
  "physicalLocation" : "string",
  "isPortVLANID" : "boolean",
  "receivedPackets" : "number",
  "sentPackets" :  "number",
  "droppedPackets" : "number",
  "sentBytes" : "number",
  "receivedBytes" : "number",
  "receivedPhysicalPackets" : "number",
  "sentPhysicalPackets" :  "number",
  "droppedPhysicalPackets" : "number",
  "sentPhysicalBytes" : "number",
  "receivedPhysicalBytes" : "number"
 }
 ],
 "sriovLogicalPorts": [
 {
  "drcIndex"  : "string",
  "physicalDrcIndex" : "string",
  "physicalPortId" :  "number",
  "physicalLocation" : "string",
  "receivedPackets" : "number",
  "sentPackets" :  "number",
  "droppedSentPackets" : "number",
  "droppedReceivedPackets" : "number",
  "sentBytes" : "number",
  "receivedBytes" : "number",
  "errorIn" : "number",
  "errorOut" : "number"
 }
 ]
},
"Storage":
{
 "virtualFiberChannelAdapters":
 [
 {
  "viosId"  : number,
  "wwpnPair":["wwpn1" ,"wwpn2" ],
  "physicalLocation" : "string"
 }
```

```
  ] ,
  "genericVirtualAdapters":
  [
  {
   "physicalLocation" : "string",
   "viosId" :  "number",
   "viosAdapterSlotId"  : "number"
  }
  ]
 }
}
],
"viosUtil":
[
{
 "id" :"number",
 "uuid":  "string",
 "name":  "string",
 "state" : "string",
 "affinityScore" : "number",
 "memory" :
 {
  "assignedMem" : "number"
 }
 "processor":
 {
  "poolId" :  "number",
  "mode"   : "string",
  "maxVirtualProcessors" : "number",
  "maxProcUnits" : "number",
  "weight" : "number",
  "entitledProcCycles" : "number",
  "utilizedCappedProcCycles" :"number",
  "utilizedUnCappedProcCycles" : "number",
  "idleProcCycles" : "number",
  "donatedProcCycles" : "number",
  "timeSpentWaitingForDispatch" : "number",
  "totalInstructions" : "number",
  "totalInstructionsExecutionTime" : "number"
 },
 "network" :
 {
  "virtualEthernetAdapters":[
  {
   "vlanId" :  "number",
   "vswitchId" :  "number",
   "physicalLocation" : "string",
   "isPortVLANID" : "boolean",
   "receivedPackets" : "number",
   "sentPackets" :  "number",
   "droppedPackets" : "number",
   "sentBytes" : "number",
   "receivedBytes" : "number",
   "receivedPhysicalPackets" : "number",
   "sentPhysicalPackets" :  "number",
   "droppedPhysicalPackets" : "number",
   "sentPhysicalBytes" : "number",
   "receivedPhysicalBytes" : "number"
  }
  ],
  "sriovLogicalPorts": [
  {
   "drcIndex"  : "string",
   "physicalDrcIndex" : "string",
   "physicalPortId" :  "number",
   "physicalLocation" : "string",
   "receivedPackets" : "number",
```

```
      "sentPackets" :  "number",
      "droppedSentPackets" : "number",
      "droppedReceivedPackets" : "number",
      "sentBytes" : "number",
      "receivedBytes" : "number",
      "errorIn" : "number",
      "errorOut" "number"
     }
     ]
    }
   }
   ]
  }
 }
}
```

LTM Virtual IO Server JSON Specification

Virtual IO Server JSON Specification for Long Term Monitoring metrics

PCM LTM URI provides the following utilization data that is fetched by the Hardware Management Console (HMC) from the Virtual I/O Server (VIOS0 when the **LongTermMonitorEnabled** configuration value for a managed system is enabled.

```
{
 "systemUtil": {
  "utilInfo": {
   "version": "string",
   "metricType": "string",
   "monitoringType": "string",
   "mtms": "string",
  },
  "utilSample": {
   "timeStamp": "string",
   "status": "number",
   "errorInfo": [
   {
    "errId": "string",
    "errMsg": "string"
   },
   ],
   "viosUtil": [
   {
    "id": "string",
    "name" : "string",
    "memory": {
     "utilizedMem": "number",
    },
    "network": {
     "genericAdapters": [
     {
      "id": "string",
      "type": "string",
      "physicalLocation" : "string",
      "receivedPackets": "number",
      "sentPackets": "number",
      "droppedPackets": "number",
      "sentBytes": "number",
      "receivedBytes": "number"
     }
     ],
     "sharedAdapters": [
     {
      "id": "string",
      "type": "string",
      "physicalLocation" : "string",
      "receivedPackets": "number",
```

