Hitachi Data Instance Director

Version 6.7

REST API User guide

MK-93HDID019-02
14th November 2018

Contents

Intended audience

This guide is intended for those who wish to develop a script or application which utilizes Hitachi Data Instance Director (HDID).

Introduction to the Hitachi Data Instance Director REST API

Hitachi Data Instance Director exposes a Representational State Transfer (REST) application programming interface (API) solely over HTTPS which can be accessed using any scripting or programming language which supports making HTTPS requests.

The API requests are governed by Role Based Access Control (RBAC) which itself can be configured using the API (or via the HDID graphical user interface (GUI)).

Once the user is authenticated (by logging in with valid credentials) they are able to perform tasks in the roles for which they are authorized.

The HDID web interface solely uses the REST API so 100% of the functionality available in the HDID web interface is available through the REST API.

REST URL root

The Hitachi Data Instance Director REST API is served from the master node and can either be accessed locally or remotely. The root URL is constructed from a number of parts as follows:

https://<host>[:<port>]/HDID/

- host - is the IP or DNS resolvable name of the master machine. Or “localhost” if accessing the API from the master machine directly.
- port – The port on which the UI / API is being servers. By default this is 443 but can be changed during master installation.

HTTP methods and responses

Requests to the REST API use the common methods for HTTP/1.1. The basic methods allow users to read (GET), update (PUT), create (POST) and delete (DELETE) resources.

Responses also use the common status codes for HTTP/1.1. Important status codes are as follows:

- 200 OK – The request has succeeded.
- 201 Created – The request has been fulfilled and the resource created.
- 202 Accepted – The request has been accepted but the progressing is not complete. The response should include an Id for a Job which can be queried to get information about the progress of the requested task.
- 400 Bad Request – The request could not be understood the client should not repeat the request without modifications.
- 401 Unauthorized – The request requires authentication.
- 404 Not Found – The requested handler or resource was not found.
- 500 Internal Server Error – The server encountered an unexpected condition which has prevented it from fulfilling the request.

Further information regarding HTTP/1.1 and its supported methods and status codes can be found in the Hypertext Transfer Protocol specification https://tools.ietf.org/html/rfc2616

Request and response formats

Requests can be made with URL query data and/or JSON body data depending on the resource and method being used. Responses are always returned as JSON.

API Concepts

Resources

Almost all interactions using the API are done against *resources*. Resources typically represent an object which can be retrieved, created, modified or deleted, as permitted by the resource handler and user permissions.

All resources are referenced by their pluralized type name, e.g., Nodes not Node, even when retrieving a single resource.

Resources can be retrieved as a list which can include filtering, sorting and field query data or as an individual object by appending the resource Id to the URL. When retrieving a list of resources the response will include a "pageInfo" object. This contains properties describing the current paging count and offset, the total amount of resources which match the query and whether they have all been returned within the response, for example:

```
"pageInfo": {
    "offset": 0,
    "count": 10
    "totalCount": 10,
    "end": true
}
```

To retrieve all Nodes (which the user has access to):

https://localhost/HDID/master/NodeManager/objects/Nodes/

Or to retrieve a single node:

https://localhost/HDID/master/NodeManager/objects/Nodes/:id

Services

Services allow actions to be performed on handlers where the action does not relate to an individual resource. Service endpoints only allow action requests and can either change the state of many resources or none at all.

An example of a service action is the one used for logging in:

https://localhost/HDID/master/UIController/services/Sessions/actions/login/invoke

Handlers

Resources are exposed by handlers. Each handler is responsible for the management of a number of resources. Most handlers are available exclusively on the master node, however some handlers are available only on client nodes and some handlers are available on all nodes.

To access a resource provided by a handler, the handler address (node) and name are required. The API does not handle requests which are not directed to a handler.

Actions

Action requests can be performed on resources or services and are used to perform specific tasks. Action requests often modify multiple resources or the entire system state. In a few situations no resources are modified but the response contains data otherwise unavailable for the associated resource.

Actions are useful in describing a task being performed against a resource. For example to authorize a client node the following action request would be made:

https://localhost/HDID/master/NodeManager/objects/Nodes/:id/actions/authorize/invoke

This allows for a much clearer request and response as opposed to simply making an update request against the node's "authorized" property.

Collections

Collections represent resources which are contained within other resources.

For example, Nodes within a Node Group

https://localhost/HDID/master/NodeManager/objects/NodeGroups/:id/collections/Nodes

Or Files and Directories within a FileSystemItem.

https://localhost/HDID/master/FileSystemBrowser/objects/FileSystemItems/:id/collections/FilteSystemItems

Some collections allow the user to add or remove items from within it. Others are fixed and cannot be changed. Depending on the resource the collection may be managed by adding and removing a list of resource Ids or by adding and removing complete objects.

Filtering

When retrieving a list of resources it is possible to apply a filter using the "query" parameter within the URL.

