Dell OpenManage Power Center v3.0 REST API Guide

This document describes the integration of REST API with OMPC to generate reports

Dell Engineering
October 2014
Revisions

<table>
<thead>
<tr>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>September 2014</td>
<td>Initial release</td>
</tr>
</tbody>
</table>

THIS WHITE PAPER IS FOR INFORMATIONAL PURPOSES ONLY, AND MAY CONTAIN TYPOGRAPHICAL ERRORS AND TECHNICAL INACCURACIES. THE CONTENT IS PROVIDED AS IS, WITHOUT EXPRESS OR IMPLIED WARRANTIES OF ANY KIND. © 2014 Dell Inc. All rights reserved. Reproduction of this material in any manner whatsoever without the express written permission of Dell Inc. is strictly forbidden. For more information, contact Dell.

PRODUCT WARRANTIES APPLICABLE TO THE DELL PRODUCTS DESCRIBED IN THIS DOCUMENT MAY BE FOUND AT: http://www.dell.com/learn/us/en/19/terms-of-sale-commercial-and-public-sector Performance of network reference architectures discussed in this document may vary with differing deployment conditions, network loads, and the like. Third-party products may be included in reference architectures for the convenience of the reader. Inclusion of such third-party products does not necessarily constitute Dell’s recommendation of those products. Please consult your Dell representative for additional information.

Trademarks used in this text:

Dell™, the Dell logo, Dell Boomi™, Dell Precision™, OptiPlex™, Latitude™, PowerEdge™, PowerVault™, PowerConnect™, OpenManage™, EqualLogic™, Compellent™, KACE™, FlexAddress™, Force10™ and Vostro™ are trademarks of Dell Inc. Other Dell trademarks may be used in this document. Cisco Nexus®, Cisco MDS®, Cisco NX-OS®, and other Cisco Catalyst® are registered trademarks of Cisco System Inc. EMC VNX® and EMC Unisphere® are registered trademarks of EMC Corporation. Intel®, Pentium®, Xeon®, Core® and Celeron® are registered trademarks of Intel Corporation in the U.S. and other countries. AMD® is a registered trademark and AMD Opteron™, AMD Phenom™ and AMD Sempron™ are trademarks of Advanced Micro Devices, Inc. Microsoft®, Windows®, Windows Server®, Internet Explorer®, MS-DOS®, Windows Vista® and Active Directory® are either trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries. Red Hat® and Red Hat® Enterprise Linux® are registered trademarks of Red Hat, Inc. in the United States and/or other countries. Novell® and SUSE® are registered trademarks of Novell Inc. in the United States and other countries. Oracle® is a registered trademark of Oracle Corporation and/or its affiliates. Citrix®, Xen®, XenServer® and XenMotion® are either registered trademarks or trademarks of Citrix Systems, Inc. in the United States and/or other countries. VMware®, Virtual SMP®, vMotion®, vCenter® and vSphere® are registered trademarks or trademarks of VMware, Inc. in the United States or other countries. IBM® is a registered trademark of International Business Machines Corporation. Broadcom® and NetXtreme® are registered trademarks of Broadcom Corporation. QLogic is a registered trademark of QLogic Corporation. Other trademarks and trade names may be used in this document to refer to either the entities claiming the marks and/or names or their products and are the property of their respective owners. Dell disclaims proprietary interest in the marks and names of others.
Table of contents

Revisions ............................................................................................................................................................................................. 2

1. Executive summary ................................................................................................................................................................... 4

2. Introduction ................................................................................................................................................................................ 4

3. Key Integration Concepts......................................................................................................................................................... 4
   3.1 Client Integration Overview .......................................................................................................................................... 4
   3.2 Base URI ............................................................................................................................................................................ 5
   3.3 Security ............................................................................................................................................................................. 6
   3.4 Authentication Mechanism ........................................................................................................................................... 6

4. Understanding the API flow ..................................................................................................................................................... 6

5. API Usage .................................................................................................................................................................................... 8
   5.1 Consume Report ............................................................................................................................................................. 8
      5.1.1 Attributes Explained ........................................................................................................................................................ 8
      5.1.2 Get System Report Type Summary .............................................................................................................................. 9
      5.1.3 Power Hoarders Report ........................................................................................................................................... 10
      5.1.4 Power Frugal Report ............................................................................................................................................... 11
      5.1.5 Power Data Report ...................................................................................................................................................... 12
      5.1.6 Power Headroom Report ............................................................................................................................................ 13
      5.1.7 General Inventory Report ............................................................................................................................................ 14
      5.1.8 Power Hoarders Rack Report ..................................................................................................................................... 15
      5.1.9 Power Frugal Rack Report ........................................................................................................................................... 16
      5.1.10 Raw Monitoring Data Report ...................................................................................................................................... 17
      5.1.11 Comparison Report ...................................................................................................................................................... 18
   5.2 Report Status .................................................................................................................................................................. 19
   5.3 Report Result ................................................................................................................................................................. 20

6. Additional resources (Optional) ............................................................................................................................................. 21

A Dell Technical White Paper
1. Executive summary

The end result of any monitoring tool is Report generation. A report can be generated on a daily, weekly, monthly, or yearly basis. It helps administrators to utilize the datacenter efficiently and eventually leads to huge Return on Investment (RoI).