```
  "sentPackets": "number",
  "droppedPackets": "number",
  "sentBytes": "number",
  "receivedBytes": "number",
  "bridgedAdapters": [
  "string"
  ]
 }
 ]
},
"storage": {
 "genericPhysicalAdapters": [
 {
  "id": "string",
  "type" :"string",
  "physicalLocation" : "string",
  "numOfReads": "number",
  "numOfWrites": "number",
  "readBytes": "number",
  "writeBytes": "number"
 }
 ],
 "genericVirtualAdapters": [
 {
  "id": "string",
  "type" :"string",
  "physicalLocation" : "string",
  "numOfReads": "number",
  "numOfWrites": "number",
  "readBytes": "number",
  "writeBytes": "number"
 }
 ],
 "fiberChannelAdapters": [
 {
  "id": "string",
  "wwpn": "string",
  "physicalLocation" : "string",
  "numOfReads": "number",
  "numOfWrites": "number",
  "readBytes": "number",
  "writeBytes": "number",
  "runningSpeed": "number",
  "ports": [
  {
   "id": "string",
   "wwpn": "string",
   "numOfReads": "number",
   "numOfWrites": "number",
   "readBytes": "number",
   "writeBytes": "number",
   "runningSpeed": "number",
   "physicalLocation" : "string"
  }
  ]
 }
 ],
 "sharedStoragePools": [
 {
  "id": "string",
  "poolDisks": [
  "string"
  ],
  "numOfReads": "number",
  "numOfWrites": "number",
  "totalSpace": "number",
  "usedSpace": "number",
```

```
      "readBytes": "number",
      "writeBytes": "number"
     }
     ],
    }
   }
   ]
  }
 }
}
```

STM Power Hypervisor JSON Specification

Power Hypervisor JSON Specification for Short Term Monitoring (STM) metrics

PCM STM URI provides the following Power Hypervisor (PYP) utilization data when **ShortTermMonitorEnabled** configuration value for a managed system is enabled.

```
{
 "systemUtil": {
  "utilInfo": {
   "version": "string",
   "metricType": "string",
   "monitoringType": "string",
   "mtms" : "string",
   "name" : "string"
  },
  "utilSample":
  {
   "timeStamp": "string",
   "status": "number",
   "errorInfo" : [
   { "errId" : "string", "errMsg" : "string"  }
   ],
   "timeBasedCycles" : "number",
   "systemFirmware" :
   {
    "utilizedProcCycles" : "number",
    "assignedMem" : "number"
   },
   "processor" :
   {
    "totalProcUnits" : "number",
    "availableProcUnits": "number",
    "configurableProcUnits" : "number",
    "procCyclesPerSecond" : "number"
   },
   "memory" :
   {
    "totalMem" : "number",
    "availableMem" : "number",
    "configurableMem" : "number"

   },
   "sharedMemoryPool" :
   [
   {
    "id" : "number",
    "name" : "string",
    "totalMem" : "number",
    "assignedMemToLpars" : "number",
    "assignedMemToSysFirmware" : "number",
    "totalIOMem" : "number",
    "mappedIOMemToLpars" : "number",
    "pageFaults" : "number",
    "pageDelays" : "number",
    "dedupedMemInPool"  : "number",
```

```
 "utilizedProcCyclesForDedup" : "number"
}
],
"sharedProcessorPool" :
[
{
 "id" : "number",
 "name" : "string",
 "assignedProcCycles" : "number",
 "utilizedProcCycles": "number",
 "maxProcUnits" : "number",
 "borrowedProcUnits": "number"
}
],
"network" :
{
 "sriovAdapters" :
 [
 {
  "drcIndex" : "string",
  "physicalPorts" :
  [
  {
   "id" : "number",
   "physicalLocation" : "string",
   "receivedPackets" : "number",
   "sentPackets" :  "number",
   "droppedSentPackets" : "number",
   "droppedReceivedPackets" : "number",
   "sentBytes" : "number",
   "receivedBytes" : "number",
   "errorIn" : "number",
   "errorOut" : "number"
  }
  ]
 }
 ],
 "HEAdapters" :
 [
 {
  "drcIndex"  : "string",
  "physicalPorts" :
  [
  {
   "id": "number",
   "physicalLocation" : "string",
   "receivedPackets" : "number",
   "sentPackets" :  "number",
   "droppedPackets" : "number",
   "sentBytes" : "number",
   "receivedBytes" : "number"
  }
  ]
 }
 ]
} ,
"lparsUtil" :
[
{
 "id" : "number",
 "uuid" : "string",
 "type" : "string",
 "name" : "string",
 "state" : "string",
 "affinityScore" : "number",
 "memory":
 {
```

