Oracle® Hyperion Profitability and Cost Management

REST API Reference

Release 11.2.16
F91487-01
January 2024

Oracle Hyperion Profitability and Cost Management REST API Reference, Release 11.2.16

F91487-01

Primary Author: EPM Information Development Team

Contents

4 APIs for Detailed Profitability

A References and Sample Client

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Documentation Feedback

To provide feedback on this documentation, click the feedback button at the bottom of the page in any Oracle Help Center topic. You can also send email to epmdoc_ww@oracle.com.

1

About the Profitability and Cost Management REST APIs

You can use the Oracle Hyperion Profitability and Cost Management REST Web Services APIs to automate a variety of Profitability and Cost Management tasks. This API provides an alternative to using the web-based user interface for many operations.

You can use one of a variety of methods to access the Profitability and Cost Management REST API. For example, you can access the REST API through web browsers and other client applications such as cURL and GNU Wget.

Alternatively, you can use the these APIs in REST client applications that are developed in languages such as JavaScript, Ruby, Perl, Java, JavaFX.

REST API Response

Every API returns some or all of the following content.

Table 1-1 REST API response

Attribute	Description	Example
Type	Product name	Profitability
Items (Optional)	List of objects returned by REST API	List of applications, list of stages, and so on. This information is displayed only when the list of objects is returned.
Status	Status code returned by the REST API	Status values -1 - In Progress 0 - Success 1 - Error
Status Message	Status message equivalent to Status	
Details	String value returned by REST API	CES Taskflow ID/Success Message in case of Success status Error Message in case Error/Exception
Links	Object representing URI of the invoked REST API	Self Link: http://myServer.mydomain.com:19000/profitability/rest/v1/applications Related Link job Status: http://myServer.mydomain.com:19000/profitability/rest/v1/applications/jobs/ChecktaskStatusJob/LM1T2_LM1T2_DeployCube_D20160113T065447_b64_1

2

APIs Common for All Application Types

Related Topics

- Get Applications
- Get Applications Type Details
- Get Applications by Type
- Get Job Services by an Application
- Delete Application
- Get Task Status by Process Name
- Get Job Services for POV
- Get POVs by an Application
- Delete POV
- Get Task Details by Process Name
- Get Essbase Applications
- Get Essbase Application Properties
- Get Essbase Application Dimensions
- Update Dimension for Native Application
- Enable Native Application
- Create Application Using Native Method

Get Applications

Use this API to list all existing Oracle Hyperion Profitability and Cost Management applications.

Table 2-1 Get Applications

Name	Description
Action Type	GET
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications
Request Payload	

Table 2-1 (Cont.) Get Applications

Name	Description

Response	{
	"items": [{
	"name": "AA1",
	"type": "MANAGEMENT_LEDGER",
	"enabled": true
	}, {
	"name": "vLM5",
	"type": "MANAGEMENT_LEDGER",
	"enabled": true
	}, {
	"name": "vLMA",
	"type": "MANAGEMENT_LEDGER",
	"enabled": false
	}],
	"type": "Profitability",
	"links": [{
	"href":
	"http://myServer.mydomain.com:19000/
	profitability/rest/v1/applications",
	"action": "GET",
	"rel": "self"
	}],
	"status": 0,
	"details": "",
	"statusMessage": "Success"
	}

Process Type	Synchronous

Table 2-2 Response Values

Parameter	Description	Example
Name	Application Name	
Type	Application Type	MANAGEMENT_LEDGER GENERAL

Request Parameters: none

Get Applications Type Details

Use this API to list the application type for all existing Profitability and Cost Management applications.

Table 2-3 Get Applications Type Details

Name	Description
Action Type	GET
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/{applicationName}
Request Payload	

Response	<pre>{
	"type": "Profitability",
	"links": [{
	"href": "http://://
	myServer.mydomain.com:19000/profitability/
	rest/v1/applications/BksMl12",
	"action": "GET",
	"rel": "self"
	}],
	"status": 0,
	"details": "MANAGEMENT_LEDGER",
	"statusMessage": "Success"
	}</pre>

Process Type	Synchronous

Table 2-4 Request Parameters

Parameter	Description	Example
applicationName	Application Name	

Table 2-5 Response Values

Parameter	Description	Example
Type	Application Type	MANAGEMENT_LEDGER GENERAL DETAIL

Get Applications by Type

Use this API to list all Profitability and Cost Management applications of the selected type.

Table 2-6 Get Applications by Type

Name	Description
Action Type	GET
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/applicationsByType?queryParameter={"applicationType":"MANAGEMENT_LEDGER"}

Table 2-6 (Cont.) Get Applications by Type

Name	Description
Request Payload	

Response	{
	"items": [{
	"name": "AA1",
	"type": "MANAGEMENT_LEDGER"
	}, {
	"name": "vLM5",
	"type": "MANAGEMENT_LEDGER"
	}, {
	"name": "vLMA",
	"type": "MANAGEMENT_LEDGER"
	}],
	"type": "Profitability",
	"links": [{
	"href": "http://
	myServer.mydomain.com:19000/
	profitability/rest/v1/applications/
	applicationsByType?
	queryParameter=%7B%22applicationType%22:%
	22MANAGEMENT_LEDGER%22%7D",
	"action": "GET",
	"rel": "self",
	"data": {
	"applicationType":
	"MANAGEMENT_LEDGER"
	}
	}],
	"status": 0,
	"details": "",
	"statusMessage": "Success"
	}

Process Type	Synchronous

Table 2-7 Response Parameters

Parameter	Description	Example
queryParameter	Application Type	{"applicationType":"MANAGEMENT_LEDGER"} {"applicationType":"GENERAL"} {"applicationType":"DETAIL"}

Table 2-8 Response Values

Parameter	Description	Example
Name	Application Name	

Table 2-8 (Cont.) Response Values

Parameter	Description	Example
Type	Application Type	MANAGEMENT_LEDGER GENERAL DETAIL

Get Job Services by an Application

Use this API to list all Profitability and Cost Management application jobs of the selected application.

Table 2-9 Get Job Services by an Application

Name	Description
Action Type	GET
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/{applicationName}/jobs
Request Payload	

Response	{
	"type": "Profitability",
	"links": [{
	"href": "http://myServer.mydomain.com:19000/profitability/rest/v1/applications/BksML12/jobs",
	"action": "GET",
	"rel": "self"
	}],
	"status": 0,
	"details": "1. Deploy Cube \nURI: http://{HostName}/profitability/rest/11.1.2.4.000/applications/{applicationName}/jobs/ledgerDeployCubeJob?isKeepData=Value&isReplaceCube=Value&isRunNow=Value&comment=Value\n2. Task Status \nURI: http://{HostName}/profitability/rest/11.1.2.4.000/applications/jobs/ChecktaskStatusJob/{processName}",
	"statusMessage": "Success"
	}

Process Type	Synchronous

Table 2-10 Request Parameters

Parameter	Description	Example
applicationName	Application Name	

Response Values: None

Table 2-11 Response Values

Parameter	Description	Example
Type	Job Type	Deploy Cube

Delete Application

Use this operation to delete an existing Profitability and Cost Management application, and its association with Oracle Hyperion Shared Services (all application types).

Table 2-12 Delete Application

Name	Description
Action Type	DELETE
Request URI	http://slc06vya.mydomain.com:6756/profitability/rest/v1/applications/{applicationName}
Request Payload	

Response	<pre>{
	"type": "Profitability",
	"links": [{
	"href": "http://
	myServer.mydomain.com:6756/
	profitability/rest/v1/applications/
	BksMl12",
	"action": "DELETE",
	"rel": "self"
	}],
	"status": 0,
	"details": "Application: BksMl12
	is deleted successfully.",
	"statusMessage": "Success"
	}</pre>

Process Type	Synchronous

Table 2-13 Request Parameters

Parameter	Description	Example
applicationName	Application Name	

Response Values: None

Get Task Status by Process Name

Use this operation to view the current status of the job process name (CES taskflow) as it is displayed on the Taskflow Status Summary (all application types).

Table 2-14 Get Task Status by Process Name

Name	Description
Action Type	GET
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/jobs/ChecktaskStatusJob/{processName}
Request Payload	

Response	{
	"type": "Profitability",
	"links": [{
	"href": "http://myServer.mydomain.com:19000/profitability/rest/v1/applications/jobs/ChecktaskStatusJob/RBkML1_ExportTemplate_D20160112T025419_836",
	"action": "GET",
	"rel": "self"
	}],
	"status": 0,
	"details": "ExportTemplate=Success,RBkML1_ExportTemplate_D20160112T025419_836=Done",
	"statusMessage": "Success"
	}

Process Type	Synchronous

Table 2-15 Request Parameters

Parameter	Description	Example
processName	Process Name/TaskflowId	RBkML1_ExportTemplate_D20160112T025419_836

Table 2-16 Response Values

Parameter	Description	Example
Task	Task Name	ExportTemplate
Status	Task Status	Success

Get Job Services for POV

Use this operation to retrieve all POV Jobs details for a selected application (all application types).