It is possible to filter on properties deep within the resource structure by using 'dot notation'. For example, filtering for authorized nodes can be achieved with the following query:

https://localhost/HDID/master/NodeManager/objects/Nodes/?query=stateInfo.authorized=true

The query syntax supports AND, OR and grouping (using brackets) to support complex queries.

Note: It is only possible to filter on supported properties. Please refer to the API Reference Guide for further information.

Sorting

When retrieving a list of resources it is possible to apply a sort order using the "order-by" parameter within the URL. This will order the data by a single property either ascending (ASC) or descending (DESC). The order keyword is required and failure to provide this will result in an error stating: "Invalid order-by parameter. Invalid format".

Note: It is only possible to sort on a single supported property within any one request. Please refer to the API Reference Guide for further information.

For example, retrieving the list of nodes by ordered alphabetically by name:

https://localhost/HDID/master/NodeManager/objects/Nodes/?order-by=name+ASC

Fields

When retrieving a resource or list of resources it is possible to request a subset of properties from the resources using the "fields" parameter within the URL. Multiple fields can be specified by comma separating the field names.

Note: Not all handlers allow retrieving specific resources properties from either a list or individual item requests. Please refer to the API Reference Guide for further information.

For example, retrieving only the name of all nodes:

https://localhost/HDID/master/NodeManager/objects/Nodes/?fields=name

Count and Offset

When retrieving a list of resources it is possible to limit the amount of resources returned by using the "count" and "offset" parameters within the URL. By default resources are returned in an arbitrary order so it is often useful to also provide an "order-by" parameter.

It is possible to request a list of resources with both count and offset of zero. This will return the pageInfo object containing the total number of resources which matched the query.

For example, retrieving the second page of fifty nodes:

https://localhost/HDID/master/NodeManager/objects/Nodes/?count=50&offset=50

Error Responses

Invalid requests will receive a JSON error response as well as the appropriate HTTP status code.

Error responses contain an error message detailing the error along with the handler on which the error was generated.

Templating Concept

HDID node, policy, data flow and schedule resources exposed by the API are built up of a number of smaller objects that result in a complex object with many optional attributes and variable depth.

Creating these resources manually can be difficult and prone to error. To aid in the creation of these resources it is advised that the user create a template within the HDID GUI and use these templates to create specialized versions of the resources for the specific use cases.

The templates would then be used as the framework to creating new resources based on the template.

For example a policy could be created containing a path classification and backup operation with an eight hour RPO or a specific schedule. The path classification would simply contain a placeholder value. The API can then be used to retrieve the resource, modify the path classification values and save the policy as a new resource. This allows the operation settings to be copied and applied to a new policy without having to redefine all of its properties.

API Reference Guide

The API Reference guide can be used to view the descriptions, properties and URL details of resources.

Resources consist of properties. Each property can be a basic data type (string, number, boolean), an enumeration, array or an object. Objects are documented within the reference guide as “Data Classes”.

Enumerations are sent and received as a string. The API reference guide details all possible values of an enumeration and a relevant description.

Data classes represent basic objects which have properties, but cannot be retrieved, created, updated or deleted directly and only exist as a property of a parent resource.

Examples

In the following examples the master node is called hdiddemo2.hds.com.

The Curl argument ‘—insecure’ is used to bypass the need to setup valid SSL certificates.

Examples cover the following:

- Login to HDID
- List Nodes
- Create a repository node on the master
- Create a policy
- Edit a policy
- Create a data flow
- Compile and activate a data flow
- Trigger a backup
- List jobs
- Restore to an alternate location
- View a session summary report
- Logout

Login

When logging into HDID a session cookie is returned, the examples will use the CURL cookie-jar feature to save and use the session information in future calls.

Login request

```
curl --request POST --insecure --cookie-jar hdidsession --data
"username=user&password=password&space=master"
"https://hdiddemo2.hds.com/HDID/master/UIController/services/User/action
s/login/invoke"
```

List Nodes

This will list all nodes. Given this is a new installation, only the master will be listed.

List all nodes

```
curl --insecure --cookie hdidsession
“https://hdiddemo2.hds.com/HDID/master/NodeManager/objects/Nodes”
```

List all nodes - response


```
{
    "node": [
        {
            "id": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-1]",
            "name": "master",
            "resourceId": "4294967297",
            "agentless": false,
            "version": "6.0.0.55027-R6.0",
            "osDetails": {
                "os": "eOS_WINDOWS",
                "osVersion": "Standard",
                "osVariant": "Windows 2008 R2",
                "architecture": "x64",
                "chassis": "Virtual",
                "osLevel": "Server"
            },
            "networkDetails": {
                "iPAddresses": [
                    "10.70.254.76"
                ]
            },
            "stateInfo": {
                "authorized": true,
                "hasRules": false,
                "upToDate": true,
                "hubState": "Connected",
                "activeRules": 0,
                "connected": true,
                "accessible": true
            },
            "master": true,
            "type": "OSHost",
            "filterAvailable": true,
            "nodeAttributes": []
        }
    ],
    "pageInfo": {
        "totalCount": 1,
        "end": true
    }
}
```

Create Repository node on the master

Find master node name

To create a repository node on the master the master node id is required. Get a list of nodes filtered so only the node required and its id is received.