```
   "poolId" : "number",
   "weight" : "number",
   "logicalMem" : "number",
   "backedPhysicalMem" : "number",
   "totalIOMem" : "number",
   "mappedIOMem" : "number",
   "dedupedMem" : "number"
  },
  "processor":
  {
   "poolId" :  "number",
   "mode"   :  "string",
   "maxVirtualProcessors" : "number",
   "maxProcUnits" : "number",
   "weight" : "number",
   "entitledProcCycles": "number",
   "utilizedCappedProcCycles" :"number",
   "utilizedUnCappedProcCycles" : "number",
   "idleProcCycles" : "number",
   "donatedProcCycles" : "number",
   "runLatchInstructions" : "number",
   "runLatchProcCycles" : "number",
   "timeSpentWaitingForProcessor" : "number",
   "numOfTimesWaitedForProcessor" : "number",
   "timeSpentWaitingForDispatch" :"number",
   "numOfTimesDispatched" :"number",
   "totalInstructions" : "number",
  "totalInstructionsExecutionTime" : "number
},

 "network" :
{
 "virtualEthernetAdapters" :
 [
  {

 "vlanId": "number",
 "vswitchId" :  "number",
 "physicalLocation" : "string",
   "isPortVLANID" : "boolean",
 "receivedPackets" : "number",
 "sentPackets" :  "number",
 "droppedPackets" : "number",
 "sentBytes" : "number",
 "receivedBytes" : "number",
 "receivedPhysicalPackets" : "number",
 "sentPhysicalPackets" :  "number",
 "droppedPhysicalPackets" : "number",
 "sentPhysicalBytes" : "number",
 "receivedPhysicalBytes" : "number"
  }
 ],
 "sriovLogicalPorts" :
 [
 {
 "drcIndex"  : "string",
 "physicalDrcIndex" : "string",
 "physicalPortId" :  "number",
   "physicalLocation" : "string",
 "receivedPackets" : "number",
 "sentPackets" :  "number",
 "droppedSentPackets" : "number",
 "droppedReceivedPackets" : "number",
 "sentBytes" : "number",
 "receivedBytes" : "number",
   "errorIn" : "number",
   "errorOut" "number"
```

```
   }
   ]
   },
    "Storage":
    {
     "virtualFiberChannelAdapters":
     [
     {
      "viosId"  : number,
      "wwpnPair":["wwpn1" ,"wwpn2" ],
    "physicalLocation" : "string"
     }
     ] ,
     "genericVirtualAdapters":
     [
     {
    "viosId" :  "number",
     "physicalLocation" : "string",
      "viosAdapterSlotId"  : "number"
     }
     ]
    }
   }
],
"viosUtil":
[
 {
 "id" :"number",
  "uuid" : "string",
  "name" : "string",
  "state" : "string",
  "affinityScore" : "number",
  "processor":
  {
   "poolId" :  "number",
   "mode"   : "string",
   "maxVirtualProcessors" : "number",
   "maxProcUnits" : "number",
   "weight" : "number",
   "entitledProcCycles" : "number",
   "utilizedCappedProcCycles" : "number",
   "utilizedUnCappedProcCycles" : "number",
   "idleProcCycles" : "number",
   "donatedProcCycles" : "number",
   "runLatchInstructions : "number",
   "runLatchProcCycles : "number",
   "timeSpentWaitingForProcessor" : "number",
   "numOfTimesWaitedForProcessor" : "number",
   "timeSpentWaitingForDispatch" :"number",
   "numOfTimesDispatched" :"number",
   "totalInstructions" : "number",
   "totalInstructionsExecutionTime" : "number
   },
   "memory" :
   {
    "assignedMem" : "number
   }
   "network" :
   {
    "virtualEthernetAdapters":
    [
    {
     "vlanId" :  "number",
     "vswitchId" :  "number",
     "physicalLocation" : "string",
     "isPortVLANID" : "boolean",
     "receivedPackets" : "number",
```