Table 2-17 Get Job Services for POV

Name	Description
Action Type	GET
Request URI	http://slc06vya.mydomain.com:19000/profitability/rest/v1/applications/{applicationName}/povs/jobs
Request Payload	

Table 2-17 (Cont.) Get Job Services for POV

Name	Description

Response	{
	"type": "Profitability",
	"links": [{
	"href": "http://
	slc06vya.mydomain.com:19000/
	profitability/rest/v1/applications/
	Ex3F3/povs/jobs",
	"action": "GET",
	"rel": "self"
	}],
	"status": 0,
	"details": "1. Copy POV \nURI:
	http://{HostName}/profitability/rest/
	11.1.2.4.000/applications/
	{applicationName}/povs/
	{povMemberGroup}/jobs/copyPOVJob/
	{destPOVMemberGroup}?
	isManageRule=true&isInputData=true&isA
	djustmentValues=false&isAllocatedValue
	s=true\n2. Copy POV \nURI: http://
	{HostName}/profitability/rest/
	11.1.2.4.000/applications/
	{applicationName}/povs/
	{povMemberGroup}/jobs/clearPOVJob?
	isManageRule=true&isInputData=true&isA
	djustmentValues=false&isAllocatedValue
	s=true\n3. Delete POV \nURI: http://
	{HostName}/profitability/rest/
	11.1.2.4.000/applications/
	{applicationName}/povs/
	{povMemberGroup}/jobs/{povMemberGroup}
	\n4. Run ML Calculation \nURI: http://
	{HostName}/profitability/rest/
	11.1.2.4.000/applications/
	{applicationName}/povs/
	{povMemberGroup}/jobs/
	runLedgerCalculationJob?
	isClearCalculated=true&isExecuteCalcul
	ations=true&isRunNow=true&comment=valu
	e&subsetStart=value&subsetEnd=value&ru
	leName=value&ruleSetName=value&exeType
	=value\n",
	"statusMessage": "Success"
	}

Process Type	Synchronous

Table 2-18 Request Parameters

Parameter	Description	Example
applicationName	Application Name	Ex3F3

Table 2-19 Response Values

Parameter	Description	Example
Type	Job Type	Copy POV Run ML Calculation

Get POVs by an Application

Use this operation to retrieve all POV details for a selected application (all application types).

Table 2-20 Get POVs by an Application

Name	Description
Action Type	GET
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/{applicationName}/povs
Request Payload	

Table 2-20 (Cont.) Get POVs by an Application

Name	Description

Response	{
	"items": [{
	"povDimensionMember1": "FY14",
	"povDimensionMember2": "JUN",
	"povDimensionMember3":
	"Actual",
	"povDimensionMember4":
	"Working",
	"povState": "Draft"
	}, {
	"povDimensionMember1": "FY14",
	"povDimensionMember2": "JUL",
	"povDimensionMember3":
	"Actual",
	"povDimensionMember4":
	"Working",
	"povState": "Draft"
	}],
	"type": "Profitability",
	"links": [{
	"href": "http://
	slc06vya.mydomain.com:19000/
	profitability/rest/v1/applications/
	Ex3F3/povs",
	"action": "GET",
	"rel": "self"
	}],
	"status": 0,
	"details": "",
	"statusMessage": "Success"
	}

Process Type	Synchronous

Table 2-21 Request Parameters

Parameter	Description	Example
applicationName	Application Name	

Table 2-22 Response Values

Parameter	Description	Example
Pov Member Group	Pov Dimension Member1	2023
	Pov Dimension Member2	January
	Pov Dimension Member3	Actual
	Pov Dimension Member4	Plan

Delete POV

Use this operation to delete an existing POV in a Profitability and Cost Management application (all application types).

Table 2-23 Delete POV

Name	Description
Action Type	DELETE
Request URI	http://myServer.mydomain.com:6756/profitability/rest/v1/applications/{applicationName}/povs/{povMemberGroup}
Request Payload	

Response	{
	"type": "Profitability",
	"links": [{
	"href": "http://slc06vya.mydomain.com:6756/profitability/rest/v1/applications/Ex3F3/povs/FY14_JUN_Actual_Working",
	"action": "DELETE",
	"rel": "self",
	"data": {}
	}, {
	"href": "http://SLC06VYA.mydomain.com:19000/profitability/rest/v1/applications/jobs/ChecktaskStatusJob/Ex3F3_Ex3F3_DeleteMLPOV_D20160116T140536_da2_1",
	"action": "GET",
	"rel": "Job status"
	}],
	"status": -1,
	"details": "Ex3F3_Ex3F3_DeleteMLPOV_D20160116T140536_da2_1",
	"statusMessage": "In Progress"
	}

Process Type	Asynchronous

Table 2-24 Request Parameters

Parameter	Description	Example
applicationName	Application Name	
povMemberGroup	povDimensionMember1	2023.
	povDimensionMember2	January
	povDimensionMember3	Actual
	povDimensionMember4	Plan

Table 2-25 Response Values

Parameter	Description	Example
TaskflowID	Process Id for the submitted task	Ex3F3_Ex3F3_DeleteMLPOV_D20160113T052141_2fd_1

Get Task Details by Process Name

Use this operation to retrieve the details like job id, name, start time, end time, user id and etc. for the given process.

Table 2-26 Get Task Details by Process Name

Name	Description
Action Type	GET
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/jobs/taskDetails/{processName}
Request Payload	
Response	

```
{
    "type": "Profitability",
    "links": [{
        "href": "http://
slc06vya.mydomain.com:19000/
profitability/rest/v1/applications/jobs/
ChecktaskStatusJob/
Ex3F3_Ex3F3_DeleteMLPOV_D20160116T140536_
da2_1",
        "action": "GET",
        "rel": "self"
    }],
    "status": 0,
    "details": "",
    "statusMessage": "Success"
}
```

Process Type	Synchronous

Table 2-27 Request Parameters

Parameter	Description	Example
processName	Taskflow Id of the job	RBkML1_ExportTemplate_D20160112T025419_836

Table 2-28 Response Values

Parameter	Description	Example
applicationName	Name of the Application for the given process name	
comment	Comment given for the job/process	
jobType	Type of the job submitted.	
startTime	Start time of the process	
executionType	Execution type of the process	
jobProperties	Parameters given for the job execution	
Status	Status of the submitted process.SUCCESS/ FAILURE	
taskflowID	Name of the process	
endTime	End time of the process	
User	Id of the user who submitted the process	
povName	Specify dimension member names of the POV for which this task details operation has been applied: povDimensionMember1 povDimensionMember2 povDimensionMember3 povDimensionMember4	povDimensionMember1 = 2012 povDimensionMember2 = January povDimensionMember3 = Actual

Get Essbase Applications

Use this operation to Get Oracle Essbase applications for a given Essbase Application Server.

Table 2-29 Get Essbase Applications

Name	Description
Action Type	GET

Table 2-29 (Cont.) Get Essbase Applications

Name	Description
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/aamApplications/{essbaseServer}
Request Payload	

Response	{
	"type": "Profitability",
	"links": [{
	"href": "http://slc06vya.mydomain.com:19000/profitability/rest/v1/aamApplications/EssbaseCluster-1",
	"action": "GET",
	"rel": "self"
	}],
	"status": 0,
	"details": "76m11R:[77m11R] \nAA3C:[AA3C] \nAmtpDim:[AmtpDim] \nAnMstr:[AnMstr] \nASOsamp:[Sample] \nBksMast:[BksMast] \nBksML12C:[BksML12C] \nBksML13C:[BksML13C] \nBksML1C:[BksML1C] \nBSMML1C:[BSMML1C] \nCMAPMast:[CMAPMast] \nCMAPR:[CMAPR] \nCust:[Cust] \nD2S1:[D2S1] \nDemo1C:[Demo1C] \nDemoC:[DemoC] \nDPOHF1C:[DPOHF1C] \nEAlocBug:[EAlocBug] \nEx11F2C:[Ex11F2C] \nEx33FlC:[Ex33FlC] \nEX3F1LOC:[EX3F1LOC] \nEX3F1SDC:[EX3F1SDC] \nEx3FL3:[Ex3FL3] \nEx3FlC:[Ex3FlC] \nEx5F1C:[Ex5F1C] \nEx5M:[Ex5M] \nEXAMP:[EXAMP] \nExDPOH:[ExDPOH] \nExDPOHL:[ExDPOHL] \nExe5:[Exe5] \nExel11:[Exel11] \nExel11C:[Exel11C] \nExel3C:[Exel3C]\n",
	"statusMessage": "Success"
	}

Process Type	Synchronous

Table 2-30 Request Parameters

Parameter	Description	Example
essbaseServer	Name of Essbase Server	EssbaseCluster-1

Table 2-31 Response Values

Parameter	Description	Example
applicationName	Essbase Application Name	76m11R

Get Essbase Application Properties

Use this operation to get Oracle Essbase application properties.