The OpenManage Power Center (OPMC) Application Programming Interface (API) can help third party to generate report for devices which are the largest consumers of power, least amount of power, overall power usage and available unused power, racks that are the most utilized in power, racks that are the least utilized in power etc.

A Representational State Transfer (REST) API will be provided to other console applications which wish to access OMPC monitoring data via proper authentication bridge in between.

This white paper outlines the requirements for using the REST API feature, the integration call procedures, and all types of report supported by OpenManage Power Center.

2. Introduction

This document describes the REST way of integration with (OMPC) v3.0. It provides examples of creating new reports or gathering the report details through REST APIs. Using these APIs, you can create various types of OMPC reports, check reports execution status, retrieve report results, or remove target reports.

3. Key Integration Concepts

3.1 Client Integration Overview

The REST client makes standard HTTP(s) requests to the REST API end-point. Each request is sent using a HTTP verb (for example, PUT, GET, POST, DELETE, HEAD, and OPTIONS) and includes a message body in JSON or XML format.

The response uses a standard HTTP status code. The REST client can be developed in various programming languages (for example, C# and Java) or scripting languages (for example, Python and Perl) that have basic support for HTTPS communication and are capable of constructing and parsing XML or JSON payloads.

This document includes examples using JSON format.
3.2 Base URI

The Base URI for OMPC REST request is - \texttt{HTTPS://HOST_ADDRESS:PORT_NUM/powercenter/api/}

| HOST_ADDRESS | The network address of the machine on which the product is installed. It could be a DNS host name or IP address. |
| PORT_NUM     | The port number that is opened as web application server for OMPC. It should be the port assigned in OMPC installation. Default port is 8643 but use the one that you have configured during the OMPC installation. |

An example to fetch Power-Hoarder report details using REST API:

\texttt{https://localhost:8643/powercenter/api/powerHoarders}
3.3 Security

The REST services will only be exposed through HTTPS to ensure that the common threats associated with HTTPs traffic are mitigated.

3.4 Authentication Mechanism

OMPC uses Basic Authentication along with HTTPS to enable authentication of REST requests.

Each request requires an Authorization header in the request, in the format of base-64 encoded string of the combination of following credentials:

- username,
- password,
- accountType: Possible values for this field are:
  - OMPC_ACCOUNT = 0;
  - WIN_LOCAL_ACCOUNT = 1;
  - WIN_DOMAIN_ACCOUNT = 2;
  - LINUX_LOCAL_ACCOUNT = 3;
  - LDAP_ACCOUNT = 4;
- Domain: It is an optional field unless accountType field value is WIN_DOMAIN_ACCOUNT.

Note:

- If the credentials are missing or invalid, a 401 (Authorization failure) error is displayed.
- This mechanism is only supported when SSL/TLS is enabled for the transport.

4. Understanding the API flow
The figure below explains the API consumption flow

**STEP I:** Each Report resource has unique URI that needs to be invoked along with Authorization Header

[At backend report generation will be triggered]

API → https://<HOST_ADDRESS>:<PORT_NUM>/powercenter/api/powerHoarders

Note**: Authorization Header is mandatory to send
Method: POST

**STEP II:** Check Report Generation Status At Server

Check Running Status API → https://<HOST_ADDRESS>:<PORT_NUM>/powercenter/api/checkStatus?reportId=8

Note**: Authorization Header is mandatory to send
Method: GET

**STEP III:** After success of step-II, report can be accessed at


Note**: Authorization Header is mandatory to send
Method: GET

Start

End
5. API Usage

5.1 Consume Report

The following section has covered report request samples and attributes meaning.

5.1.1 Attributes Explained

The following attributes are used in the JSON request objects.