List only the master node and ask for just the ID

```
curl --insecure --cookie hdidsession
“https://hdiddemo2.hds.com/HDID/master/NodeManager/objects/Nodes?query=(
master=true)&fields=id”
```

List only the master node and ask for just the ID - Response

```
{
    "node": [
        {
            "id": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-1]"
        }
    ],
    "pageInfo": {
        "totalCount": 1,
        "end": true
    }
}
```

Create repository node using the highlighted id for the proxy node

Note, because some POST commands need a large amount of JSON this is defined in a file and the file is passed to CURL using the @ facility so the JSON can be formatted and is readable.

Create the repository on the master node

```
curl --request POST --insecure --cookie hdidsession --data
"@createrepo.json" --header "Content-Type: application/json"
"https://hdiddemo2.hds.com/HDID/master/NodeManager/objects/RepositoryAge
ntlessNodes"
```

Contents of createrepo.json

```
{
      "nodeType":"Repository",
      "name":"repository1",
      "resourceGroup":null,
      "configuration":
      {
            "proxyName":"master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-1]",
            "import":false,
            "rootPath":"C:\\repository1",
            "encrypted":false
      }
}
```

Create the repository on the master node - Response

```
{
    "timestamp": "2017-10-10T17:54:21",
    "name": "repository1",
    "id": "repository1@00-D123F1-64A4C1-4EBE94-C8980B[1-1-5]",
    "configuration": {
        "proxyName": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-1]",
        "rootPath": "C:\\repository1",
        "import": false,
        "encrypted": false
    }
}
```

Note, the repository ID will be used when creating the dataflow.

`repository1@00-D123F1-64A4C1-4EBE94-C8980B[1-1-5]`

Create Policy

First we will create a backup policy to backup C:\testdata and have an RPO for 8 hours, we will then edit the policy and change the RPO to 1 hour.

Create policy

```
curl --request POST --insecure --cookie hdidsession --data
"@createpolicy.json" --header "Content-Type: application/json"
"https://hdiddemo2.hds.com/HDID/master/PolicyHandler/objects/Policies/"
```

Contents of createpolicy.json


```
{
	"name":"Test Policy",
	"description":"A test policy created from CURL",
	"classificationChains":[
	{
		"items":[
		{
			"category":"Data",
			"rules":
			{
				"andAny":true,
				"andList":[],
				"orMode":"eCLASSIFICATION_OR_FIRST_ONLY",
				"orAny":true,
				"orList":[]
			},
			"displayName":"Path",
			"propertyName":"path",
			"normalizedTypeName":"path",
			"_selected":false,
			"typeName":"path",
			"itemType":"Classification",
			"path":
			{
				"include":"C:\\testdata",
				"exclude":""
			},
			"_policyItemType":"classifications",
			"_hasFilters":false
		}]
	}],
	"operationChains":[
	{
		"items":[
		{
			"category":"Standard",
			"displayName":"Backup",
			"propertyName":"backup",
			"normalizedTypeName":"backup",
			"_selected":false,
			"typeName":"Backup",
			"itemType":"Operation",
			"backup":
			{
```

```
                        "label":"Backup",
                        "run":
                        {
                              "mode":"eRUN_ANYTIME"
                        },
                        "recoveryPointObjective":
                        {
                              "period":8,
                              "units":"eRPO_HOURS"
                        },
                        "retention":
                        {
                              "period":6,
                              "units":"eRETENTION_MONTHS"
                        },
                        "recoveryTimeObjective":
                        {
                              "incrementalTapeBackup":false
                        },
                        "sourceOption":
                        {
                              "quiesceApplications":true,
                              "preScriptEnabled":false,
                              "postScriptEnabled":false
                        },
                        "type":
                        {
                              "mode":"eTRANSFER_ASYNCHRONOUS_JOURNALED"
                        }

                  },
                  "_policyItemType":"operations"
            }]
      }]
}
```