Table 2-32 Get Essbase Application Properties

Name	Description
Action Type	GET
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/aamApplications
Request Payload	

Response	{
	"type": "Profitability",
	"links": [{
	"href": "http://slc06vya.mydomain.com:19000/profitability/rest/v1/aamApplications",
	"action": "GET",
	"rel": "self"
	}],
	"status": 0,
	"details": "essbaseAppServers: [EssbaseCluster-1] \nwebServer:SLC06VYA.mydomain.com \ninstanceNames:[PROFITABILITY_WEB_APP] \nsharedServicesProjects:[Essbase Studio Server 11.1.2 Servers, EssbaseCluster-1, Foundation, Default Application Group] \n",
	"statusMessage": "Success"
	}

Process Type	Synchronous

Request Parameters: None

Table 2-33 Response Values

Parameter	Description	Example
essbaseAppServers	Essbase Application Server	EssbaseCluster-1
webServer	HPCM Application server	SLC06VYA.mydomain.com
instanceNames	Instance Name	PROFITABILITY_WEB_APP

Table 2-33 (Cont.) Response Values

Parameter	Description	Example
sharedServicesProjects	Share Services Project	EssbaseCluster-1

Get Essbase Application Dimensions

Use this operation to get dimensions for a given application and a given cube name (all application types).

Table 2-34 Get Essbase Application Dimensions

Name	Description
Action Type	GET
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/aamApplications/{essbaseAppServer}/{applicationName}?queryParameter={"cubeName"."BksML12C")
Request Payload	

Response	{
	"type": "Profitability",
	"links": [{
	"href": "http://slc06vya.mydomain.com:19000/profitability/rest/v1/aamApplications/EssbaseCluster-1/BksML12C?queryParameter=%7B%22cubeName%22:%22BksML12C%22%7D%22",
	"action": "GET",
	"rel": "self",
	"data": {
	"cubeName": "BksML12C"
	}
	}],
	"status": 0,
	"details": "[Accounts, Activities, Balance, CostCenters, Customers, Drivers, Period, Products, Region, Rule, Scenario, Year]",
	"statusMessage": "Success"
	}

Process Type	Synchronous

Table 2-35 Request Parameters

Parameter	Description	Example
essbaseAppServer	Essbase Application Server	EssbaseCluster-1
applicationName	Essbase Application Name	BksML12C
queryParameter	Cube Name	{"cubeName":"BksML12C"}

Table 2-36 Response Values

Parameter	Description	Example
Dimension Names	Dimension Name	Accounts, Activities, Balance, CostCenters, Customers, Drivers, Period, Products, Region, Rule, Scenario, Year

Update Dimension for Native Application

Use this operation to add or update a dimension for any native (file-based) application using a comma-delimited text file (.csv or .txt).

Table 2-37 Update Dimension for Native Application

Name	Description
Action Type	POST
Request URI	http://{HostName}/profitability/rest/v1/fileApplications/{applicationName}/updateDimension

Request Payload	<pre>{
	"dataFileName":"input.txt"
	}</pre>

Table 2-37 (Cont.) Update Dimension for Native Application

Name	Description

Response	{
	"type": "Profitability",
	"links": [
	{
	"href": " http://
	myServer.mydomain.com:19000/profitability/
	rest/v1/fileApplications/BksML12/
	updateDimension ",
	"action": "POST",
	"rel": "self",
	"data": {
	"dataFileName":"input.txt"
	}
	}, {
	"href": "http://
	myServer.mydomain.com:19000/profitability/
	rest/v1/applications/jobs/ChecktaskStatusJob/
	BksML12_BksML12_UpdateDimension_D20160118T051020_
	bb8_1",
	"action": "GET",
	"rel": "Job status"
	}
	],
	"status": -1,
	"details": "BksML12_BksML12_ UpdateDimension
	_D20160118T051020_bb8_1",
	"statusMessage": "In_Progress"
	}

Process Type	Asynchronous

Table 2-38 Request Parameters

Parameter	Description	Example
applicationName	Application Name	BksML12
dataFileName	Name of the flat file already present in the import_export location of the server. (Mandatory parameter).	DimA.txt

Table 2-39 Response Values

Parameter	Description	Example
Details	CES Task Id	BksML12_BksML12_UpdateDimension_D20160118T051020_bb8_1

Enable Native Application

Use this resource to enable any native (file-based) application.

Table 2-40 Enable Native Application

Name	Description
Action Type	POST
Request URI	http://{HostName}/profitability/rest/v1/fileApplications/{applicationName}/enableApplication
Request Payload	

Response	{
	"type": "Profitability",
	"links": [
	{
	"href": "http://slc06vya.mydomain.com:19000/profitability/rest/v1/fileApplications/BksMl12/enableApplication",
	"action": "POST",
	"rel": "self"
	}
	],
	"status": 0,
	"details": "BksMl12_BksMl12_EnableApplication_D20160113T075011_53c_1",
	"statusMessage": "SUCCESS"
	}

Process Type	Asynchronous

Table 2-41 Request Parameters

Parameter	Description	Example
applicationName	Application Name	BksMl12

Table 2-42 Response Values

Parameter	Description	Example
Details	CES Task Id	BksMl12_BksMl12_EnableApplication_D20160113T075011_53c_1

Create Application Using Native Method

Use this operation to create a new application of any type using the "native" (file-based) method. Creating an application using this method will allow you to add and

update dimensions from .csv files, using the Update Dimension for Native Application operation.

Table 2-43 Create Application Using Native Method

Name	Description
Action Type	POST
Request URI	http://{HostName}:19000/profitability/rest/v1/fileApplications/{applicationName}

Request Payload	<pre>{
	"description": "description",
	"instanceName": "PROFITABILITY_WEB_APP",
	"essApplicationServer":"EssbaseCluster-1",
	"sharedServicesProject":"EssbaseCluster-1",
	"applicationType": "MANAGEMENT_LEDGER",
	"webServer":"slc06vya.mydomain.com",
	"ruleDimensionName":"Rule",
	"balanceDimensionName":"Balance",
	"unicode": "true"
	}</pre>

Response	<pre>{
	"type": "Profitability",
	"links": [
	{
	"href": "http://myServer.mydomain.com:19000/profitability/rest/v1/fileApplications/BksML12",
	"action": "POST",
	"rel": "self",
	"data": {
	"ruleDimensionName": "Rule",
	"unicode": true,
	"essApplicationServer": "EssbaseCluster-1",
	"applicationType": "MANAGEMENT_LEDGER",
	"balanceDimensionName": "Balance",
	"sharedServicesProject": "EssbaseCluster-1",
	"instanceName": "PROFITABILITY_WEB_APP",
	"description": "description",
	"webServer": "slc06vya.mydomain.com"
	}
	}
	],
	"status": 0,
	"details": "Application for Flat file import has been created with name :: BksML12",
	"statusMessage": "SUCCESS"
	}</pre>

Process Type	Asynchronous

Table 2-44 Request Parameters

Parameter	Description	Example
applicationName	Application Name	BksML1
applicationType	Application Type	Valid values are: MANAGEMENT_LEDGER DETAILED or DETAIL
ruleDimensionName	For Management Ledger applications (optional) for overriding the Rule dimension name	Rule
balanceDimensionName	For Management Ledger applications (optional) for overriding the Balance dimension name	Balance
measuresDimensionName	For Detailed applications (optional) for overriding the default Measures dimension name.	MeasuresDP

Response Values: None

3
APIs for Management Ledger

Related Topics

- ML Deploy Cube
- Run ML Clear POV
- Copy POV Data
- Run ML Calculation
- ML Rule Balancing
- ML Essbase Data Load
- Export ML Template
- Import ML Template
- Create ML Application Using Master Cube Method

ML Deploy Cube

Use this operation to deploy or redeploy the calculation cube for a selected Management Ledger application.

Table 3-1 ML Deploy Cube

Name	Description
Action Type	POST
Request URI	http://{HostName}/profitability/rest/v1/applications/{applicationName}/jobs/ledgerDeployCubeJob

Request Payload	<pre>{
	"isKeepData":"true",
	"isReplaceCube":"true",
	"isRunNow":"true",
	"comment":"Test Ml Deploy"
	}</pre>

Table 3-1 (Cont.) ML Deploy Cube

Name	Description

Response	{
	"type": "Profitability",
	"links": [
	{
	"href": "http://slc06vya.mydomain.com:19000/profitability/rest/v1/applications/LM1T2/jobs/ledgerDeployCubeJob",
	"action": "POST",
	"rel": "self",
	"data": {
	"isRunNow": true,
	"isKeepData": true,
	"isReplaceCube": true,
	"comment": "Test Ml Deploy"
	}
	},
	{
	"href": "http://SLC06VYA.mydomain.com:19000/profitability/rest/v1/applications/jobs/ChecktaskStatusJob/LM1T2_LM1T2_DeployCube_D20160113T065447_b64_1",
	"action": "GET",
	"rel": "Job status"
	}
	],
	"status": -1,
	"details": "LM1T2_LM1T2_DeployCube_D20160113T065447_b64_1",
	"statusMessage": "IN_PROGRESS"
	}