```
      "sentPackets" :  "number",
      "droppedPackets" : "number",
      "sentBytes" : "number",
      "receivedBytes" : "number",
      "receivedPhysicalPackets" : "number",
      "sentPhysicalPackets" :  "number",
      "droppedPhysicalPackets" : "number",
      "sentPhysicalBytes" : "number",
      "receivedPhysicalBytes" : "number"
     }
     ],
     "sriovLogicalPorts" :
     [
     {
      "drcIndex"  : "string",
      "physicalDrcIndex" : "string",
      "physicalPortId" :  "number",
      "physicalLocation" : "string",
      "receivedPackets" : "number",
      "sentPackets" :  "number",
      "droppedSentPackets" : "number",
      "droppedReceivedPackets" : "number",
      "sentBytes" : "number",
      "receivedBytes" : "number",
      "errorIn" : "number",
      "errorOut" "number"
     }
     ]
    }
   }
   ]
  }
 }
}
```

STM Virtual I/O Server JSON Specification

Virtual I/O Server JSON Specification for Short Term Monitoring metrics

PCM STM URI provides the following Virtual I/O Server (VIOS) utilization data when ShortTermMonitorEnabled configuration value for a managed system is enabled.

```
{
 "systemUtil": {
  "utilInfo": {
   "version": "string",
   "metricType": "string",
   "monitoringType": "string",
   "mtms": "string",
  },
  "utilSample": {
   "timeStamp": "string",
   "status": "number",
   "errorInfo": [
   {
    "errId": "string",
    "errMsg": "string"
   },
   ],
   "viosUtil": [
   {
    "id": "string",
    "name" : "string",
    "processor": {
     "userCounter": "number",
     "kernelCounter": "number",
     "purrCounter": "number",
```

```
 "spurrCounter": "number",
 "timeBaseCounter": "number"
},
"memory": {
 "utilizedMem": "number",
 "usedForNetworkBuffer": "number",
 "usedForOtherOperations": "number",
 "swapSpaceUsed": "number"
},
"network": {
 "genericAdapters": [
 {
  "id": "string",
  "type": "string",
  "physicalLocation" : "string",
  "receivedPackets": "number",
  "sentPackets": "number",
  "droppedPackets": "number",
  "sentBytes": "number",
  "receivedBytes": "number"
 }
 ],
 "sharedAdapters": [
 {
  "id": "string",
  "type": "string",
  "physicalLocation" : "string",
  "receivedPackets": "number",
  "sentPackets": "number",
  "droppedPackets": "number",
  "sentBytes": "number",
  "receivedBytes": "number",
  "bridgedAdapters": [
  "string"
  ]
 }
 ]
},
"storage": {
 "genericPhysicalAdapters": [
 {
  "id": "string",
  "type" :"string",
  "physicalLocation": "string",
  "numOfReads": "number",
  "numOfWrites": "number",
  "readBytes": "number",
  "writeBytes": "number",
 }
 ],
 "genericVirtualAdapters": [
 {
  "id": "string",
  "type" :"string",
  "physicalLocation" : "string",
  "numOfReads": "number",
  "numOfWrites": "number",
  "readBytes": "number",
  "writeBytes": "number"
 }
 ],
 "fiberChannelAdapters": [
 {
  "id": "string",
  "wwpn": "string",
  "physicalLocation": "string",
  "numOfReads": "number",
```

```
 "numOfWrites": "number",
 "readBytes": "number",
 "writeBytes": "number",
 "runningSpeed": "number",
 "ports": [
 {
  "id": "string",
  "wwpn": "string",
  "physicalLocation": "string",
  "numOfReads": "number",
  "numOfWrites": "number",
  "readBytes": "number",
  "writeBytes": "number",
  "runningSpeed": "number"
 }
 ]
}
],
"sharedStoragePools": [
{
 "id": "string",
 "poolDisks": [
 "string"
 ],
 "poolVirtualDevices": [
 "string"
 ],
 "numOfReads": "number",
 "numOfWrites": "number",
 "totalSpace": "number",
 "usedSpace": "number",
 "readBytes": "number",
 "writeBytes": "number"
}
],
"physicalDevices": [
{
 "id": "string",
 "uid": "string",
 "diskAdapterIds": "string",
 "poolId": "string",
 "numOfReads": "number",
 "numOfWrites": "number",
 "readBytes": "number",
 "writeBytes": "number",
 "readServiceTime": "number",
 "writeServiceTime": "number",
 "timeSpentInWaitQueue": "number",
 "waitQueueSize": "number",
 "numOfTimesServiceQueueIsFull": "number",