Create policy - Response


```
data
{
    "id": "f254f8a32fc34c6098e6f6410734f026",
    "classificationChains": [
        {
            "items": [
                {
                    "typeName": "path",
                    "path": {
                        "include": "C:\\testdata",
                        "exclude": ""
                    },
                    "id": 1,
                    "propertyName": "path"
                }
            ]
        }
    ],
```

```
    "operationChains": [
          {
              "items": [
                  {
                      "typeName": "Backup",
                      "backup": {
                          "label": "Backup",
                          "type": {
                              "mode": "eTRANSFER_ASYNCHRONOUS_JOURNALED"
                          },
                          "retention": {
                              "units": "eRETENTION_MONTHS",
                              "period": 6
                          },
                          "recoveryPointObjective": {
                              "units": "eRPO_HOURS",
                              "period": 8
                          },
                          "recoveryTimeObjective": {
                              "incrementalTapeBackup": false
                          },
                          "run": {
                              "mode": "eRUN_ANYTIME"
                          },
                          "sourceOption": {
                              "quiesceApplications": true,
                              "preScriptEnabled": false,
                              "postScriptEnabled": false
                          }
                      },
                      "id": 1,
                      "propertyName": "backup"
                  }
              ]
          }
      ],
      "description": "A test policy created from CURL",
      "nextClassificationId": 2,
      "nextOperationId": 2,
      "version": 1,
      "permissions": [
          {
              "propertyType": "pt_user",
              "pt_user": {
                  "upn": "Administrator@master"
              },
              "access": "eA_READWRITE"
          }
      ],
      "name": "Test Policy"
}
```

The policy ID will be required when editing the policy and creating the dataflow.

`f254f8a32fc34c6098e6f6410734f026`

Edit Policy

Editing a policy is the same as creating except the policy ID is required and a PUT is used not a POST. We will change the RPO from 8 days to 1 day.

Edit policy

```
curl --request PUT --insecure --cookie hdidsession --data
"@editpolicy.json" --header "Content-Type: application/json"
"https://hdiddemo2.hds.com/HDID/master/PolicyHandler/objects/Policies/f2
54f8a32fc34c6098e6f6410734f026"
```

Where the contents of editpolicy.json is:

Contents of editpolicy.json

```
curl --request PUT --insecure --cookie hdidsession --data

{
	"id":"f254f8a32fc34c6098e6f6410734f026",
	"version":1,
	"name":"Test Policy",
	"description":"A test policy created from CURL",
	"classificationChains":[
	{
		"items":[
		{
			"category":"Data",
			"rules":
			{
				"andAny":true,
				"andList":[],
				"orMode":"eCLASSIFICATION_OR_FIRST_ONLY",
				"orAny":true,
				"orList":[]
			},
			"displayName":"Path",
			"propertyName":"path",
			"normalizedTypeName":"path",
			"_selected":false,
			"typeName":"path",
			"itemType":"Classification",
			"path":
			{
				"include":"C:\\testdata",
				"exclude":""
			},
			"_policyItemType":"classifications",
			"_hasFilters":false
		}]
	}],
```

```
        "operationChains":[
        {
                "items":[
                {
                        "category":"Standard",
                        "displayName":"Backup",
                        "propertyName":"backup",
                        "normalizedTypeName":"backup",
                        "_selected":false,
                        "typeName":"Backup",
                        "itemType":"Operation",
                        "backup":
                        {
                                "label":"Backup",
                                "run":
                                {
                                        "mode":"eRUN_ANYTIME"
                                },
                                "recoveryPointObjective":
                                {
                                        "period":1,
                                        "units":"eRPO_HOURS"
                                },
                                "retention":
                                {
                                        "period":6,
                                        "units":"eRETENTION_MONTHS"
                                },
                                "recoveryTimeObjective":
                                {
                                        "incrementalTapeBackup":false
                                },
                                "sourceOption":
                                {
                                        "quiesceApplications":true,
                                        "preScriptEnabled":false,
                                        "postScriptEnabled":false
                                },
                                "type":
                                {
                                        "mode":"eTRANSFER_ASYNCHRONOUS_JOURNALED"
                                }

                        },
                        "_policyItemType":"operations"
                }]
        }]
}
```

Edit policy - Response


```
{
    "id": "f254f8a32fc34c6098e6f6410734f026",
    "classificationChains": [
        {
            "items": [
                {
                    "typeName": "path",
                    "path": {
                        "include": "C:\\testdata",
                        "exclude": ""
                    },
                    "id": 1,
                    "propertyName": "path"
                }
            ]
        }
    ],
"operationChains": [
        {
            "items": [
                {
                    "typeName": "Backup",
                    "backup": {
                        "label": "Backup",
                        "type": {
                            "mode": "eTRANSFER_ASYNCHRONOUS_JOURNALED"
                        },
                        "retention": {
                            "units": "eRETENTION_MONTHS",
                            "period": 6
                        },
                        "recoveryPointObjective": {
                            "units": "eRPO_HOURS",
                            "period": 1
                        },
                        "recoveryTimeObjective": {
                            "incrementalTapeBackup": false
                        },
                        "run": {
                            "mode": "eRUN_ANYTIME"
                        },
                        "sourceOption": {
                            "quiesceApplications": true,
                            "preScriptEnabled": false,
                            "postScriptEnabled": false
                        }
                    },
                    "id": 1,
                    "propertyName": "backup"
                }
            ]
        }
    ],

    "description": "A test policy created from CURL",
    "nextClassificationId": 2,
    "nextOperationId": 2,
    "version": 2,
```

```
    "permissions": [
        {
            "propertyType": "pt_user",
            "pt_user": {
                "upn": "Administrator@master"
            },
            "access": "eA_READWRITE"
        }
    ],
    "name": "Test Policy"
}
```

Create Data Flow

The data flow will be the master node implementing the create policy going to the repository via a batch mover. The node and policy IDs determined in the previous steps are used here.