Process Type	Asynchronous

Table 3-2 Request Parameters

Parameter	Description	Example
applicationName	Application Name	LM1T2

Table 3-3 Response Values

Parameter	Description	Example
details	CES Task Id	LM1T2_LM1T2_DeployCube_D20160113T051333_38e_1

Run ML Clear POV

Use this operation to clear model artifacts and data from a POV combination for Management Ledger applications. This operation is equivalent to the functionality supported by the Point of View Manager screen's Clear POV Data control

Table 3-4 Run ML Clear POV

Name	Description
Action Type	POST
Request URI	http://{HostName}/profitability/rest/v1/applications/{applicationName}/povs/{povMemberGroup}/jobs/clearPOVJob

Request Payload	<pre>{
	"isManageRule":"true",
	"isInputData":"true",
	"isAdjustmentValues":"true",
	"isAllocatedValues":"true",
	"stringDelimter":"_"
	}</pre>

Table 3-4 (Cont.) Run ML Clear POV

Name	Description

Response	{
	"type": "Profitability",
	"links": [
	{
	"href": "http://slc06vya.mydomain.com:19000/profitability/rest/v1/applications/LM1T2/povs/2014_January_Actual/jobs/clearPOVJob",
	"action": "POST",
	"rel": "self",
	"data": {
	"isAllocatedValues": true,
	"isInputData": true,
	"isManageRule": true,
	"stringDelimter": "_",
	"isAdjustmentValues": true
	}
	},
	{
	"href": "http://SLC06VYA.mydomain.com:19000/profitability/rest/v1/applications/jobs/ChecktaskStatusJob/LM1T2_LM1T2_ClearMLPOV_D20160113T065742_62d_1",
	"action": "GET",
	"rel": "Job status"
	}
	],
	"status": -1,
	"details": "LM1T2_LM1T2_ClearMLPOV_D20160113T065742_62d_1",
	"statusMessage": "IN_PROGRESS"
	}

Process Type	Asynchronous

Table 3-5 Request Parameters

Parameter	Description	Example
applicationName	Application Name	LM1T2
povMemberGroup	POV member group	2023_January_Actual

Table 3-6 Response Values

Parameter	Description	Example
details	CES Task Id	LM1T2_LM1T2_ClearMLPOV_D20160113T054651_746_1

Copy POV Data

Use this operation to copy model artifacts and data from a Source POV combination to a Destination POV combination for Management Ledger applications. This operation is equivalent to the functionality supported by the Point of View Manager screen's Copy POV Data control.

Table 3-7 Copy POV Data

Name	Description
Action Type	POST
Request URI	http://{HostName}/profitability/rest/v1/applications/{applicationName}/povs/{srcPOVMemberGroup}/jobs/copyPOVJob/{destPOVMemberGroup}
Request Payload	{ "isManageRule":"true", "isInputData":"true", "isAdjustmentValues":"true", "isAllocatedValues":"true", "stringDelimter":"_", "createDestPOV":"true" }

Table 3-7 (Cont.) Copy POV Data

Name	Description

Response	{
	"type": "Profitability",
	"links": [
	{
	"href": "http://myServer.mydomain.com:19000/profitability/rest/v1/applications/LM1T2/povs/2014_January_Actual/jobs/copyPOVJob/2014_March_Actual",
	"action": "POST",
	"rel": "self",
	"data": {
	"isAllocatedValues": true,
	"isInputData": true,
	"createDestPOV": true,
	"isManageRule": true,
	"stringDelimter": "_",
	"isAdjustmentValues": true
	}
	},
	{
	"href": "http://myServer.mydomain.com:19000/profitability/rest/v1/applications/jobs/ChecktaskStatusJob/LM1T2_LM1T2_CopyMLPOV_D20160113T065943_75b_1",
	"action": "GET",
	"rel": "Job status"
	}
	],
	"status": -1,
	"details": "LM1T2_LM1T2_CopyMLPOV_D20160113T065943_75b_1",
	"statusMessage": "IN_PROGRESS"
	}

Process Type	Asynchronous

Table 3-8 Request Parameters

Parameter	Description	Example
applicationName	Application Name	LM1T2
srcPOVMemberGroup	Source POV member group	2023_January_Actual
destPOVMemberGroup	Destination POV member group	2023_March_Actual

Table 3-9 Response Values

Parameter	Description	Example
details	CES Task Id	LM1T2_LM1T2_ClearMLPOV_D20160113T054651_746_1

Run ML Calculation

Use this operation to run calculations for a selected Management Ledger application.

Table 3-10 Run ML Calculation

Name	Description
Action Type	POST
Request URI	http://{HostName}/profitability/rest/v1/applications/{applicationName}/povs/{povGroupMember}/jobs/runLedgerCalculationJob

Request Payload	{
	"isClearCalculated":"true",
	"isExecuteCalculations":"true",
	"isRunNow":"true",
	"comment":"This is run by user1",
	"subsetStart":10,
	"subsetEnd":20,
	"ruleName":"Utilities Expense Adjustment",
	"ruleSetName":"Occupancy Expense Allocations",
	"exeType":"ALL_RULES",
	"stringDelimter":"_"
	}

Table 3-10 (Cont.) Run ML Calculation

Name	Description

Response	{
	"type": "Profitability",
	"links": [
	{
	"href": "http://slc06vya.mydomain.com:19000/profitability/rest/v1/applications/BksML1/povs/2010_January_Actual/jobs/runLedgerCalculationJob",
	"action": "POST",
	"rel": "self",
	"data": {
	"isExecuteCalculations": true,
	"subsetStart": 10,
	"ruleSetName": "Occupancy Expense Allocations",
	"comment": "This is run by user1",
	"subsetEnd": 20,
	"stringDelimter": "_",
	"ruleName": "Utilities Expense Adjustment",
	"isClearCalculated": true,
	"isRunNow": true,
	"exeType": "ALL_RULES"
	}
	},
	{
	"href": "http://SLC06VYA.mydomain.com:19000/profitability/rest/v1/applications/jobs/ChecktaskStatusJob/BksML1_BksML1_RunCalcs_D20160113T070358_1da_1",
	"action": "GET",
	"rel": "Job status"
	}
	],
	"status": -1,
	"details": "BksML1_BksML1_RunCalcs_D20160113T070358_1da_1",
	"statusMessage": "IN_PROGRESS"
	}

Process Type	Asynchronous

Table 3-11 Request Parameters

Parameter	Description	Example
applicationName	Application Name	BksML1
povGroupMember	POV member group	2023_January_Actual

Table 3-12 Response Values

Parameter	Description	Example
Details	CES Task Id	BksML1_BksML1_RunCalcs_D20160113T060906_868_1

ML Rule Balancing

Use this operation to retrieve Rule Balancing Data for a particular POV for a given Management Ledger application.

Table 3-13 ML Rule Balancing

Name	Description
Action Type	GET
Request URI	http://{HostName}/profitability/rest/v1/applications/{applicationName}/povs/ {povGroupMembers}/ruleBalance?queryParameter={"modelViewName":"modelViewName"}
	Note: If the model view name is empty, the default model view is used.
Request Payload	

Table 3-13 (Cont.) ML Rule Balancing

Name	Description

Response	{
	"items": [
	{
	"ruleNumber": "",
	"rules": [],
	"balanceTypeRule": true,
	"scale": 2,
	"sequence": 0,
	"name": "NoRule",
	"description": null,
	"runningBalance": 49357098.03,
	"balance": 49357098.03,
	"allocationIn": null,
	"allocationOut": null,
	"adjustmentIn": null,
	"adjustmentOut": null,
	"input": 49357098.03,
	"runningRemainder": 49357098.03,
	"remainder": 49357098.03,
	"netChange": null,
	"offset": null,
	"inputAsString": "49,357,098.03",
	"adjInAsString": "-",
	"adjOutAsString": "-",
	"allocInAsString": "-",
	"allocOutAsString": "-",
	"balanceAsString": "49,357,098.03",
	"runningBalanceAsString": "49,357,098.03",
	"runningRemainderAsString": "49,357,098.03",
	"remainderAsString": "49,357,098.03",
	"netChangeAsString": "-",
	"offsetAsString": "-"
	},
	],
	"type": "Profitability",
	"links": [
	{
	"href": "http://myServer.mydomain.com:19000/
	profitability/rest/v1/applications/Ex3F1/povs/
	FY14_JUN_Actual_Working/ruleBalance?
	queryParameter=%7B%22modelViewName%22:%22BU%2010601%2
	2%7D",
	"action": "GET",
	"rel": "self",
	"data": {
	"modelViewName": "BU 10601"
	}
	}

Table 3-13 (Cont.) ML Rule Balancing

Name	Description
	], "status": 0, "details": "", "statusMessage": "SUCCESS" }
Process Type	Synchronous

Table 3-14 Request Parameters

Parameter	Description	Example
applicationName	Application Name	Ex3F1
povGroupMember	POV member group	FY23_JUN_Actual_Working
modelViewName	Model view name	BU 10601

Response Values: None

ML Essbase Data Load

Use this resource to load data to Oracle Essbase for a Management Ledger application without using the EAS Console. Note that it can upload a local data file to Profitability and Cost Management to be used for the data load to Essbase. To do this, the client code needs to attach the local data file content to the payload of this resource.