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Type</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>timeResolutionCode</td>
<td>Int</td>
<td>For &quot;Last&quot; X hours/days/weeks/months value should be 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>For &quot;Date range&quot; value should be 2</td>
</tr>
<tr>
<td>timeResolutionType</td>
<td>Int</td>
<td>For &quot;Hour(s)&quot; value should be 1, For &quot;Day(s)&quot; value should be 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>For &quot;Week(s)&quot; value should be 3, For &quot;Month(s)&quot; value should be 4</td>
</tr>
<tr>
<td>timeResolutionValue</td>
<td>Int</td>
<td>For Hour(s) Range should be in between 1~24 ; For Day(s) Range should be</td>
</tr>
<tr>
<td></td>
<td></td>
<td>in between 1~365</td>
</tr>
<tr>
<td></td>
<td></td>
<td>For Week(s) Range should be in between 1~52; For Months(s) Range should</td>
</tr>
<tr>
<td></td>
<td></td>
<td>be in between 1~12 Default will be 1</td>
</tr>
<tr>
<td>aggPeriodType</td>
<td>Int</td>
<td>For &quot;Hour(s)&quot; value should be 1, For &quot;Day(s)&quot; value should be 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>For &quot;Week(s)&quot; value should be 3, For &quot;Month(s)&quot; value should be 4</td>
</tr>
<tr>
<td>aggPeriodValue</td>
<td>Int</td>
<td>Integer value as per aggPeriodType</td>
</tr>
<tr>
<td>entityList</td>
<td>List</td>
<td>Multiple Entity (ex: Datacenter) can be selected at a time</td>
</tr>
<tr>
<td>entityGroupList</td>
<td>List</td>
<td>Multiple Entity Group can be selected at a time</td>
</tr>
<tr>
<td>mandatory</td>
<td>Int</td>
<td>The attributes which cannot be modified by the user at the creation as</td>
</tr>
<tr>
<td></td>
<td></td>
<td>well as edit time 1 represent checked and disabled and 0 represent none</td>
</tr>
<tr>
<td>required</td>
<td>Int</td>
<td>User can select or unselect while create and edit (As it's a check box</td>
</tr>
<tr>
<td></td>
<td></td>
<td>we need to pass value 0 or 1) 1 represent Checked and disabled and 0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>represent Unchecked (Optional field- Check box state)</td>
</tr>
<tr>
<td>needed</td>
<td>Int</td>
<td>The attributes which needs to be hidden at the time of report creation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>as well as edit. (Attributes not relevant to a particular report). 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>represent to show field and 0 represent hide (Hidden field)</td>
</tr>
</tbody>
</table>
5.1.2 Get System Report Type Summary

The GetSystemReportTypeSummary option returns the predefined report template for a specific report type.

**API usage:** POST `<BASE_URL>/getSystemReportTypeSummary?reportType=<ReportType>`

**Parameters:**

<table>
<thead>
<tr>
<th>#</th>
<th>Name</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>reportType</td>
<td>Report type string, one of:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>powerHoarders</td>
</tr>
<tr>
<td></td>
<td></td>
<td>powerFrugal</td>
</tr>
<tr>
<td></td>
<td></td>
<td>powerData</td>
</tr>
<tr>
<td></td>
<td></td>
<td>powerHeadroom</td>
</tr>
<tr>
<td></td>
<td></td>
<td>generalInventory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>powerHoardersRack</td>
</tr>
<tr>
<td></td>
<td></td>
<td>powerFrugalRack</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rawMonitoringData</td>
</tr>
<tr>
<td></td>
<td></td>
<td>compareReport</td>
</tr>
</tbody>
</table>

**Sample Response:**

```json
{
    "criteriaObj": null,
    "errorObj": null,
    "requestObj": null,
    "responseCode": 0,
    "responseObj": {
        "aggPeriodType": 0,
        "aggPeriodValue": 0,
        "attributes": [
            {
                "fieldName": "Name",
                "key": "name",
                "mandatory": 1,
                "needed": 1,
                "required": 1
            },
            {
                "fieldName": "Hostname",
                "key": "hostname",
                "mandatory": 1,
                "needed": 1,
                "required": 1
            },
            {
                "fieldName": "Device Type",
                "key": "deviceType",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Model",
                "key": "model",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Service Tag",
                "key": "serviceTag",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Location",
                "key": "location",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Average Power",
                "key": "avgPower",
                "mandatory": 1,
                "needed": 1,
                "required": 1
            },
            {
                "fieldName": "Minimum Power",
                "key": "minPower",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Maximum Power",
                "key": "maxPower",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Power Capabilities",
                "key": "powerCapabilities",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Stranded Power",
                "key": "strandedPower",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Allocated Power",
                "key": "allocatedPower",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Consumed Power",
                "key": "consumedPower",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Average Temperature",
                "key": "avgTemp",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Minimum Temperature",
                "key": "minTemp",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Maximum Temperature",
                "key": "maxTemp",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "EPR Status",
                "key": "eprState",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Connection State",
                "key": "connectionState",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Health State",
                "key": "healthState",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Device Ip",
                "key": "deviceIp",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Time Stamp",
                "key": "timeStamp",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Peak Power",
                "key": "peakPower",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "IT Energy",
                "key": "energyIt",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "IT Energy Cost",
                "key": "energyItCost",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Cooling Energy",
                "key": "energyCooling",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Cooling Energy Cost",
                "key": "energyCoolingCost",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Total Energy",
                "key": "energyTotal",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Total Energy Cost",
                "key": "energyTotalCost",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            },
            {
                "fieldName": "Instantaneous Power",
                "key": "instantPower",
                "mandatory": 0,
                "needed": 0,
                "required": 0
            }
        ],
        "description": "Devices which are the largest consumers of power (highest average)",
        "endDateTime": null,
        "entityGroupList": [],
        "entityList": [],
        "maxResults": 10,
        "startDateTime": null
    }
}
```
5.1.3 Power Hoarders Report

The API below is used for consuming "powerHoarders" report.