Create Data Flow

```
curl --request POST --insecure --cookie hdidsession --data
"@createdataflow.json" --header "Content-Type: application/json"
"https://hdiddemo2.hds.com/HDID/master/DataFlowHandler/objects/DataFlows
/"
```

Createdataflow.json contents

```
{
	"data":
	{
		"name":"Test Dataflow",
		"description":"Dataflow created from CURL",
		"nodes":[
		{
			"name":"master",
			"type":"OSHost",
			"nodeId":"master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-
1]",
			"id":"c1",
			"assignedPolicies":[
			{
				"id":"f254f8a32fc34c6098e6f6410734f026",
				"operations":[]
			}]
		},
		{
			"name":"repository1",
			"type":"Repository",
			"nodeId":"repository1@00-D123F1-64A4C1-4EBE94-
C8980B[1-1-5]",
			"id":"c2",
			"assignedPolicies":[
			{
				"id":"f254f8a32fc34c6098e6f6410734f026",
				"operations":[
				{
					"id":1,
					"propertyType":"ot_repositoryDestination",
					"ot_repositoryDestination":
					{
						"templateId":"000000000-Standard
Repository Store"
					}
				}]
			}]
		},
```

```
            "connections":[
            {
                  "label":"",
                  "type":"eCT_BATCH",
                  "netCompression":false,
                  "id":"c3",
                  "source":"c1",
                  "destination":"c2",
                  "routedPolicies":["f254f8a32fc34c6098e6f6410734f026"]
            }]
      }
}
```


Create Data Flow - Response


```
{
    "id": "0d048895-5fb6-471f-9c25-3a3275d4bafa",
    "version": 1,
    "data": {
        "name": "Test Dataflow",
        "connections": [
            {
                "id": "36",
                "type": "eCT_BATCH",
                "source": "34",
                "destination": "35",
                "bandwidthSettings": {
                    "hoursOfWeek": [],
                    "throttleDefaultSpeed": false
                },
                "label": "",
                "routedPolicies": [
                    "f254f8a32fc34c6098e6f6410734f026"
                ],
                "netCompression": false
            }
        ],
"nodes": [
            {
                "id": "34",
                "nodeId": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-
1]",
                "drawInstructions": {
                    "x": 0,
                    "y": 0
                },
                "assignedPolicies": [
                    {
                        "id": "f254f8a32fc34c6098e6f6410734f026",
                        "operations": []
                    }
                ],
                "isDestination": false,
                "type": "OSHost",
                "isGroup": false
            },
```

```
                {
                  "id": "35",
                  "nodeId": "repository1@00-D123F1-64A4C1-4EBE94-C8980B[1-1-
5]",
                  "drawInstructions": {
                      "x": 0,
                      "y": 0
                  },
                  "assignedPolicies": [
                      {
                          "id": "f254f8a32fc34c6098e6f6410734f026",
                          "operations": [
                              {
                                  "id": 1,
                                  "ot_repositoryDestination": {
                                      "templateId": "000000000-Standard
Repository Store"
                                  },
                                  "propertyType":
"ot_repositoryDestination",
                                  "operationType": ""
                              }
                          ]
                      }
                  ],
                  "isDestination": true,
                  "type": "Repository",
                  "isGroup": false
              }
          ],
          "description": "Dataflow created from CURL",
          "compilerDirectives": ""
      },
      "permissions": [
          {
              "propertyType": "pt_user",
              "pt_user": {
                  "upn": "Administrator@master"
              },
              "access": "eA_READWRITE"
          }
      ],
      "numInProgress": 0,
      "numNotifications": 0,
      "isActive": false,
      "numOffline": 0,
      "versionActive": 0
}
```

The data flow ID will be needed to compile the data flow.

`0d048895-5fb6-471f-9c25-3a3275d4bafa`

Compile and Activate Data Flow

This is a two stage process, first compile and then distribute the data flow.