Table 3-15 ML Essbase Data Load

Name	Description
Action Type	POST
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/{applicationName}/ jobs/essbaseDataLoadJob

Request Payload	{
	"clearAllDataFlag":"true",
	"dataLoadValue":"OVERWRITE_EXISTING_VALUES",
	"rulesFileName":"rule.txt",
	"dataFileName":"input.txt"
	}

Table 3-15 (Cont.) ML Essbase Data Load

Name	Description

Response	<pre>{
	  "type": "Profitability",
	  "links": [
	  {
	  "href": " http://myServer.mydomain.com:19000/profitability/rest/v1/applications/BksML12/ jobs/essbaseDataLoadJob ",
	  "action": "POST",
	  "rel": "self",
	  "data": {
	  "clearAllDataFlag":"true",
	  "dataLoadValue":"OVERWRITE_EXISTING_VALUES",
	  "rulesFileName":"rule.txt",
	  "dataFileName":"input.txt"
	  }
	  }, {
	  "href": "http://myServer.mydomain.com:19000/profitability/rest/v1/applications/jobs/ChecktaskStatusJob/BksML12_BksML12_LoadData_D20160118T051020_ba8_1",
	  "action": "GET",
	  "rel": "Job status"
	  }
	  ],
	  "status": -1,
	  "details": "BksML12_BksML12_LoadData_D20160118T051020_ba8_1",
	  "statusMessage": "In_Progress"
	}</pre>

Process Type	Asynchronous

Table 3-16 Request Parameters

Parameter	Description	Example
applicationName	Application Name	BksML12
clearAllDataFlag	Clear existing data before load: true/false	
dataLoadValue	OVERWRITE_EXISTING_VALUES ADD_EXISTING_VALUES	
rulesFileName	Name of the rule file already present in import_export location of the server machine. (Optional parameter)	Rule1.txt

Table 3-16 (Cont.) Request Parameters

Parameter	Description	Example
dataFileName	Name of the data file already present in the import_export location of the server. (Mandatory parameter)	Inpdat1.txt

Response Values: None

Export ML Template

Use this operation to export Management Ledger applications as a template zip file. The same template file can be imported into other Profitability and cost management instances. When an already existing file name is used then it will overwrite content in the existing file.

> **Note:**
>
> Input Data export is not currently available when a template is exported using the REST API. However, this feature is supported from the user interface.

Table 3-17 Export ML Template

Name	Description
Action Type	POST
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/Ex3F3/jobs/templateExportJob

Request Payload	{
	"fileName":"testFile"
	}

Table 3-17 (Cont.) Export ML Template

Name	Description

Response	{
	"type": "Profitability",
	"links": [{
	"href": "http://myServer.mydomain.com:19000/profitability/rest/v1/applications/Ex3F3/jobs/templateExportJob?fileName=test123",
	"action": "POST",
	"rel": "self",
	"data": {}
	}, {
	"href": "http://myServer.mydomain.com:19000/profitability/rest/v1/applications/jobs/ChecktaskStatusJob/Ex3F3_Ex3F3_ExportTemplate_D20160113T090737_166_1",
	"action": "GET",
	"rel": "Job status"
	}],
	"status": -1,
	"details": "Ex3F3_Ex3F3_ExportTemplate_D20160113T090737_166_1",
	"statusMessage": "IN_PROGRESS"
	}

Process Type	Asynchronous

Table 3-18 Request Parameters

Parameter	Description	Example
fileName	File name	testFile

Table 3-19 Response Values

Parameter	Description	Example
details	CES task id	Ex3F3_Ex3F3_ExportTemplate_D20160113T051000_d66_1

Import ML Template

Use this operation to import an entire Management Ledger application from a template zip file (which was created by a template export operation). This creates a new application which contains the same dimension metadata and artifacts as the exported application.

Table 3-20 Import ML Template

Name	Description
Action Type	POST
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/BksTs/jobs/templateImportJob

Request Payload	{
	"description":"description",
	"instanceName":"PROFITABILITY_WEB_APP",
	"essApplicationServer":"EssbaseCluster-1",
	"sharedServicesProject":"EssbaseCluster-1",
	"applicationType": "MANAGEMENT_LEDGER",
	"fileName":" testFile12345.zip",
	"isApplicationOverwrite":"true"
	}

Response	{
	"type": "Profitability",
	"links": [{
	"href": "http://slc06vya.mydomain.com:19000/profitability/rest/v1/applications/BksT1/jobs/templateImportJob",
	"action": "POST",
	"rel": "self",
	"data": {
	"instanceName": "PROFITABILITY_WEB_APP",
	"essApplicationServer": "EssbaseCluster-1",
	"sharedServicesProject": "EssbaseCluster-1",
	"description": "description",
	"isApplicationOverwrite": true,
	"applicationType": "MANAGEMENT_LEDGER",
	"fileName": " testFile12345.zip"
	}
	}],
	"status": 1,
	"details": "the given File not found : filename is testFile12345.zip",
	"statusMessage": "ERROR"
	}

Process Type	Asynchronous

Table 3-21 Request Parameters

Parameter	Description	Example
description	Description for application (mandatory parameter)	description
instanceName	Instance name for the application (mandatory parameter)	PROFITABILITY_WEB_APP
essApplicationServer	Essbase Application Server name (mandatory parameter)	EssbaseCluster-1
sharedServicesProject	Shared Services Project Name (mandatory parameter)	EssbaseCluster-1
applicationType	Type of application	
fileName	Name of the zip file already present in export import location. (Mandatory parameter)	
isApplicationOverwrite	if value is true then it will delete existing application and then creates new application & if false then it will return with message that application already exist(default is males and it is an optional parameter)	true

Table 3-22 Response Values

Parameter	Description	Example
details	CES task id	AppMgmt_AppMgmt_ImportTemplate_D20160113T051745_d85_1

Create ML Application Using Master Cube Method

Use this operation to create a Management Ledger application using an Oracle Essbase master cube.

Table 3-23 Create ML Application Using Master Cube Method

Name	Description
Action Type	POST
Request URI	http://:///myServer.mydomain.com:19000/profitability/rest/v1/fileApplications/{applicationName}

Table 3-23 (Cont.) Create ML Application Using Master Cube Method

Name	Description

Request Payload	<pre>{
	"description":"description",
	"instanceName":"PROFITABILITY_WEB_APP",
	"essApplicationServer":"EssbaseCluster-1",
	"sharedServicesProject":"EssbaseCluster-1",
	"applicationType":"MANAGEMENT_LEDGER",
	"essApplication":"BksMast",
	"essApplicationDatabase":"BksMast",
	"dimensions":
	["Year","Period","Accounts","Products"],
	"webServer":"slc06vya.mydomain.com"
	}</pre>

Table 3-23 (Cont.) Create ML Application Using Master Cube Method

Name	Description

Response	{
	"type": "Profitability",
	"status": -1,
	"statusMessage": "In Progress",
	"details":
	"AppMgmt_DeployApplication_D20160127T030614_464",
	"links": [{
	"href": "http://
	slc06vya.mydomain.com:19000/profitability/
	rest/v1/aamApplications/Bks11",
	"action": "POST",
	"rel": "self",
	"data": {
	"dimensions": ["Year", "Period",
	"Accounts", "Products"],
	"essApplicationDatabase": "BksMast",
	"essApplicationServer":
	"EssbaseCluster-1",
	"applicationType":
	"MANAGEMENT_LEDGER",
	"essApplication": "BksMast",
	"sharedServicesProject":
	"EssbaseCluster-1",
	"instanceName":
	"PROFITABILITY_WEB_APP",
	"description": "description",
	"webServer": "slc06vya.mydomain.com"
	}
	}, {
	"href": "http://
	SLC06VYA.mydomain.com:19000/profitability/
	rest/v1/applications/jobs/ChecktaskStatusJob/
	AppMgmt_DeployApplication_D20160127T030614_464",
	"action": "GET",
	"rel": "Job Status"
	}]
	}

Process Type	Asynchronous

Table 3-24 Request Parameters

Parameter	Description	Example
applicationName	Application Name	ABks12
description		
instanceName	Instance Name	PROFITABILITY_WEB_APP

Table 3-24 (Cont.) Request Parameters

Parameter	Description	Example
essApplicationServer	Essbase Application Server	EssbaseCluster-1
sharedServicesProject	Shared Services Project	EssbaseCluster-1
applicationType	Application Type	MANAGEMENT_LEDGER
webServer	HPCM web server	myServer.mydomain.com
ruleDimensionName	Rule Dimension Name	Rule
balanceDimensionName	Balance Dimension Name	Balance
unicode	true/fasle	

Table 3-25 Response Values

Parameter	Description	Example
Application Name	Application Name	ABks12

4

APIs for Detailed Profitability

Related Topics

- Get Stages by Application
- Import from Staging
- Copy POV Data
- Clear POV Data
- Prepare Detailed Views for Reporting
- Apply Bulk Edit
- Get DP Driver Definitions
- Run DP Calculation
- Get Assignment Rule Definitions

Get Stages by Application

Use this operation to retrieve all stage details for the given application name. This operation returns name and display order of a stage by using the request rest URI.