API usage:

POST <BASE_URL>/powerHoarders

Sample Request:

```
{
    "requestObj": {
        "description": "Devices which are the largest consumers of power (highest average)",
        "startDateTime": "2014-07-13T18:30:00.000Z",
        "endDateTime": "2014-07-23T18:30:00.000Z",
        "entityList": [
            {"serviceTagOrSerialNum": ",","ipAddress": "192.168.0.1","hostName": ""},
            {"serviceTagOrSerialNum": ",","ipAddress": "192.168.0.2","hostName": ""},
            {"serviceTagOrSerialNum": ",","ipAddress": "192.168.0.3","hostName": ""}
        ],
        "entityGroupList": [],
        "aggPeriodType": -1,
        "aggPeriodValue": -1,
        "maxResults": -1,
        "attributes": [
            {"fieldName": "Name","key": "name","mandatory": 1,"needed": 1,"required": 1},
            {"fieldName": "Hostname","key": "hostname","mandatory": 1,"needed": 1,"required": 1},
            {"fieldName": "Device Type","key": "deviceType","mandatory": 1,"needed": 1,"required": 1},
            {"fieldName": "Model","key": "model","mandatory": 1,"needed": 1,"required": 1},
            {"fieldName": "Service Tag","key": "serviceTag","mandatory": 1,"needed": 1,"required": 1},
            {"fieldName": "Location","key": "location","mandatory": 1,"needed": 1,"required": 1},
            {"fieldName": "Average Power","key": "avgPower","mandatory": 1,"needed": 1,"required": 1}
        ]
    }
}
```

Sample Response:

```
{
    "criteriaObj": null,
    "errorObj": null,
    "requestObj": null,
    "responseCode": 0,
    "responseObject": {"link": "https://localhost:8080/powercenter/api/checkStatus?reportId=8"}
}
```
5.1.4 Power Frugal Report

The API below is used for consuming "powerFrugal" report.

API usage:

POST <BASE_URL>/powerFrugal

Sample Request:

```json
{
  "requestObj": {
    "description": "Devices which use the least amount of power (lowest average)",
    "startDateTime": "2014-07-13T18:30:00.000Z",
    "endDateTime": "2014-07-23T18:30:00.000Z",
    "entityList": [
      {'serviceTagOrSerialNum': '', 'ipAddress': '192.168.2.11', 'hostName': ''},
      {'serviceTagOrSerialNum': '', 'ipAddress': '192.168.2.16', 'hostName': ''},
      {'serviceTagOrSerialNum': '', 'ipAddress': '192.168.2.19', 'hostName': ''}],
    "aggPeriodType": -1,
    "aggPeriodValue": -1,
    "maxResults": -1,
    "attributes": [
      {'fieldName': 'Name', 'key': 'name', 'mandatory': 1, 'needed': 1, 'required': 1},
      {'fieldName': 'Hostname', 'key': 'hostname', 'mandatory': 1, 'needed': 1, 'required': 1},
      {'fieldName': 'Device Type', 'key': 'deviceType', 'mandatory': 1, 'needed': 1, 'required': 1},
      {'fieldName': 'Model', 'key': 'model', 'mandatory': 1, 'needed': 1, 'required': 1},
      {'fieldName': 'Service Tag', 'key': 'serviceTag', 'mandatory': 1, 'needed': 1, 'required': 1},
      {'fieldName': 'Location', 'key': 'location', 'mandatory': 1, 'needed': 1, 'required': 1},
      {'fieldName': 'Average Power', 'key': 'avgPower', 'mandatory': 1, 'needed': 1, 'required': 1}
    ]
  }
}
```

Sample Response Body:

```json
{
  "criteriaObj": null,
  "errorObj": null,
  "requestObj": null,
  "responseCode": 0,
  "responseObject": {
    "link": "https://localhost:8080/powercenter/api/checkStatus?reportId=82"
  }
}
```
5.1.5 Power Data Report

The API below is used for consuming "powerData" report.