Compile data flow

```
curl --request PUT --insecure --cookie hdidsession --data
"{\"ids\":[\"0d048895-5fb6-471f-9c25-3a3275d4bafa\"]}" --header
"Content-Type: application/json"
“https://hdiddemo2.hds.com/HDID/master/RulesManager/services/Rules/actio
ns/compile/invoke”
```

Compile data flow - Response

```
{
    "summary": [
        "Compiling rules for data flow 'Test Dataflow'",
        "Rules summary for data flow 'Test Dataflow'",
        "'master' implements:",
        "    'Test Policy'",
        "'master' sends:",
        "    'Test Policy' from the local filesystem to 'master'",
        "'repository1' (on master) receives:",
        "    'Test Policy' from 'master' for repository store
'Standard'",
        "Compilation of data flow 'Test Dataflow' completed
successfully"
    ],
    "warnings": [],
    "errors": []
}
```

Distribute data flow

```
curl --request PUT --insecure --cookie hdidsession --data
"{\"ids\":[\"0d048895-5fb6-471f-9c25-3a3275d4bafa\"]}" --header
"Content-Type: application/json"
“https://hdiddemo2.hds.com/HDID/master/RulesManager/services/Rules/actio
ns/distribute/invoke”
```

There is no response body to the action distribute

Trigger Backups

To trigger a backup first get the list of triggers and use the response to populate the trigger request.

List triggers

```
curl --request GET --insecure --cookie hdidsession
"https://hdiddemo2.hds.com/HDID/master/TriggerHandler/objects/
Triggers"
```

List triggers - response


```
{
    "trigger": [
        {
            "id": "0",
            "policyId": "f254f8a32fc34c6098e6f6410734f026",
            "policyName": "Test Policy",
            "operationId": 1,
            "operationName": "Backup",
            "dataflowName": "Test Dataflow",
            "dataflowId": "0d048895-5fb6-471f-9c25-3a3275d4bafa",
            "sourceNodeId": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-
1]",
            "sourceNodeInstance": 34,
            "destinationNodeId": "repository1@00-D123F1-64A4C1-4EBE94-
C8980B[1-1-5]",
            "destinationNodeInstance": 35,
            "moverType": "eMOVER_BATCH",
            "syncGroup": "",
            "dataOriginNodeInstance": 34,
            "dataOriginNodeId": "master@BY-Z99ZTX-97WXQO-IAQP47-
EHRPJI[0-1-1]"
        }
    ],
    "pageInfo": {
        "totalCount": 1,
        "end": true
    }
}
```

Trigger backup

```
curl --request PUT --insecure --cookie hdidsession --data
"@trigger.json" --header "Content-Type: application/json"
“https://hdiddemo2.hds.com/HDID/master/TriggerHandler/services/Triggers/
actions/triggeroperation/invoke”
```

Trigger.json contents

```
{
      "sourceNodeId":"master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-1]",
      "dataOriginNodeId":"master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-1]",
      "dataflowId":"0d048895-5fb6-471f-9c25-3a3275d4bafa",
      "policyId":"f254f8a32fc34c6098e6f6410734f026",
      "operationId":1,
      "moverType":"eMOVER_BATCH",
      "syncGroup":""
}
```

Trigger backup - Response

```
 {
    "job": [
        {
            "id": "{d7ea2271-2852-4893-98f2-c4f67c98a8d6}"
        }
    ]
}
```

List Jobs

List Jobs

```
curl --request GET --insecure --cookie hdidsession
"https://hdiddemo2.hds.com/HDID/master/JobStatusHandler/objects/
Jobs"
```

List Jobs - Response


```
"job": [
        {
            "id": "{2fb32a26-3917-4d38-b62f-dea101355ee5}",
            "parameters": [
                {
                    "key": "source",
                    "value": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-
1]"
                },
                {
                    "key": "destination",
                    "value": "repository1@00-D123F1-64A4C1-4EBE94-
C8980B[1-1-5]"
                },
                {
                    "key": "sessionId",
                    "value": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-
1]#1507724302569-3324-2"
                },
                {
                    "key": "policyId",
                    "value": "f254f8a32fc34c6098e6f6410734f026"
                },
                {
                    "key": "policyVersion",
                    "value": "2"
                },
                {
                    "key": "policyName",
                    "value": "Test Policy"
                },
                {
                    "key": "dataflowId",
                    "value": "0d048895-5fb6-471f-9c25-3a3275d4bafa"
                },
                {
                    "key": "dataflowName",
                    "value": "Test Dataflow"
                },
```

```
                {
                    "key": "moverType",
                    "value": "eMOVER_BATCH"
                },
                {
                    "key": "operationId",
                    "value": "1"
                }
            ],
            "parentId": "{d7ea2271-2852-4893-98f2-c4f67c98a8d6}",
            "timeStarted": "2017-10-11T12:18:24Z",
            "timeUpdated": "2017-10-11T12:18:27Z",
            "description": "Initial Resynchronization of FileSystem Data
to Repository Store",
            "actions": [],
            "timeCompleted": "2017-10-11T12:18:27Z",
            "progress": "Stage 13 of 13",
            "type": "eJOBTYPE_BACKUP",
            "status": "eJOB_SUCCEEDED",
            "subType": {
                "id": "Resynchronize Repository Store",
                "subSystem": "Repository"
            },
            "mainNode": {
                "id": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-1]",
                "instance": 34
            },
            "reporterNode": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-1]"
        }
    ],
    "pageInfo": {
        "totalCount": 1,
        "end": true
    }
}
```