Table 4-1 Get Stages by Application

Name	Description
Action Type	GET
Request URI	http:///://myServer.mydomain.com:19000/profitability/rest/v1/applications/BksDP30/stages
Request Payload	

Table 4-1 (Cont.) Get Stages by Application

Name	Description

Response	{
	"items": [{
	"stageName": "Ledger Data",
	"displayOrder": 1
	}, {
	"stageName": "Activity",
	"displayOrder": 2
	}, {
	"stageName": "Cost Objects",
	"displayOrder": 3
	}, {
	"stageName": "Profitability",
	"displayOrder": 4
	}, {
	"stageName": "test1",
	"displayOrder": 5
	}, {
	"stageName": "test2",
	"displayOrder": 6
	}],
	"type": "Profitability",
	"links": [{
	"href": "http://slc06vya.mydomain.com:19000/profitability/rest/v1/applications/BksDP30/stages",
	"action": "GET",
	"rel": "self"
	}],
	"status": 0,
	"details": "",
	"statusMessage": "SUCCESS"
	}

Process Type	Synchronous

Table 4-2 Request Parameters

Parameter	Description	Example
Application Name n	Application name	

Table 4-3 Response Values

Parameter	Description	Example
List of stages	Stages	

Import from Staging

Use this operation to execute the selected import configuration into a Profitability and Cost Management application.

Table 4-4 Import from Staging

Name	Description
Action Type	POST
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/BksDP30/jobs/importConfigJob

Request Payload	{
	"importConfigName":"Import_Config_Test"
	}

Response	{
	"type": "Profitability",
	"links": [{
	"href": "http://slc06vya.mydomain.com:19000/profitability/rest/v1/applications/BksDP30/jobs/importConfigJob",
	"action": "POST",
	"rel": "self",
	"data": {
	"importConfigName": "Import_Config_Test"
	}
	}],
	"status": 1,
	"details": "Invalid import configuration name: ''Import_Config_Test'' associated with application: ''BksDP30''",
	"statusMessage": "ERROR"
	}

Process Type	Synchronous

Table 4-5 Request Parameters

Parameter	Description	Example
importConfigName	Name of the import configuration to be executed	importConfigName

Table 4-6 Response Values

Parameter	Description	Example
result	CES Task ID generated for executing the Import Configuration.	Ex3F3_Ex3F3_ExportTemplate_D20160113T051000_d66_1

Copy POV Data

Use this operation to copy Model artifacts and Data from a Source POV combination to a Destination POV combination. This operation is equivalent to the functionality supported from the UI by selecting Manage Model, then POV Manager, and then Copy.

Table 4-7 Copy POV Data

Name	Description
Action Type	POST
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/BksDP30/povs/2010_February_Actual/jobs/copySpDpPOVJob/2012_January_Actual

Request Payload	{
	"copyCostLayerData":"true",
	"copyRevenueLayerData":"true",
	"copyAssignments":"true",
	"copyDriverAssociations":"false",
	"copyDriverValues":"true",
	"copyCostRevenueValues":"false",
	"copyCalculationRules":"false",
	"stringDelimter":"_"
	}

Table 4-7 (Cont.) Copy POV Data

Name	Description

Response	{
	"type": "Profitability",
	"links": [{
	"href": "http://
	slc06vya.mydomain.com:19000/profitability/
	rest/v1/applications/BksDP30/povs/
	2010_February_Actual/jobs/copySpDpPOVJob/
	2012_January_Actual",
	"action": "POST",
	"rel": "self",
	"data": {
	"copyDriverAssociations": false,
	"copyCostLayerData": true,
	"copyAssignments": true,
	"stringDelimter": "_",
	"copyCalculationRules": false,
	"copyCostRevenueValues": false,
	"copyDriverValues": true,
	"copyRevenueLayerData": true
	}
	}, {
	"href": "http://
	SLC06VYA.mydomain.com:19000/profitability/
	rest/v1/applications/jobs/ChecktaskStatusJob/
	BksDP30_BksDP30_CopyPOV_D20160114T041354_115_1",
	"action": "GET",
	"rel": "Job status"
	}],
	"status": -1,
	"details":
	"BksDP30_BksDP30_CopyPOV_D20160114T041354_115_1",
	"statusMessage": "In Progress"
	}

Process Type	Asynchronous

Table 4-8 Request Parameters

Parameter	Description	Example
copyCostLayerData	Boolean flag specifying whether Cost Layer data should be copied. Valid values are TRUE or FALSE.	true
copyRevenueLayerData	Boolean flag specifying whether Revenue Layer data should be copied. Valid values are TRUE or FALSE.	true

Table 4-8 (Cont.) Request Parameters

Parameter	Description	Example
copyAssignments	Boolean flag specifying whether assignments data should be copied. Valid values are TRUE or FALSE.	true
copyDriverAssociations	Boolean flag specifying whether driver associations data should be copied. Valid values are TRUE or FALSE.	true
copyDriverValues	Boolean flag specifying whether driver data should be copied. Valid values are TRUE or FALSE.	true
copyCalculationRules	Boolean flag specifying whether Calculation Rules should be copied for a Profitability and Cost Management Detailed Application. Valid values are TRUE or FALSE.	true
stringDelimter	Delimiter used to separate logical entities in parameters. For example, when passing POVs at a time, please use this to delimit POVs _ Eg: 2009_January_ActuaL when passing POV MEMBERS TO 2009, January , Actual	_

Table 4-9 Response Values

Parameter	Description	Example
details	CES task id	BksDP30_BksDP30_CopyPOV_D20160112T015313_580_1

Clear POV Data

Use this operation to clear the POV data for selection stage and other details.

Table 4-10 Clear POV Data

Name	Description
Action Type	POST
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/BksDP30/povs/2010_January_Actual/jobs/clearSpDpPOVJob

Table 4-10 (Cont.) Clear POV Data

Name	Description

Request Payload	<pre>{
	"stages":"Ledger Data_Activity",
	"clearCostLayer":"true",
	"clearRevenueLayer":"true",
	"clearDriverSelectionRules":"true",
	"clearDriverSelectionExceptions":"false",
	"clearAssignmentRuleSelections":"true",
	"clearRegularAssignments":"false",
	"clearCalculationRules":"false",
	"stringDelimter":"_"
	}</pre>

Response	<pre>{
	"type": "Profitability",
	"links": [{
	"href": "http://
	slc06vya.mydomain.com:19000/profitability/
	rest/v1/applications/BksDP30/povs/
	2010_January_Actual/jobs/clearSpDpPOVJob",
	"action": "POST",
	"rel": "self",
	"data": {
	"clearCalculationRules": false,
	"clearRevenueLayer": true,
	"stages": "Ledger Data_Activity",
	"clearDriverSelectionExceptions":
	false,
	"clearRegularAssignments": false,
	"clearDriverSelectionRules": true,
	"clearAssignmentRuleSelections":
	true,
	"clearCostLayer": true,
	"stringDelimter": "_"
	}
	}],
	"status": 0,
	"details": "Cleared POV Successfully",
	"statusMessage": "Success"
	}</pre>

Process Type	Synchronous

Table 4-11 Request Parameters

Parameter	Description	Example
stages	Specify dimension member names of the POV for which this Bulk Edit operation should be applied: stages 1 stages 2 stages 3 stages N, stages N = [displayOrder = (int), example:1 stageName = (string)	Ledger Data_Activity
clearCostLayer	Boolean flag specifying whether Cost Layer should be cleared. Valid values are TRUE or FALSE.	true
clearRevenueLayer	Boolean flag specifying whether Revenue Layers should be cleared. Valid values are TRUE or FALSE.	true
clearDriverSelectionRules	Boolean flag specifying whether Driver Selection Rules should be cleared. Valid values are TRUE or FALSE.	true
clearDriverSelectionExceptions	Boolean flag specifying whether Driver Selection Exceptions should be cleared. Valid values are TRUE or FALSE.	true
clearAssignmentRuleSelections	Boolean flag specifying whether clear assignment rule selection should be cleared. Valid values are TRUE or FALSE.	true
clearRegularAssignments	Boolean flag specifying whether clear regular assignments should be cleared. Valid values are TRUE or FALSE.	true
clearCalculationRules	Boolean flag specifying whether clear calculation rules should be cleared. Valid values are TRUE or FALSE.	true

Table 4-11 (Cont.) Request Parameters

Parameter	Description	Example
stringDelimter	Delimiter used to separate logical entities in parameters For example, when passing POVs at a time, please use this to delimit POVs _ Eg: 2023_January_ActuaL when passing POV MEMBERS TO 2023, January , Actual	

Table 4-12 Response Values

Parameter	Description	Example
Success message	Cleared POV Successfully	

Prepare Detailed Views for Reporting

Use this operation to prepare views an application.