}
],
"virtualDevices": [
{
 "id": "string",
 "uid": "string",
 "poolId": "string",
 "totalSpace": "number",
 "usedSpace": "number",
 "numOfReads": "number",
 "numOfWrites": "number",
 "readBytes": "number",
 "writeBytes": "number",
 "readServiceTime": "number",
 "writeServiceTime": "number",
 "timeSpentInWaitQueue": "number",
```

```
      "waitQueueSize": "number",
      "numOfTimesServiceQueueIsFull": "number",
     },
     ],
    }
   }
   ]
  }
 }
}
```

Managed System Processed and Aggregated Metrics JSON Specification

The JSON specification for the managed system processed and aggregated metrics. All the parameters for the processed and aggregated metrics are same except that the Processed metrics provides only Average metrics. It does not provide minimum and maximum records.

PCM Processed or aggregated metrics URI provides the following managed system utilization data when the **AggregationEnabled** configuration value for a managed system is enabled.

```
{ "systemUtil": {
  "utilInfo": {
   "version": "string",
   "metricType": "string",
   "frequency": "number",
   "startTimeStamp": "string",
   "endTimeStamp": "string",
   "mtms" : "string",
   "name" : "string",
   "uuid" : "string",
   "metricArrayOrder" : ["Avg","Min","Max" ]
  },
  "utilSamples": [
  {
   "sampleInfo"
   {
    "timeStamp": "string",
    "numOfSamplesAggregated": "number",
    "status": "number",
    "errorInfo" : [
    { "errId" : "string",
     "errMsg" : "string",
     "uuid" : "string",
     "reportedBy" : "string",
     "occurenceCount" : "number"
    }
    ],
   },
   "systemFirmwareUtil" :
   {
    "utilizedProcUnits" : [
    "number",
    "number",
    "number"],
    "assignedMem" : [
    "number",
    "number",
    "number"]
   },
   "serverUtil" :
   {
    "processor" :
    {
     "totalProcUnits" : [
     "number",
```

```
 "number",
 "number"],
 "utilizedProcUnits" : [
 "number",
 "number",
 "number"],
 "availableProcUnits": [
 "number",
 "number",
 "number"],
 "configurableProcUnits" : [
 "number",
 "number",
 "number"]
},
"memory" :
{
 "totalMem" : [
 "number",
 "number",
 "number"],
 "availableMem" : [
 "number",
 "number",
 "number"],
 "configurableMem" : [
 "number",
 "number",
 "number"],
 "assignedMemToLpars" : [
 "number",
 "number",
 "number"]
},
"sharedMemoryPool" :
[
{
 "id" : "number",
 "totalMem" : [
 "number",
 "number",
 "number"],
 "assignedMemToLpars" : [
 "number",
 "number",
 "number"],
 "totalIOMem" : [
 "number",
 "number",
 "number"],
 "mappedIOMemToLpars" : [
 "number",
 "number",
 "number"],
 "assignedMemToSysFirmware" : [
 "number",
 "number",
 "number"]
}
],
"sharedProcessorPool" :
[
{
 "id" : "number",
 "name" : "string",
 "assignedProcUnits" : [
 "number",
```

```
 "number",
 "number"],
 "utilizedProcUnits": [
 "number",
 "number",
 "number"],
 "availableProcUnits": [
 "number",
 "number",
 "number"],
 "configuredProcUnits" : [
 "number",
 "number",
 "number"],
 "borrowedProcUnits": [
 "number",
 "number",
 "number"]
}
],
"network" :
{
 "sriovAdapters" :
 [
 {
  "drcIndex"  : "string",
  "physicalPorts" :  [
  {
   "id": "number",
   "physicalLocation": "string",
   "receivedPackets" :[
   "number",
   "number",
   "number"],
   "sentPackets" :  [
   "number",
   "number",
   "number"],
   "droppedPackets" : [
   "number",
   "number",
   "number"],
   "sentBytes" : [
   "number",
   "number",
   "number"],
   "receivedBytes" : [
   "number",
   "number",
   "number"],
   "errorIn" : [
   "number",
   "number",
   "number"],
   "errorOut" : [
   "number",
   "number",
   "number"]
  }
  ]
 }
 ],
 "HEAdapters" :
 [
 {
  "drcIndex"  : "string",
  "physicalPorts" : [
```