Restore a backup to an alternate location

To perform a restore first list the backups to get the snapshot ID of the backup required to restore

List Repository backups

```
curl --request GET --insecure --cookie hdidsession
"https://hdiddemo2.hds.com/HDID/master/MetaDataAggregator/objects/FullRe
coverableData/"
```

List Repository backups - Response


```
{
    "fullRecoverableData": [
        {
            "moverType": "eMOVER_BATCH",
            "sourceNodeType": "OSHost",
            "sourceOsType": "eOS_WINDOWS",
            "id": "repository1@00-D123F1-64A4C1-4EBE94-C8980B[1-1-
5]gQAAAAAAAAABQAAAAAAAAAMAAAACAAAAAAAAAAAAAAA=",
            "sourceCaptureDate": "2017-10-11T12:03:26Z",
            "policyName": "Test Policy",
            "snapshotId":
"gQAAAAAAAAABQAAAAAAAAAMAAAACAAAAAAAAAAAAAAA=",
            "expiryDate": "2018-04-11T12:03:26Z",
            "sourceNodeId": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-
1]",
            "rbacOwner": {
                "type": "eRESOURCE",
                "id": "4294967297"
            },
            "captureDate": "2017-10-11T12:03:29Z",
            "storageLocationType": "repository",
            "sourcePropertyName": "",
            "repository": {
                "storeName": "Standard",
                "storeId": "16385",
                "outOfSync": false
            },
            "storageLocationName": "repository",
            "metaDataStoreName": "mds",
            "dataType": "eDATA_FILESYSTEM",
            "storageObjectName": "snapshots",
            "storageNodeId": "repository1@00-D123F1-64A4C1-4EBE94-
C8980B[1-1-5]",
            "policyId": "f254f8a32fc34c6098e6f6410734f026",
            "hadProblems": false,
            "policyVersion": 2,
            "operationId": 1,
            "operationType": "eOPERATION_BACKUP"
        }

    ],
```

```
    "pageInfo": {
        "totalCount": 1,
        "end": true
    }
}
```


Restore to alternate location

```
curl --request PUT --insecure --cookie hdidsession --data
"@restore.json" --header "Content-Type: application/json"
"https://hdiddemo2.hds.com/HDID/master/RestoreHandler/services/Restores/
actions/restore/invoke"
```

Restore.json contents


```
{
      "fileSystemParameters":
      {
            "destination":
            {
                  "destinationNodeId":"master@BY-Z99ZTX-97WXQO-IAQP47-
EHRPJI[0-1-1]",
                  "useAltLocation":true,
                  "path":"C:\\restored"
            },
            "route":false,
            "collisionPolicy":"eRH_CP_OVERWRITERENAME",
            "recoverableData":[
            {
                  "storageLocationName":"repository",
                  "storageNodeId":"repository1@00-D123F1-64A4C1-4EBE94-
C8980B[1-1-5]",
                  "id":"gQAAAAAAAAABQAAAAAAAAAMAAAACAAAAAAAAAAAAAAA=",

      "snapshotID":"gQAAAAAAAAABQAAAAAAAAAMAAAACAAAAAAAAAAAAAAA=",
                  "includeItems":[],
                  "excludeItems":[],
                  "sourcetype":"eRH_DT_FILESYSTEM"
            }]
      },
      "type":"eRH_DT_FILESYSTEM"
}
```


Restore to alternate location - Response (this is the parent job id)


```
{
    "id": "{6d47db69-2f70-4b21-80f5-502c3744ffdc}"
}
```

Delete Snapshot

```
curl --request DELETE --insecure --cookie hdidsession
“https://hdiddemo2.hds.com/HDID/repository1@00-D123F1-64A4C1-4EBE94-
C8980B%5B1-1-
5%5D/Repository/objects/Snapshots/gQAAAAAAAAABQAAAAAAAAAMAAAACAAAAAAAAAA
AAAAA%3D/”
```

Delete Snapshot Response


```
{
    "id": "delete_16385_2"
}
```

View Session summary report

View the session summary report

```
curl --request GET --insecure --cookie hdidsession
"https://hdiddemo2.hds.com/HDID/master/ReportHandler/objects/SessionSumm
aryReports/0/collections/entries”
```