Table 4-13 Prepare Detailed Views for Reporting

Name	Description
Action Type	POST
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/BksDP30/jobs/reportingViewGeneratorJob

Request Payload	
	{
	"dimensions":"Scenario_TScenario"
	}

Table 4-13 (Cont.) Prepare Detailed Views for Reporting

Name	Description

Response	{
	"type": "Profitability",
	"links": [{
	"href": "http://slc06vya.mydomain.com:19000/profitability/rest/v1/applications/BksDP30/jobs/reportingViewGeneratorJob",
	"action": "POST",
	"rel": "self",
	"data": {
	"dimensions": "Scenario_TScenario"
	}
	}],
	"status": 0,
	"details": "Succesful",
	"statusMessage": "SUCCESS"
	}

Process Type	Asynchronous

Table 4-14 Request Parameters

Parameter	Description	Example
dimensions	Optionally, specify the list of name and short name properties for one or more dimensions to be included in generating the reporting views.	Scenario_TScenario

Response Values: None

Apply Bulk Edit

Use this operation to perform Bulk Edit for the given source assignment rules with destination rules or drivers.

Table 4-15 Apply Bulk Edit

Name	Description
Action Type	POST
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/BksDp30/povs/2023_January_Actual/jobs/applyBulkEditJob

Table 4-15 (Cont.) Apply Bulk Edit

Name	Description

Request Payload	<pre>{
	"operation":"ADD_DRIVERS",
	"comment":"bulk edit test",
	"sourceRules":"Apply All Building
	Activities,Apply Business Support Driver,Apply
	CoMarket",
	"drivers":"DRV Business Support",
	"isSelectEntireStageForDelete":"false",
	"isSelectAllRulesOrDriversForDelete":"false",
	"stringDelimter":"_"
	}</pre>

Table 4-15 (Cont.) Apply Bulk Edit

Name	Description

Response	{
	"type": "Profitability",
	"links": [{
	"href": "http://
	myServer.mydomain.com:19000/profitability/
	rest/v1/applications/BksDp30/povs/
	2012_January_Actual/jobs/applyBulkEditJob",
	"action": "POST",
	"rel": "self",
	"data": {
	"drivers": "DRV Business Support",
	"isSelectEntireStageForDelete":
	false,
	
	"isSelectAllRulesOrDriversForDelete": false,
	"operation": "ADD_DRIVERS",
	"stringDelimter": "_",
	"sourceRules": "Apply All Building
	Activities,Apply Business Support Driver,Apply
	CoMarket",
	"comment": "bulk edit test"
	}
	}, {
	"href": "http://
	myServer.mydomain.com:19000/profitability/
	rest/v1/applications/jobs/ChecktaskStatusJob/
	BksDP30_BksDP30_BulkEdit_D20160113T062803_8a4_1",
	"action": "GET",
	"rel": "Job status"
	}],
	"status": -1,
	"details":
	"BksDP30_BksDP30_BulkEdit_D20160113T062803_8a4_1"
	,
	"statusMessage": "IN_PROGRESS"
	}

Process Type	Asynchronous

Table 4-16 Request Parameters

Parameter	Description	Example
Operation	Specify the Bulk Edit operation: BulkEditOperations.ADD_DRIVERS BulkEditOperations.REMOVE_DRIVERS BulkEditOperations.ADD_ASSIGNMENT_RULES BulkEditOperations.REMOVE_ASSIGNMENT_RULES	ADD_DRIVERS
Comment	Specify a comment for this Bulk Edit Operation.	bulk edit test
sourceRules	A List of Source Assignment Rule names being selected for this Bulk Edit Operation	Apply All Building Activities,Apply Business Support Driver,Apply CoMarket
Drivers	Name of the Driver to be applied to the selected Source Assignment Rules as part of this Bulk Edit Operation. Note: Only one Driver name may be provided when using the BulkEditOperations.ADD_DRIVER operation; however, a list of Driver names can be provided when using the BulkEditOperations.REMOVE_DRIVERS operation	DRV Business Support
destinationRules	A List of Destination Assignment Rule names being selected for this Bulk Edit Operation, Note: This value should be passed only with BulkEditOperations.ADD_ASSIGNMENT_RULES and BulkEditOperations.REMOVE_ASSIGNMENT_RULES.	All to Invoice,Building to Invoice
isSelectEntireStageForDelete	Boolean flag specifying if the entire stage should be selected for delete. Valid values are TRUE or FALSE	true

Table 4-16 (Cont.) Request Parameters

Parameter	Description	Example
isSelectAllRulesOrDriversForDelete	Boolean flag specifying if all the destination assignment rules or drivers should be selected for delete. Valid values are TRUE or FALSE.	true
stringDelimter	Delimiter used to separate logical entities in parameters For example, when passing POVs at a time, please use this to delimit POVs Eg: 2009_January_ActuaL when passing POV MEMBERS TO 2009, January , Actual This is optional field.	_

Response Values: None

Get DP Driver Definitions

Use this operation to list all Driver definitions.

Table 4-17 Get DP Driver Definitions

Name	Description
Action Type	GET
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/BksDP30/getDriverDefinitions
Request Payload	

Table 4-17 (Cont.) Get DP Driver Definitions

Name	Description

Response	{
	"items":]{
	"name": "OC 8 Gross Margin 210",
	"type": "Calculated Measure"
	}, {
	"name": "OC 9 Op Expense 210",
	"type": "Calculated Measure"
	}],
	"type": "Profitability",
	"links": [{
	"href": "http://
	myServer.mydomain.com:19000/profitability/
	rest/v1/applications/BksDP30/
	getDriverDefinitions",
	"action": "GET",
	"rel": "self"
	}],
	"status": 0,
	"details": "",
	"statusMessage": "Success"
	}

Process Type	Asynchronous

Response Values: None

Response Values: None

Run DP Calculation

Use this operation to process and run calculations for a selected application.

Table 4-18 Run DP Calculation

Name	Description
Action Type	POST
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/BksDP30/povs/2012_January_Actual/jobs/detailedCalculationProcessJob/2012_January_Actual

Table 4-18 (Cont.) Run DP Calculation

Name	Description

Request Payload	<pre>{
	"comment":"This is run by user1",
	"isClearCalculated":"true",
	"isExecuteCalculations":"true",
	"isCreateContributionDetail":"true",
	"isCreateDetailCalculatedDriverTables":"false",
	"isAbortOnError":"true",
	"isRunSingleCalcRuleSequence":"true",
	"calcRuleSequence":1,
	"isTransferToSrcStgDb":"false",
	"isTransferToDstStgDb":"false",
	"isTransferToContribDb":"false",
	"postScript":"POST",
	"preScript":"PRE",
	"stringDelimter":"_"
	}</pre>

Table 4-18 (Cont.) Run DP Calculation

Name	Description

Response	{
	"type": "Profitability",
	"links": [{
	"href": "http://slc06vya.mydomain.com:19000/profitability/rest/v1/applications/BksDP30/povs/2012_January_Actual/jobs/detailedCalculationProcessJob/2012_January_Actual",
	"action": "POST",
	"rel": "self",
	"data": {
	"isExecuteCalculations": true,
	"isTransferToContribDb": false,
	"isTransferToSrcStgDb": false,
	"postScript": "POST",
	"calcRuleSequence": 1,
	"isRunSingleCalcRuleSequence": true,
	"isTransferToDstStgDb": false,
	"comment": "This is run by user1",
	"isAbortOnError": true,
	"stringDelimter": "_",
	"preScript": "PRE",
	"isClearCalculated": true,
	"isCreateContributionDetail": true,
	"isCreateDetailCalculatedDriverTables": false
	}
	}, {
	"href": "http://SLC06VYA.mydomain.com:19000/profitability/rest/v1/applications/jobs/ChecktaskStatusJob/BksDP30_BksDP30_RunCalcs_D20160113T072425_5fc_1",
	"action": "GET",
	"rel": "Job status"
	}],
	"status": -1,
	"details": "BksDP30_BksDP30_RunCalcs_D20160113T072425_5fc_1",
	"statusMessage": "IN_PROGRESS"
	}

Process Type	Asynchronous

Table 4-19 Request Parameters

Parameter	Description	Example
comment	Specify a comment for this Run DP Calculation Job.	This is run by user1

Table 4-19 (Cont.) Request Parameters

Parameter	Description	Example
isClearCalculated	Boolean flag specifying whether previously calculated values should be cleared. Valid values are TRUE or FALSE	true
isExecuteCalculations	Boolean flag specifying whether calculations should be executed as part of this operation. Valid values are TRUE or FALSE. **Note:** When the isExecuteCalculations flag is set to TRUE, you must provide values for isCreateContributionDetail, isCreateDetailCalculatedDriverTables	true
isCreateContributionDetail	Boolean flag specifying contribution detail should be created. Valid values are TRUE or FALSE.	true
isCreateDetailCalculatedDriverTables	Boolean flag specifying whether previously calculated values should be cleared. Valid values are TRUE or FALSE	false
isAbortOnError	Boolean flag specifying whether to abort on occurrence of any error while processing calculations. Valid values are TRUE or FALSE.	true
isRunSingleCalcRuleSequence	Boolean flag specifying whether to run single calculation Rule Sequence or not. Valid values are TRUE or FALSE. **Note:** When the isRunSingleCalcRuleSequence flag is set to TRUE, you must provide value for calcRuleSequence	true
calcRuleSequence	Valid int value (calculation rule sequence number).	1
isTransferToSrcStgDb	Boolean flag specifying whether to do data transfers to source stage Database or not. Valid values are TRUE or FALSE.	false
isTransferToDstStgDb	Boolean flag specifying whether to do data transfers to destination stage Database or not. Valid values are TRUE or FALSE	false
isTransferToContribDb	Boolean flag specifying whether to do data transfers to Contribution Database or not. Valid values are TRUE or FALSE.	false
postScript	Name of the post-calculation script	POST

Table 4-19 (Cont.) Request Parameters

Parameter	Description	Example
preScript	Name of the pre-calculation script	PRE
stringDelimter	Optional delimiter used to separate logical entities in parameters For example, when passing POVs at a time, please use this to delimit POVs, such as: 2023_January_ActuaL when passing POV MEMBERS TO 2023, January , Actual	_

Response Values: None

Get Assignment Rule Definitions

Use this operation to retrieve all assignment rule definitions (not associations), for a particular stage for a given application.