```
    {
     "id": "number",
     "physicalLocation": "string",
     "receivedPackets" :[
     "number",
     "number",
     "number"],
     "sentPackets" :  [
     "number",
     "number",
     "number"],
     "droppedPackets" : [
     "number",
     "number",
     "number"],
     "sentBytes" : [
     "number",
     "number",
     "number"],
     "receivedBytes" : [
     "number",
     "number",
     "number" ]
    }
    ]
   }
   ]
  }
},
"viosUtil":
[
{
 "id" :"number",
 "uuid" : "string",
 "name" : "string",
 "state" : "string",
 "affinityScore" :  "number",
 "memory" :
 {
  "assignedMem" : [
  "number",
  "number",
  "number"],
  "utilizedMem" : [
  "number",
  "number",
  "number"]
 },
 "processor":
 {
  "poolId" :  "number",
  "weight" : "number",
  "mode"   : "string",
  "maxVirtualProcessors" : [
  "number",
  "number",
  "number"],
  "maxProcUnits" : [
  "number",
  "number",
  "number"],
  "entitledProcUnits" : [
  "number",
  "number",
  "number"],
  "utilizedProcUnits": [
  "number",
```

```
 "number",
 "number"],
 "utilizedCappedProcUnits" : [
 "number",
 "number",
 "number"],
 "utilizedUncappedProcUnits" : [
 "number",
 "number",
 "number"],
 "idleProcUnits" : [
 "number",
 "number",
 "number"],
 "donatedProcUnits" : [
 "number",
 "number",
 "number"],
 "timeSpentWaitingForDispatch" : [
 "number",
 "number",
 "number"],
 "timePerInstructionExecution" : [
 "number",
 "number",
 "number"]
},
"network" :
{
 "clientLpars" : "string[]",
 "genericAdapters" :
 [
 {
  "id": "string",
  "type": "string",
  "physicalLocation": "string",
  "receivedPackets" : [
  "number",
  "number",
  "number"],
  "sentPackets" : [
  "number",
  "number",
  "number"],
  "droppedPackets" : [
  "number",
  "number",
  "number"],
  "sentBytes" : [
  "number",
  "number",
  "number"],
  "receivedBytes" : [
  "number",
  "number",
  "number" ]
 }
 ],
 "sharedAdapters":
 [
 {
  "id": "string",
  "type": "string",
  "physicalLocation": "string",
  "receivedPackets" : [
  "number",
  "number",
```

```
 "number"],
 "sentPackets" : [
 "number",
 "number",
 "number"],
 "droppedPackets" : [
 "number",
 "number",
 "number"],
 "sentBytes" : [
 "number",
 "number",
 "number"],
 "receivedBytes" : [
 "number",
 "number",
 "number" ],
 "bridgedAdapters": [
 "string"
 ]
}
],
"virtualEthernetAdapters":
[
{
 "physicalLocation": "string",
 "vlanId" :  "number",
 "vswitchId" :  "number",
 "isPortVLANID" : "boolean",
 "receivedPackets" : [
 "number",
 "number",
 "number"],
 "sentPackets" : [
 "number",
 "number",
 "number"],
 "droppedPackets" : [
 "number",
 "number",
 "number"],
 "sentBytes" : [
 "number",
 "number",
 "number"],
 "receivedBytes" : [
 "number",
 "number",
 "number" ],
 "receivedPhysicalPackets" : [
 "number",
 "number",
 "number" ],
 "sentPhysicalPackets" : [
 "number",
 "number",
 "number"],
 "droppedPhysicalPackets" : [
 "number",
 "number",
 "number"],
 "sentPhysicalBytes" : [
 "number",
 "number",
 "number"],
 "receivedPhysicalBytes" : [
 "number",
```

```
 "number",
 "number"]
}
],
"sriovLogicalPorts" :
[ {
 "drcIndex" : "string",
 "physicalLocation": "string",
 "physicalDrcIndex" : "string",
 "physicalPortId" :  "number",
 "receivedPackets" : [
 "number",
 "number",
 "number"],
 "sentPackets" : [
 "number",
 "number",
 "number"],
 "droppedPackets" : [
 "number",
 "number",
 "number"],
 "sentBytes" : [
 "number",
 "number",
 "number"],
 "receivedBytes" : [
 "number",
 "number",
 "number" ],
 "errorIn" : [
 "number",
 "number",
 "number"],
 "errorOut" : [
 "number",
 "number",
 "number" ]
}
]
},
"storage" :
{
 "clientLpars" : "string[]",
 "genericVirtualAdapters":
 [
 {
  "id": "string",
  "type": "string",
  "physicalLocation" : "string",
  "numOfReads": [
  "number",
  "number",
  "number"],
  "numOfWrites": [
  "number",
  "number",
  "number"],
  "readBytes": [
  "number",
  "number",
  "number"],
  "writeBytes": [
  "number",
  "number",
  "number"]
 }
```