**API usage:**

POST `<BASE_URL>/powerData`

**Sample Request:**

```json
{
   "requestObj": {
      "description": "Device or group min, max, average and instantaneous power numbers",
      "startDateTime": "2014-07-13T18:30:00.000Z",
      "endDateTime": "2014-07-23T18:30:00.000Z",
      "entityList": [
         {
            "serviceTagOrSerialNum": "",
            "ipAddress": "192.168.0.1",
            "hostName": ""
         },
         {
            "serviceTagOrSerialNum": "",
            "ipAddress": "192.168.0.2",
            "hostName": ""
         },
         {
            "serviceTagOrSerialNum": "",
            "ipAddress": "192.168.0.3",
            "hostName": ""
         }
      ],
      "entityGroupList": [],
      "aggPeriodType": -1,
      "aggPeriodValue": -1,
      "maxResults": -1,
      "attributes": [
         {
            "fieldName": "Name",
            "key": "name",
            "mandatory": 1,
            "needed": 1,
            "required": 1
         },
         {
            "fieldName": "Hostname",
            "key": "hostname",
            "mandatory": 1,
            "needed": 1,
            "required": 1
         },
         {
            "fieldName": "Device Type",
            "key": "deviceType",
            "mandatory": 1,
            "needed": 1,
            "required": 1
         },
         {
            "fieldName": "Model",
            "key": "model",
            "mandatory": 1,
            "needed": 1,
            "required": 1
         },
         {
            "fieldName": "Service Tag",
            "key": "serviceTag",
            "mandatory": 1,
            "needed": 1,
            "required": 1
         },
         {
            "fieldName": "Location",
            "key": "location",
            "mandatory": 1,
            "needed": 1,
            "required": 1
         },
         {
            "fieldName": "Average Power",
            "key": "avgPower",
            "mandatory": 1,
            "needed": 1,
            "required": 1
         },
         {
            "fieldName": "Minimum Power",
            "key": "minPower",
            "mandatory": 1,
            "needed": 1,
            "required": 1
         },
         {
            "fieldName": "Maximum Power",
            "key": "maxPower",
            "mandatory": 1,
            "needed": 1,
            "required": 1
         },
         {
            "fieldName": "Instantaneous Power",
            "key": "instantPower",
            "mandatory": 1,
            "needed": 1,
            "required": 1
         }
      ]
   }
}
```

**Sample Response Body:**

```json
{
   "criteriaObj": null,
   "errorObj": null,
   "requestObj": null,
   "responseCode": 0,
   "responseObj": {
      "link": "https://localhost:8080/powercenter/api/checkStatus?reportld=9"
   }
}
```
5.1.6 Power Headroom Report

The API below is used for consuming "powerHeadroom" report.

API usage:

POST <BASE_URL>/powerHeadroom

Sample Request body:

```json
{
    "requestObj": {
        "description": "Devices which are the largest consumers of power(highest average)",
        "startTime": "2014-07-13T18:30:00.000Z",
        "endTime": "2014-07-23T18:30:00.000Z",
        "entityList": [],
        "entityGroupList": [{
            "entityId": 54,
            "name": "XXX_001",
            "type": "DATACENTER",
            "deviceServiceTag": "-1",
            "groupPath": "XXX/RT_000_v1"
        },
        {
            "entityId": 55,
            "name": "YYY_002",
            "type": "DATACENTER",
            "deviceServiceTag": "-1",
            "groupPath": "YYY/RT_V_002"
        }],
        "aggPeriodType": -1,
        "aggPeriodValue": -1,
        "maxResults": -1,
        "attributes": [
            {"fieldName": "Name","key": "name","mandatory": 1,"needed": 1,"required": 1},
            {"fieldName": "Average Power","key": "avgPower","mandatory": 1,"needed": 1,"required": 1},
            {"fieldName": "Location","key": "location","mandatory": 1,"needed": 1,"required": 1},
            {"fieldName": "Power Capabilities","key": "powerCapabilities","mandatory": 0,"needed": 0,"required": 0},
            {"fieldName": "Stranded Power","key": "strandedPower","mandatory": 1,"needed": 1,"required": 1}]
    }
}
```

Sample Response Body:

```json
{
    "criteriaObj": null,
    "errorObj": null,
    "requestObj": null,
    "responseCode": 0,
    "responseObj": {
        "link": "https://localhost:8080/powercenter/api/checkStatus?reportId=55"
    }
}
```
5.1.7 General Inventory Report

The API below is used for consuming "generalInventory" report.

API usage:

POST <BASE_URL>/generalInventory

Sample Request body:

```json
{
    'requestObj': {
        'description': 'Devices which are the largest consumers of power(highest average)',
        'startDateTime': '2014-07-13T18:30:00.000Z',
        'endDateTime': '2014-07-23T18:30:00.000Z',
        'entityList': [
            {
                'serviceTagOrSerialNum': '',
                'ipAddress': '192.168.0.1',
                'hostName': ''
            },
            {
                'serviceTagOrSerialNum': '',
                'ipAddress': '192.168.0.2',
                'hostName': ''
            },
            {
                'serviceTagOrSerialNum': '',
                'ipAddress': '192.168.0.3',
                'hostName': ''
            }
        ],
        'entityGroupList': [],
        'aggPeriodType': -1,
        'aggPeriodValue': -1,
        'maxResults': -1,
        'attributes': [
            {'fieldName': 'Name', 'key': 'name', 'mandatory': 1, 'needed': 1, 'required': 1},
            {'fieldName': 'Hostname', 'key': 'hostname', 'mandatory': 1, 'needed': 1, 'required': 1},
            {'fieldName': 'Device Type', 'key': 'deviceType', 'mandatory': 1, 'needed': 1, 'required': 1},
            {'fieldName': 'Model', 'key': 'model', 'mandatory': 1, 'needed': 1, 'required': 1},
            {'fieldName': 'Service Tag', 'key': 'serviceTag', 'mandatory': 1, 'needed': 1, 'required': 1},
            {'fieldName': 'Location', 'key': 'location', 'mandatory': 1, 'needed': 1, 'required': 1},
            {'fieldName': 'Average Power', 'key': 'avgPower', 'mandatory': 1, 'needed': 1, 'required': 1},
            {'fieldName': 'Instantaneous Power', 'key': 'instantPower', 'mandatory': 1, 'needed': 1, 'required': 1},
            {'fieldName': 'Stranded Power', 'key': 'strandedPower', 'mandatory': 1, 'needed': 1, 'required': 1},
            {'fieldName': 'Average Temperature', 'key': 'avgTemp', 'mandatory': 1, 'needed': 1, 'required': 1}
        ]
    },
    'entityGroupList': [],
    'aggPeriodType': -1,
    'aggPeriodValue': -1,
    'maxResults': -1
}
```

Sample Response Body:

```json
{
    'criteriaObj': null,
    'errorObj': null,
    'requestObj': null,
    'responseCode': 0,
    'responseObj': {'link': 'https://localhost:8080/powercenter/api/checkStatus?reportId=67'}
}
```
5.1.8 Power Hoarders Rack Report

The API below is used for consuming "powerHoardersRack" report.

API usage:

POST <BASE_URL>/powerHoardersRack

Sample Request body:

```json
{
  "requestObj": {
    "description": "Racks that are the most utilized in power",
    "startDateDateTime": "2014-07-13T18:30:00.000Z",
    "endDateDateTime": "2014-07-23T18:30:00.000Z",
    "entityList": [
      {
        "serviceTagOrSerialNum": "",
        "ipAddress": "192.168.0.1",
        "hostName": ""
      },
      {
        "serviceTagOrSerialNum": "",
        "ipAddress": "192.168.0.2",
        "hostName": ""
      },
      {
        "serviceTagOrSerialNum": "",
        "ipAddress": "192.168.0.3",
        "hostName": ""
      }
    ],
    "entityGroupList": [],
    "aggPeriodType": -1,
    "aggPeriodValue": -1,
    "maxResults": -1,
    "attributes": [
      {
        "fieldName": "Name",
        "key": "name",
        "mandatory": 1,
        "needed": 1,
        "required": 1
      },
      {
        "fieldName": "Location",
        "key": "location",
        "mandatory": 1,
        "needed": 1,
        "required": 1
      },
      {
        "fieldName": "Average Power",
        "key": "avgPower",
        "mandatory": 1,
        "needed": 1,
        "required": 1
      },
      {
        "fieldName": "Stranded Power",
        "key": "strandedPower",
        "mandatory": 1,
        "needed": 1,
        "required": 1
      }
    ]
  }
}
```

Sample Response Body:

```json
{
  "criteriaObj": null,
  "errorObj": null,
  "requestObj": null,
  "responseCode": 0,
  "responseObj": {
    "link": "https://localhost:8080/powercenter/api/checkStatus?reportId=34"
  }
}
```
5.1.9 Power Frugal Rack Report

The API below is used for consuming "powerFrugalRack" report.

**API usage:**

POST `<BASE_URL>/powerFrugalRack`

**Sample Request body:**

```json
{
    'requestObj': {
        'description': 'Devices which are the largest consumers of power(highest average),
        'startDateTime': "2014-07-13T18:30:00.000Z",
        'endDateTime': "2014-07-23T18:30:00.000Z",
        'entityList': [
            {'serviceTagOrSerialNum': '', 'ipAddress': '192.168.0.1', 'hostName': ''},
            {'serviceTagOrSerialNum': '', 'ipAddress': '192.168.0.2', 'hostName': ''},
            {'serviceTagOrSerialNum': '', 'ipAddress': '192.168.0.3', 'hostName': ''}],
        'entityGroupList': [],
        'aggPeriodType': -1,
        'aggPeriodValue': -1,
        'maxResults': -1,
        'attributes': [
            {'fieldName': 'Name', 'key': 'name', 'mandatory': 1, 'needed': 1, 'required': 1},
            {'fieldName': 'Location', 'key': 'location', 'mandatory': 1, 'needed': 1, 'required': 1},
            {'fieldName': 'Average Power', 'key': 'avgPower', 'mandatory': 1, 'needed': 1, 'required': 1},
            {'fieldName': 'Stranded Power', 'key': 'strandedPower', 'mandatory': 1, 'needed': 1, 'required': 1}
        ]
    }
}
```

**Sample Response Body:**

```json
{
    'criteriaObj': null,
    'errorObj': null,
    'requestObj': null,
    'responseCode': 0,
    'responseObj': {'link': 'https://localhost:8080/powercenter/api/checkStatus?reportId=22'}
}
```
5.1.10 Raw Monitoring Data Report

The API below is used for consuming "rawMonitoringData" report.