Table 4-20 Get Assignment Rule Definitions

Name	Description
Action Type	GET
Request URI	http://myServer.mydomain.com:19000/profitability/rest/v1/applications/BksDP30/assignmentRules?queryParameter={"stageName":"Customer Activity Cost"}
Request Payload	

Table 4-20 (Cont.) Get Assignment Rule Definitions

Name	Description

Response	{
	"items": [{
	"name": "Apply Testing"
	}, {
	"name": "Apply Service Calls"
	}, {
	"name": "Apply Ship and Order Assembly"
	}, {
	"name": "Apply Business Support Driver"
	}, {
	"name": "Warehouse Sppt Act Charges All Customer No Product"
	}, {
	"name": "Apply All Building Activities"
	}],
	"type": "Profitability",
	"links": [{
	"href": "http://slc06vya.mydomain.com:19000/profitability/rest/v1/applications/BksDP30/assignmentRules?queryParameter=%7B%22stageName%22:%22Customer%20Activity%20Cost%22%7D",
	"action": "GET",
	"rel": "self",
	"data": {
	"stageName": "Customer Activity Cost"
	}
	}],
	"status": 0,
	"details": "",
	"statusMessage": "Success"
	}

Process Type	Synchronous

Request Parameters: None

Response Values: None

A

References and Sample Client

In this Appendix

Sample Code for Authentication

```
package oracle.epm.webservices.profitability.util;


import com.sun.jersey.api.client.Client;
import com.sun.jersey.api.client.ClientResponse;
import com.sun.jersey.api.client.WebResource;
import com.sun.jersey.core.util.Base64;


public class RestClient {

        public static void main(String[] args) {
                // TODO Auto-generated method stub
                String auth = new String(Base64.encode("admin:password1"));
                ClientConfig clientConfig = new DefaultClientConfig();
                clientConfig.getClasses().add(JsonMessageBodyReader.class);

clientConfig.getFeatures().put(JSONConfiguration.FEATURE_POJO_MAPPING,
                Boolean.TRUE);
                Client client = Client.create(clientConfig);
         WebResource webResource =
    client.resource("http://myServer.mydomain.com:19000/profitability/
rest/v1/applications");
 ClientResponse response = webResource.header("Authorization", "Basic " +
auth).type("application/json").accept("application/
json")..accept(MediaType.APPLICATION_JSON).get(ClientResponse.class);
         ProfitabilityRestResponse<DriverDTO> response = cr.getEntity(new
        GenericType<ProfitabilityRestResponse<DriverDTO>>() {});
                         if (cr.getStatus() >= 400) {
                            if(response.getError() != null) {
                               String message =
response.getError().getMessage();
                               System.out.println(message);
                            }
                            else {
                               String message = cr.getEntity(String.class);
                               System.out.println(message);
                            }
                         } else {
```

```
                                List<DriverDTO> drivers =
response.getItems();
                                if(drivers == null ||
drivers.isEmpty()) {
                                System.out.println("No Drivers
definitions are available for the application: " +
appName);
                                return;
                            }

                            for(DriverDTO driver : drivers) {
                                System.out.println("Driver name = " +
driver.getName() + ", Driver type = " +

driver.getType());
                            }
                        }
    }
```

Sample Client

Overview

The Sample Client is intended as a guide, to assist you in creating your own custom scripts for calling REST Web services to automate your Profitability and Cost Management tasks. It consists of pre-compiled Java code for calling REST services, and an executable script (one for Windows and one for Linux) that serves as a wrapper for calling the Java code.

The Java code is pre-compiled, and the source code is included in case you need to modify and recompile the code. You can only run the Sample Client on the same server as Profitability and Cost Management.

The following instructions are designed to be used on a Windows server; modify them accordingly for Linux.

Setting Up the Sample Client Environment

To set up the sample client environment:

1. Open a command window on the same machine on which Oracle Hyperion Profitability and Cost Management is installed
2. Edit the `hpm_ws_client.properties` to reflect your local settings:
   - Delimiter used to separate String literals in paramters `string.delimiter=_`
   - Delimiter used to separate logical entities in parameters:
     When passing multiple POVs at a time, use # as the delimiter to delimit POVs. For example, `2023_January_Actual#2023_March_Actual` to pass two POVs
   - SecurityPolicy associated with ProfitabilityService, that needs to be used by the sample client.

> **Note:**
>
> The sample client is programmed to work with only one value: `USERNAME_TOKEN`

`hpcm.service.security.policy=USERNAME_TOKEN`

- The following two values are needed for security policy USERNAME_TOKEN
  - #HSS user name of the Profitability user. `hss.username=admin`
  - #Password of the username above. `hss.password=password123`
- The following two values are needed for RESTFUL Web Service URL:
  - HPCM REST URL to access. Example:

    ```
    http://myServer.mydomain.com:19000
     hpcm.rest.url.host=http://myServer.mydomain.com:19000
    ```

  - HPCM REST URL version which is to be accessed. Example: `version=v1`

3. From a command or shell window, set the following environment variables:

Table A-1 Environment Variables for the Sample Client

Variable	Location
JAVA_HOME	Location where Java Development Kit is available: **Windows:** `SET JAVA_HOME=C:\Oracle\Middleware\Oracle_Home\oracle_common\jdk` **Linux:** `JAVA_HOME=/usr/c/Oracle/Middleware/Oracle_Home/oracle_common/jdk`
MIDDLEWARE_HOME	Location where Oracle Middleware home is installed. **Windows:** `SET MIDDLEWARE_HOME=C:\Oracle\Middleware For Linux: export MIDDLEWARE_HOME=/usr/c/Oracle/Middleware`

> **Note:**
>
> In a command window, navigate to `%EPM_ORACLE_HOME%\products\Profitability\samples\wsclient` and then enter the command `hpm_ws_rest_client.bat -help` to get a list of available functions.

4. Use the format and operations specified in the sample client file to build your custom script. See Using the Sample Client File for details.

Using the Sample Client File

The sample client file is intended as a guide to help you build your own custom scripts to access Profitability and Cost Management data through Web Services. The sample client files are available in `%EPM_ORACLE_HOME%/products/Profitability/samples/wsclient`. These files have been created using Batch Script (Windows) and Shell Script (UNIX/Linux).

To use the sample client script file:

1. In the command window, go to `%EPM_ORACLE_HOME%\products\Profitability\samples\wsclient`. See Setting Up the Sample Client Environment.
2. Enter the following command to get information on a specific operation:
   `hpm_ws_rest_client.bat - help`
3. Select the operation to be performed and enter the following command:
   hpm_ws_rest_client.bat - help *OPERATION_NAME*

   For example, to see the usage details of the `getPovs` operation, enter the command:
   `hpm_ws_rest_client.bat - help getPovs`

   To use the sample client script file to perform an operation, enter the command:

   hpm_ws_rest_client.bat *OPERATION_NAME PARAMETERS*

   See for samples in Usage Examples.

Usage Examples

Example 1: List All Applications - To obtain a list of all the available applications, enter the command:

```
hpm_ws_client.bat getApplications
```

Example 2: List All POVs - To obtain a list of all POVs for a given application, enter the command:

```
hpm_ws_rest_client.bat getPovs APPLICATION_NAME
```

Recompiling the Code

The installation deploys both the compiled binary and the source for the REST sample client:

- As a compiled binary file, in `wsclient/lib/hpcm-wsclient-sample.jar`
- As source code, in `wsclient/src/oracle/epm/webservices/rest/profitability/client`

If the source code needs to be recompiled for any reason, you can recompile using Ant. The `build.xml` file for Ant is available in the `wsclient` folder.

To recompile the source code:

Compile the source code with the clean and build targets using a command similar to the following.

```
ant clean build
```