```
],
"genericPhysicalAdapters" : [
{
 "id": "string",
 "type": "string",
 "physicalLocation": "string",
 "numOfReads": [
 "number",
 "number",
 "number"],
 "numOfWrites": [
 "number",
 "number",
 "number"],
 "readBytes": [
 "number",
 "number",
 "number"],
 "writeBytes": [
 "number",
 "number",
 "number"]
}
],
"fiberChannelAdapters":
[
{
 "id": "string",
 "wwpn": "string",
 "physicalLocation": "string",
 "numOfPorts": "number",
 "numOfReads": [
 "number",
 "number",
 "number"],
 "numOfWrites": [
 "number",
 "number",
 "number"],
 "readBytes": [
 "number",
 "number",
 "number"],
 "writeBytes": [
 "number",
 "number",
 "number"],
 "runningSpeed": [
 "number",
 "number",
 "number"]
}
],
"sharedStoragePools":
[
{
 "id": "string",
 "totalSpace": [
 "number",
 "number",
 "number"],
 "usedSpace": [
 "number",
 "number",
 "number"],
 "numOfReads": [
 "number",
```

```
      "number",
      "number"],
      "numOfWrites": [
      "number",
      "number",
      "number"],
      "readBytes": [
      "number",
      "number",
      "number"],
      "writeBytes": [
      "number",
      "number",
      "number"]
     }
     ]
    }
   }
   ]
  }
  ]
 }
}
```

Logical Paritition Processed/Aggregated Metrics JSON Specification

The JSON specification for the logical partition Processed and Aggregated metrics. All the parameters for the Processed and Aggregated metrics are same except that the Processed metrics provides only Average metrics. It does not provide Minimum and Maximum records.

PCM Processed or Aggregated metrics URI provides the following logical partition utilization data when **AggregationEnabled** configuration value for a managed system is enabled.

```
{
 "systemUtil": {
  "utilInfo": {
   "version": "string",
   "metricType": "string",
   "frequency": "number",
   "startTimeStamp": "string",
   "endTimeStamp": "string",
   "mtms" : "string",
   "name" : "string",
   "metricArrayOrder" : ["Avg", "Min", "Max" ]
  },
  "utilSamples": [
  {
   "sampleInfo"
   {
    "timeStamp": "string",
    "numOfSamplesAggregated": "number",
    "status": "number",
    "errorInfo" : [
    { "errId" : "string",
     "errMsg" : "string",
     "uuid" : "string",
     "reportedBy" : "string",
     "occurenceCount" : "number"
    }
    ],
   },
   "lparsUtil" :
   [
   {
    "id" : "number",
    "uuid" : "string",
    "name" : "string",
```

```
"state" : "string",
"type" : "string",
"affinityScore" :  "number",
"memory":
{
 "poolId" : "number",
 "weight" : "number",
 "logicalMem" : [
 "number",
 "number",
 "number"],
 "backedPhysicalMem" : [
 "number",
 "number",
 "number"],
 "totalIOMem" : [
 "number",
 "number",
 "number"],
 "mappedIOMem" : [
 "number",
 "number",
 "number"]
},
"processor":
{
 "poolId" : "number",
 "weight" : "number",
 "mode"   : "string",
 "maxVirtualProcessors": [
 "number",
 "number",
 "number"],
 "maxProcUnits" : [
 "number",
 "number",
 "number"],
 "entitledProcUnits" : [
 "number",
 "number",
 "number"],
 "utilizedProcUnits":[
 "number",
 "number",
 "number"],
 "utilizedCappedProcUnits" : [
 "number",
 "number",
 "number"],
 "utilizedUncappedProcUnits" : [
 "number",
 "number",
 "number"],
 "idleProcUnits" :  [
 "number",
 "number",
 "number"],
 "donatedProcUnits" : [
 "number",
 "number",
 "number"],
 "timeSpentWaitingForDispatch" : [
 "number",
 "number",
 "number"],
 "timePerInstructionExecution" : [
 "number",
```