API usage:

POST <BASE_URL>/rawMonitoringData

Sample Request body:

```json
{
  "requestObj": {
    "description": "Raw Monitoring Data Sample",
    "startDateTime": "2014-07-13T18:30:00.000Z",
    "endDateTime": "2014-07-23T18:30:00.000Z",
    "entityList": [
      {
        "serviceTagOrSerialNum": "",
        "ipAddress": "192.168.0.1",
        "hostName": ""
      },
      {
        "serviceTagOrSerialNum": "",
        "ipAddress": "192.168.0.2",
        "hostName": ""
      },
      {
        "serviceTagOrSerialNum": "",
        "ipAddress": "192.168.0.3",
        "hostName": ""
      }
    ],
    "entityGroupList": [],
    "aggPeriodType": -1,
    "aggPeriodValue": -1,
    "maxResults": -1,
    "attributes": [
      {
        "fieldName": "Name",
        "key": "name",
        "mandatory": 1,
        "needed": 1,
        "required": 1
      },
      {
        "fieldName": "Location",
        "key": "location",
        "mandatory": 1,
        "needed": 1,
        "required": 1
      },
      {
        "fieldName": "Average Power",
        "key": "avgPower",
        "mandatory": 1,
        "needed": 1,
        "required": 1
      },
      {
        "fieldName": "Minimum Power",
        "key": "minPower",
        "mandatory": 1,
        "needed": 1,
        "required": 1
      },
      {
        "fieldName": "Maximum Power",
        "key": "maxPower",
        "mandatory": 1,
        "needed": 1,
        "required": 1
      },
      {
        "fieldName": "Instantaneous Power",
        "key": "instantPower",
        "mandatory": 1,
        "needed": 1,
        "required": 1
      },
      {
        "fieldName": "Average Temperature",
        "key": "avgTemp",
        "mandatory": 1,
        "needed": 1,
        "required": 1
      },
      {
        "fieldName": "Minimum Temperature",
        "key": "minTemp",
        "mandatory": 1,
        "needed": 1,
        "required": 1
      },
      {
        "fieldName": "Maximum Temperature",
        "key": "maxTemp",
        "mandatory": 1,
        "needed": 1,
        "required": 1
      }
    ]
  }
}
```

Sample Response Body:

```json
{
  "criteriaObj": null,
  "errorObj": null,
  "requestObj": null,
  "responseCode": 0,
  "responseObj": {
    "link": "https://localhost:8080/powercenter/api/checkStatus?reportId=77"
  }
}
```
5.1.11 Comparison Report

The API below is used for consuming "comparison" report.

API usage:

POST <BASE_URL>/compareReport

Sample Request body:

```json
{
  "requestObj": {
  "description": "Compare Report",
  "startDateTime": "2014-07-13T18:30:00.000Z",
  "endDateTime": "2014-07-23T18:30:00.000Z",
  "entityList": [
    {
      "serviceTagOrSerialNum": ",";
      "ipAddress": "192.168.0.10.1",
      "hostName": ""
    },
    {
      "serviceTagOrSerialNum": ",";
      "ipAddress": "192.168.0.2",
      "hostName": ""
    },
    {
      "serviceTagOrSerialNum": ",";
      "ipAddress": "192.168.0.3",
      "hostName": ""
    }
  ],
  "entityGroupList": [],
  "aggPeriodType": -1,
  "aggPeriodValue": -1,
  "maxResults": -1,
  "attributes": [
    {"fieldName": "Name","key": "Name","mandatory": 1,"needed": 1,"required": 1},
    {"fieldName": "Average Power","key": "avgPower","mandatory": 1,"needed": 1,"required": 1},
    {"fieldName": "Minimum Power","key": "minPower","mandatory": 1,"needed": 1,"required": 1},
    {"fieldName": "Maximum Power","key": "maxPower","mandatory": 1,"needed": 1,"required": 1},
    {"fieldName": "Instantaneous Power","key": "instantPower","mandatory": 1,"needed": 1,"required": 1},
    {"fieldName": "Average Temperature","key": "avgTemp","mandatory": 1,"needed": 1,"required": 1},
    {"fieldName": "Minimum Temperature","key": "minTemp","mandatory": 1,"needed": 1,"required": 1},
    {"fieldName": "Maximum Temperature","key": "maxTemp","mandatory": 1,"needed": 1,"required": 1},
    {"fieldName": "Peak Power","key": "peakPower","mandatory": 1,"needed": 1,"required": 1},
    {"fieldName": "IT Energy","key": "energyIT","mandatory": 1,"needed": 1,"required": 1},
    {"fieldName": "IT Energy Cost","key": "energyITCost","mandatory": 1,"needed": 1,"required": 1},
    {"fieldName": "Cooling Energy","key": "energyCooling","mandatory": 1,"needed": 1,"required": 1},
    {"fieldName": "Cooling Energy Cost","key": "energyCoolingCost","mandatory": 1,"needed": 1,"required": 1},
    {"fieldName": "Total Energy","key": "energyTotal","mandatory": 1,"needed": 1,"required": 1},
    {"fieldName": "Total Energy Cost","key": "energyTotalCost","mandatory": 1,"needed": 1,"required": 1}
  ]
}
```