View the session summary report - Response


```
{
    "sessionEntry": [
        {
            "type": "Initial",
            "sourceNodeId": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-
1]",
            "moverNodeId": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-
1]",
            "destinationNodeId": "repository1@00-D123F1-64A4C1-4EBE94-
C8980B[1-1-5]",
            "destinationStoreType": "eST_FILESYSTEM_BATCH",
            "policyName": "Test Policy",
            "timeStarted": "2017-10-11T12:13:04Z",
            "timeEnded": "2017-10-11T12:13:11Z",
            "destinationStore": "Standard",
            "operationType": "eOPERATION_BACKUP",
            "operationId": "1",
            "activityStartTime": "2017-10-11T12:13:06Z",
            "activityEndTime": "2017-10-11T12:13:11Z",
            "status": "eSESSION_SUCCESS",
            "activity": "eACTIVITY_BACKUP",
            "dataActivity": {
                "startTime": "2017-10-11T12:13:11Z",
                "endTime": "2017-10-11T12:13:11Z",
                "transferred": "27321",
                "processed": "24347",
                "scanned": "0"
            },
"fileActivity": {
                "createdFileCount": "0",
                "createdDirectoryCount": "0",
                "modifiedFileCount": "0",
                "modifiedDirectoryCount": "0",
                "deletedFileCount": "0",
                "deletedDirectoryCount": "0",
                "scannedFileCount": "0",
                "scannedDirectoryCount": "0"
            },
```

```
            "id": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-
1]#1507723984576-3324-1",
            "operationName": "Backup",
            "policyId": "f254f8a32fc34c6098e6f6410734f026",
            "policyRevision": 2
        },
        {
            "type": "Initial",
            "sourceNodeId": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-
1]",
            "moverNodeId": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-1]",
            "destinationNodeId": "repository1@00-D123F1-64A4C1-4EBE94-
C8980B[1-1-5]",
            "destinationStoreType": "eST_FILESYSTEM_BATCH",
            "policyName": "Test Policy",
            "timeStarted": "2017-10-11T12:18:22Z",
            "timeEnded": "2017-10-11T12:18:27Z",
            "destinationStore": "Standard",
            "operationType": "eOPERATION_BACKUP",
            "operationId": "1",
            "activityStartTime": "2017-10-11T12:18:24Z",
            "activityEndTime": "2017-10-11T12:18:27Z",
            "status": "eSESSION_SUCCESS",
            "activity": "eACTIVITY_BACKUP",
            "dataActivity": {
                "startTime": "2017-10-11T12:18:27Z",
                "endTime": "2017-10-11T12:18:27Z",
                "transferred": "27321",
                "processed": "24347",
                "scanned": "0"
            },
            "fileActivity": {
                "createdFileCount": "0",
                "createdDirectoryCount": "0",
                "modifiedFileCount": "0",
                "modifiedDirectoryCount": "0",
                "deletedFileCount": "0",
                "deletedDirectoryCount": "0",
                "scannedFileCount": "0",
                "scannedDirectoryCount": "0"
            },
            "id": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-
1]#1507724302569-3324-2",
            "operationName": "Backup",
            "policyId": "f254f8a32fc34c6098e6f6410734f026",
            "policyRevision": 2
        },
        {
            "type": "Restore",
            "sourceNodeId": "repository1@00-D123F1-64A4C1-4EBE94-C8980B[1-
1-5]",
            "destinationNodeId": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-
1-1]",
            "timeStarted": "2017-10-11T12:43:29Z",
            "timeEnded": "2017-10-11T12:43:29Z",
            "activityStartTime": "2017-10-11T12:43:29Z",
            "activityEndTime": "2017-10-11T12:43:29Z",
            "status": "eSESSION_SUCCESS",
            "activity": "eACTIVITY_RESTORE",
            "dataActivity": {
                "startTime": "2017-10-11T12:43:29Z",
                "endTime": "2017-10-11T12:43:29Z",
                "transferred": "60"
            },
```

```
            "fileActivity": {
                "createdFileCount": "0",
                "createdDirectoryCount": "0",
                "modifiedFileCount": "0",
                "modifiedDirectoryCount": "0",
                "deletedFileCount": "0",
                "deletedDirectoryCount": "0"
            },
            "id": "master@BY-Z99ZTX-97WXQO-IAQP47-EHRPJI[0-1-
1]#1507725809445-3920-0"
        }
    ],
    "pageInfo": {
        "offset": 0,
        "count": 3,
        "totalCount": 3,
        "end": true
    }
}
```

Logout

Logout

```
curl --request POST --insecure --cookie-jar hdidsession
"https://hdiddemo2.hds.com/HDID/v1.0.0/master/UIController/services/User
/actions/logout/invoke"
```

There is no response body to the logout action.

Hitachi Vantara

Corporate Headquarters
2845 Lafayette Street
Santa Clara, CA 95050-2639 USA
www.HitachiVantara.com
community.HitachiVantara.com

Regional Contact Information
Americas: +1 866 374 5822 or info@hitachivantara.com
Europe, Middle East and Africa: +44 (0) 1753 618000 or info.emea@hitachivantara.com
Asia Pacific: +852 3189 7900 or info.marketing.apac@hitachivantara.com