```
 "number",
 "number"]
},
"network" :
{
 "virtualEthernetAdapters":
 [
 {
  "physicalLocation": "string",
  "vlanId"    :  "number",
  "vswitchId" :  "number",
  "isPortVLANID"  : "boolean",
  "viosId"  :    "number" ,
  "sharedEthernetAdapterId" : "string",
  "receivedPackets" : [
  "number",
  "number",
  "number"],
  "sentPackets" : [
  "number",
  "number",
  "number"],
  "droppedPackets" : [
  "number",
  "number",
  "number"],
  "sentBytes" : [
  "number",
  "number",
  "number"],
  "receivedBytes" : [
  "number",
  "number",
  "number" ],
  "receivedPhysicalPackets" : [
  "number",
  "number",
  "number"],
  "sentPhysicalPackets" :  [
  "number",
  "number",
  "number"],
  "droppedPhysicalPackets" : [
  "number",
  "number",
  "number"],
  "sentPhysicalBytes" : [
  "number",
  "number",
  "number"],
  "receivedPhysicalBytes" : [
  "number",
  "number",
  "number"]
 }
 ],
 "sriovLogicalPorts" :
 [
 {
  "drcIndex"  : "string",
  "physicalLocation" : "string",
  "physicalDrcIndex" : "string",
  "physicalPortId" :  "number",
  "receivedPackets" : [
  "number",
  "number",
  "number"],
```

```
 "sentPackets" : [
 "number",
 "number",
 "number"],
 "droppedPackets" : [
 "number",
 "number",
 "number"],
 "sentBytes" : [
 "number",
 "number",
 "number"],
 "receivedBytes" : [
 "number",
 "number",
 "number" ],
 "errorIn" : [
 "number",
 "number",
 "number" ],
 "errorOut" : [
 "number",
 "number",
 "number" ]
 }
 ]
},
"storage" :
{
 "genericVirtualAdapters":
 [
 {
  "id": "string",
  "type": "string",
  "viosId" :  "number",
  "physicalLocation" : "string",
  "numOfReads": [
  "number",
  "number",
  "number"],
  "numOfWrites": [
  "number",
  "number",
  "number"],
  "readBytes": [
  "number",
  "number",
  "number"],
  "writeBytes": [
  "number",
  "number",
  "number"]
 }
 ],
 "virtualFiberChannelAdapters":
 [
 {
  "id": "string",
  "wwpn": "string",
  "wwpn2": "string",
  "physicalLocation" : "string",
  "physicalPortWWPN" : "string",
  "viosId" : "number",
  "numOfReads": [
  "number",
  "number",
  "number"],
```

```
      "numOfWrites": [
      "number",
      "number",
      "number"],
      "readBytes": [
      "number",
      "number",
      "number"],
      "writeBytes": [
      "number",
      "number",
      "number"],
      "runningSpeed": [
      "number",
      "number",
      "number"]
     }
     ]
    }
   }
   ]
  }
  ]
 }
}
```

Notices

This information was developed for products and services offered in the US.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (your company name) (year).
Portions of this code are derived from IBM Corp. Sample Programs.
© Copyright IBM Corp. _enter the year or years_.

If you are viewing this information in softcopy, the photographs and color illustrations may not appear.

Privacy policy considerations

IBM Software products, including software as a service solutions, (“Software Offerings”) may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user, or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering’s use of cookies is set forth below.

Depending upon the configurations deployed, this Software Offering may use session cookies that collect each user’s user name and IP address for purposes of session management. These cookies can be disabled, but disabling them will also eliminate the functionality they enable.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM’s Privacy Policy at http://www.ibm.com/privacy and IBM’s Online Privacy Statement at http://www.ibm.com/privacy/details the section entitled “Cookies, Web Beacons and Other Technologies” and the “IBM Software Products and Software-as-a-Service Privacy Statement” at http://www.ibm.com/software/info/product-privacy.

Programming interface information

This Hardware Management Console Programmer's Guide publication documents intended Programming Interfaces that allow the customer to write programs to obtain the services of IBM Hardware Management Console Version 8 Release 8.4.0 Maintenance Level 0.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at Copyright and trademark information at www.ibm.com/legal/copytrade.shtml.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Terms and conditions

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability: These terms and conditions are in addition to any terms of use for the IBM website.

Personal Use: You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative works of these publications, or any portion thereof, without the express consent of IBM.

Commercial Use: You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights: Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

Printed in USA