Sample Response Body:

```json
{
  "criteriaObj": null,
  "errorObj": null,
  "requestObj": null,
  "responseCode": 0,
  "responseObj": {"link": "https://localhost:8080/powercenter/api/checkStatus?reportId=17"}
}
```
5.2 Report Status

The CheckStatus option returns the status of specific report. The possible report statuses are:

- Running
- Deleted
- Pending
- Erroneous
- Completed

When the status for a report is “Completed”, the result is displayed.

**API usage:** POST <BASE_URL>/chekStatus?reportId=<Id>

**Parameters:**

<table>
<thead>
<tr>
<th>#</th>
<th>Name</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>reportId</td>
<td>Report Id returned by previous add report call</td>
</tr>
</tbody>
</table>

**Sample Request:** NA

**Sample Response:**

```json
{
  "criteriaObj": null,
  "errorObj": null,
  "requestObj": null,
  "responseCode": 0,
  "responseObj": {
    "message": "Completed",
    "status": true
  }
}
```
5.3 Report Result

GetReportResult is used for returning the results of specific report instance from last running. To avoid large amount of data, we are adding two new parameters for pagination (related update highlighted in yellow). A typical combination is currentPage=0 and rowCountPerPage=10 calling this API for the first time.

**API usage:** POST <BASE_URL>/getReportResult?reportId=<Id>&currentPage=<page>&rowCountPerPage=<count>

**Parameters:**

<table>
<thead>
<tr>
<th>#</th>
<th>Name</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>reportId</td>
<td>Report Id returned by previous add report call</td>
</tr>
<tr>
<td>2</td>
<td>currentPage</td>
<td>Current page to fetch, should be &gt;= 0 and not greater than total page count</td>
</tr>
<tr>
<td>3</td>
<td>rowCountPerPage</td>
<td>Row count for each page, should be &gt; 0 and &lt;= 1000</td>
</tr>
</tbody>
</table>

**Request sample:**

**Sample Request:** NA

**Sample Response:**

```json
{
    "criteriaObj": {
        "filterObj": [],
        "paginationObj": {
            "currentPage": 1,"rowCountPerPage": 10,"totalItemsCount": 50
        },
        "sortObj": []
    },
    "errorObj": null,"responseObj": {"reportResultGeneratedDate": "2014-09-24T10:11:43.739Z","reportStartedDate": "2014-09-24T10:11:34.264Z","rows": [{
        "row": [{"key": "name","needed": 1,"value": "Server - 192.168.0.1"}
        {"key": "location","needed": 1,"value": "/ABC/RT_V_002/"}
        {"key": "hostname","needed": 1,"value": ""}
        {"key": "serviceTag","needed": 1,"value": "002.019"}
        {"key": "avgPower","needed": 1,"value": "51"}
        {"key": "deviceType","needed": 1,"value": "Server"}
        {"key": "model","needed": 1,"value": "XXXXXXX"}],
        "row": [{"key": "name","needed": 1,"value": "Server - 192.168.0.1"}
        {"key": "location","needed": 1,"value": "/YYY/RT_V_002/"}
        {"key": "hostname","needed": 1,"value": ""}
        {"key": "serviceTag","needed": 1,"value": "002.002"}
        {"key": "avgPower","needed": 1,"value": "50"}
        {"key": "deviceType","needed": 1,"value": "Server"}
        {"key": "model","needed": 1,"value": "XXXXXXXX"}],
    }]
}
```
6. Additional resources (Optional)

You can visit dell.com/support which is focused on meeting your needs with proven services and support.

You can also visit DellTechCenter.com, an IT Community where you can connect with Dell Customers and Dell employees for sharing knowledge, best practices, and information about Dell products and installations.

For more information, visit www.dell.com/powercenter.